MARTIN RUSS

SOUND SYNTHESIS AND SAMPLING

THIRD EDITION

Sound Synthesis and Sampling

This page intentionally left blank

Sound Synthesis and Sampling

Third Edition

Martin Russ

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK
OXFORD • PARIS • SAN DIEGO • SAN FRANCISCO • SINGAPORE
SYDNEY • TOKYO

Focal Press is an imprint of Elsevier

Focal Press

Focal Press is an imprint of Elsevier
Linacre House, Jordan Hill, Oxford OX2 8DP, UK
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

First edition 1996
Reprinted 1998, 1999, 2000 (twice), 2002 (twice)
Second edition 2004
Reprinted 2005, 2006
Third edition 2009

Notice
No responsibility is assumed by the publisher for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Control Number: 2008936153

ISBN: 978-0-240-52105-3

For information on all Focal Press publications
visit our website at www.focalpress.com

Typeset by Charon Tec Ltd., A Macmillan Company.
(www.macmillansolutions.com)

Printed and bound in the USA

09 10 11 12   12 11 10 9 8 7 6 5 4 3 2 1

Contents

This page intentionally left blank

Preface to First Edition

This is a book about sound synthesis and sampling. It is intended to provide a reference guide to the many techniques and approaches that are used in both commercial and research sound synthesizers. The coverage is more concerned with the underlying principles, so this is not a ‘build your own synthesizer’ type of book, nor is it a guide to producing specific synthesized sounds. Instead it aims to provide a solid source of information on the diverse and complex field of sound synthesis. In addition to the details of the techniques of synthesis, some practical applications are described to show how synthesis can be used to make sounds.

It is designed to meet the requirements of a wide range of readers, from enthusiasts to undergraduate level students. Wherever possible, a non-mathematical approach has been taken, and the book is intended to be accessible to readers without a strong scientific background.

This book brings together information from a wealth of material, which I have been collecting and compiling for many years. Since the early 1970s I have been involved in the design, construction and use of synthesizers. More recently this has included the reviewing of electronic musical instruments for *Sound On Sound*, the leading hi-tech music recording magazine in the United Kingdom.

The initial prompting for this book came from Francis Rumsey of the University of Surrey’s Music Department, with support from Margaret Riley at Focal Press. I would like to thank them for their enthusiasm, time and encouragement throughout the project. I would also like to thank my wife and children for putting up with my disappearance for long periods over the last year.

Martin Russ, February 1996

This page intentionally left blank

Preface to Second Edition

This second edition has revised and updated all of the material in the first edition, correcting a few minor errors, and adding a completely new chapter on performance aspects (Chapter 8), which shows how synthesizers have become embedded within more sophisticated musical performance instruments, rather than always being stand-alone synthesizers *per se*. This theme is also explored further in the extended 'Future of Synthesis' chapter.

I have strived to maintain the abstraction of the techniques away from specific manufacturers, and with only a few exceptions, the only place where details of individual instruments or software will be found is in the 'Examples' sections at the end of each chapter.

Taking a cue from other books in the Focal Press Music Technology series, I have added additional notes alongside the main text, as well as panels which are intended to reinforce significant points.

I must thank Beth Howard and others at Focal Press who have helped me to finish this edition. Their patience and support has been invaluable.

I would also like to thank the many readers, reviewers and other sources of feedback for their suggestions – as many as possible of these have been incorporated in this edition. I welcome additional suggestions for improvement, as well as corrections – please send these to me via Focal Press.

Martin Russ, October 2003

This page intentionally left blank

Preface to Third Edition

This is a book about sound synthesis and sampling, rather than one about synthesizers and samplers. This has always been a significant difference to me.

So, if you want to know about the what, who, where and when of synthesizers and samplers then there is some information in this book, and there are many other resources available: books, manufacturer's brochures, the Internet, etc.

But if, like me, you want to know about the how and why of synthesis and sampling, and how they fit into the overall context of sound-making, then this is the place. I am one of those people who just has to know how something works, and how it fits into the overall process, in order to be able to use it. So this book is nothing more than my attempt to try and understand how sounds can be made and how they can be used to make music.

The first edition of this book was published in 1996. The second edition was published in 2004, 8 years later. This edition is being published after an interval of only 4 years, which reflects the rapid changes that have taken place with the move to sound-making on personal computers.

Since the second edition was published, I have had more feedback from readers, and it was especially nice to be able to talk to some of them in person. As usual, this has helped to steer the direction of this new edition, and I would like to thank everyone who has helped me to write it.

Martin Russ, May 2008

This page intentionally left blank

Visual Map

Applications

About this Book

This book is divided into nine chapters, followed by References, a guide to Jargon, and finally, an Index. The Jargon section is designed to try and prevent the confusion that often results from the wide variation in the terminology which is used in the field of synthesizers. Each entry consists of the term which is used in this book, followed by the alternative names which can be used for that term. Previous editions of this book have also included a glossary, which has increased in size and complexity so much that it has now been moved into a different medium: the Internet. You will find more details at my website: http://www.martinruss.com

Book guide

The chapters can be divided into five major divisions:

1. Background
2. Techniques
3. Applications
4. Analysis
5. Reference

Background: Chapter 1 sets the background, and places synthesis in a historical perspective.
Techniques: Chapters 2–6 describe the main methods of producing and manipulating sound, arranged in an approximate historical ordering.
Applications: Chapters 7 and 8 show how the techniques described can be used to synthesize sound and music, in a range of locations from fixed studios to mobile live performance.
Analysis: Chapter 9 provides analysis of the development of sound synthesis and some speculation on future developments.
Reference: References, link to the online Glossary, Guide to Jargon and Index.

Chapter guide

Chapter 1 Background

This chapter introduces the concept of synthesis, and briefly describes the history. It includes brief overviews of acoustics, electronics, digital sampling and instrument digital interface (MIDI).

Chapter 2 Making Sounds Physically

This chapter goes back to the fundamentals of making sounds using physical methods: hitting, scraping, twanging and blowing. It also looks at how mechanical methods can be used to control, record and reproduce sounds.

Chapter 3 Making Sounds with Analogue Electronics

This chapter describes the main methods which are used for analogue sound synthesis: Subtractive, Additive, AM, FM, Ring Modulation, Ringing Oscillators and others. It also looks at analogue techniques for sound sampling and recording.

Chapter 4 Making Sounds with Hybrid Electronics

This chapter shows the way that synthesis, sampling and recording techniques changed from the primarily analogue electronic circuit designs of the 1960s and 1970s to the predominantly digital circuitry of the 1980s and 1990s. Synthesizers and samplers whose design incorporates a mixture of both design techniques are included.

Chapter 5 Making Sounds with Digital Electronics

This chapter looks at the major techniques which are used for digital sound synthesis: FM, Waveshaping, Physical Modeling, Granular, FOF, Analysis–Synthesis and Resynthesis. It also looks at the convergence between sampling and synthesis that led to S&S (Sampling & Synthesis) synthesizers.

Chapter 6 Making Sounds with Computer Software

This chapter covers the rise of the personal computer from a simple sequencer accessory used in conjunction with hardware to a complete integrated recording studio implemented entirely in software.

Chapter 7 Sound-Making Techniques

This chapter deals with the use of synthesis, sampling and recording to make music and other sounds.

Chapter 8 Controllers

This chapter looks at the ways that sound-making equipment can be controlled and used in live performance.

Chapter 9 The Future of Sound-Making

This chapter attempts to place sound synthesis in a wider context, by describing the probable development of music hardware and software in the future.

Chapter section guide

Within each chapter, there are sections which deal with specific topics. The format and intention of some of these may be unfamiliar to the reader, and thus they deserve special mention.

Environment

The chapters that cover the different types of synthesis and sampling are split into two parts. The first part describes the sound-making techniques, whilst the second part describes the physical environment relevant to that technique. This places the technique in context. For example, the chapter covering analogue synthesis and sampling also covers analogue sequencing and recording.

Examples

These sections are illustrated with block diagrams of the internal function and front panel controls of some representative example instruments or software, together with some notes on their main features. This should provide a more useful idea of their operation than just black and white photographs. Further information and photographs of a wide range of synthesizers and other electronic musical instruments can be found on the Internet. A historical snapshot of the 1980s can be found in Julian Colbeck's comprehensive 'Keyfax' books (Colbeck, 1985) or Mark Vail's 'Vintage Synthesizers' (Vail, 1993) retrospective book, which is a collection of articles from the American magazine *Keyboard*.

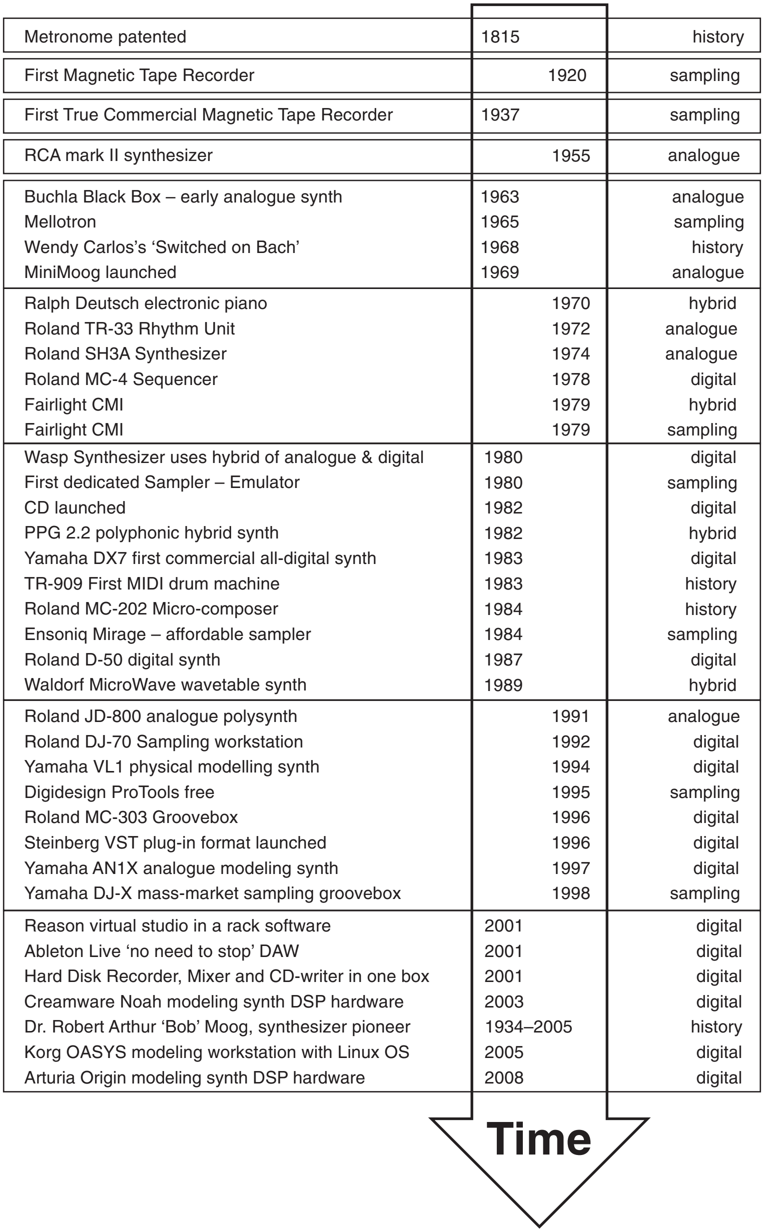
Time line

The Time Lines are intended to show the development of a topic in a historical context. Reading text which contains lots of references to dates and people can be confusing. By presenting the major events in time order, the developments and relationships can be seen more clearly. The time lines are deliberately split up so that only entries relevant to each chapter are shown. This keeps the material in each individual time line succinct.

Overall timeline

Chapters 2–6 of this book do not represent a precise historical record, even though the apparent progression from analogue, via hybrid, to digital and software synthesis methods is a compelling metaphor. Synthesis techniques, like fashion, regularly recycle the old and mostly forgotten with 'retro' revivals of buzzwords like FM, analogue, valves, FETs, modular, resynthesis and more.

The overall timeline shown here is intended to show just some of the complex flow of the synthesis timeline.



Metronome patented	1815	history
First Magnetic Tape Recorder	1920	sampling
First True Commercial Magnetic Tape Recorder	1937	sampling
RCA mark II synthesizer	1955	analogue
Buchla Black Box – early analogue synth	1963	analogue
Mellotron	1965	sampling
Wendy Carlos's 'Switched on Bach'	1968	history
MiniMoog launched	1969	analogue
Ralph Deutsch electronic piano	1970	hybrid
Roland TR-33 Rhythm Unit	1972	analogue
Roland SH3A Synthesizer	1974	analogue
Roland MC-4 Sequencer	1978	digital
Fairlight CMI	1979	hybrid
Fairlight CMI	1979	sampling
Wasp Synthesizer uses hybrid of analogue & digital	1980	digital
First dedicated Sampler – Emulator	1980	sampling
CD launched	1982	digital
PPG 2.2 polyphonic hybrid synth	1982	hybrid
Yamaha DX7 first commercial all-digital synth	1983	digital
TR-909 First MIDI drum machine	1983	history
Roland MC-202 Micro-composer	1984	history
Ensoniq Mirage – affordable sampler	1984	sampling
Roland D-50 digital synth	1987	digital
Waldorf MicroWave wavetable synth	1989	hybrid
Roland JD-800 analogue polysynth	1991	analogue
Roland DJ-70 Sampling workstation	1992	digital
Yamaha VL1 physical modelling synth	1994	digital
Digidesign ProTools free	1995	sampling
Roland MC-303 Groovebox	1996	digital
Steinberg VST plug-in format launched	1996	digital
Yamaha AN1X analogue modeling synth	1997	digital
Yamaha DJ-X mass-market sampling groovebox	1998	sampling
Reason virtual studio in a rack software	2001	digital
Ableton Live 'no need to stop' DAW	2001	digital
Hard Disk Recorder, Mixer and CD-writer in one box	2001	digital
Creamware Noah modeling synth DSP hardware	2003	digital
Dr. Robert Arthur 'Bob' Moog, synthesizer pioneer	1934–2005	history
Korg OASYS modeling workstation with Linux OS	2005	digital
Arturia Origin modeling synth DSP hardware	2008	digital

Questions

Each chapter ends with a few questions, which can be used as either a quick comprehension test or a guide to the major topics covered in that chapter.

PART 1

Background

This page intentionally left blank

CHAPTER 1

Background

1.1 What is synthesis?

'Synthesis' is defined in the 2003 edition of the *Chambers 21st Century Dictionary* as 'building up; putting together; making a whole out of parts'. The process of synthesis is thus a bringing together, and the 'making a whole' is significant because it implies more than just a random assembly: synthesis should be a creative process. It is this artistic aspect, which is often overlooked in favor of the more technical aspects of the subject. Although a synthesizer may be capable of producing almost infinite varieties of output, controlling and choosing them requires human intervention and skill.

The word 'synthesis' is frequently used in just two major contexts: the creation of chemical compounds and production of electronic sounds. But there are a large number of other types of synthesis.

1.1.1 Types

All synthesizers are very similar in their concept – the major differences are in their output formats and the way they produce that output. For example, some of the types of synthesizers are as follows:

- Texture synthesizers, used in the graphics industry, especially in 3D graphics.
- Video synthesizers, used to produce and process video signals.
- Color synthesizers, used as part of 'son et lumiere' presentations.
- Speech synthesizers, used in computer and telecommunications applications.
- Sound synthesizers, used to create and process sounds and music.
- Word synthesizers, more commonly known as authors using 'word processor' software!

CONTENTS

Context

Technology

Synthesizers have two basic functional blocks: a 'control interface', which is how the parameters that define the end product are set; and a 'synthesis engine', which interprets the parameter values and produces the output. In most cases there is a degree of abstraction involved between the control interface and the synthesis engine itself. This is because the complexity of the synthesis process is often very high, and it is often necessary to reduce the apparent complexity of the control by using some sort of simpler conceptual model. This enables the user of the synthesizer to use it without requiring a detailed knowledge of the inner workings. This idea of models and abstraction of interface is a recurring theme which will be explored many times in this book (Figure 1.1.1).

1.1.2 Sound synthesis

Sound synthesis is the process of producing sound. It can reuse existing sounds by processing them, or it can generate sound electronically or mechanically. It may use mathematics, physics or even biology; and it brings together art and science in a mix of musical skill and technical expertise. Used carefully, it can produce emotional performances, which paint sonic landscapes with a rich and huge set of timbres, limited only by the imagination and knowledge of the creator. Sounds can be simple or complex, and the methods used to create them are diverse. Sound synthesis is not solely concerned with sophisticated computer-generated timbres, although this is often the most publicized aspect. The wide availability of high-quality recording and synthesis technology has made the generation of sounds much easier for musicians and technicians, and future developments promise even easier access to ever more powerful techniques. But the technology is nothing more than a set of tools that can be used to make sounds: the creative skills of the performer, musician or technician are still essential to avoid music becoming mundane.

This chapter introduces the concept of synthesis, and briefly describes the history. It includes brief overviews of acoustics, electronics, digital sampling and musical instrument digital interface (MIDI).

Many members of the general public have unrealistic expectations of the capabilities of synthesizers. The author has encountered feedback comments such as 'I thought it did it all by itself!' when he has shown that he can indeed 'play' a synthesizer.

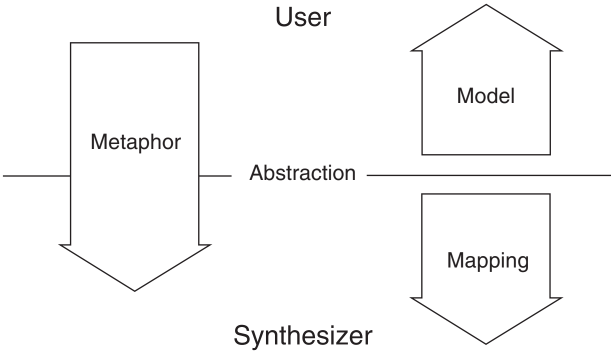


FIGURE 1.1.1 The user uses a metaphor in order to access the functions of the synthesizer. The synthesizer provides a model to the user and maps this model to internal functionality. This type of abstraction is used in a wide variety of electronic devices, particularly those employing digital circuitry.

1.1.3 Synthesizers

Sounds are synthesized using a sound synthesizer. The synthesis of sounds has a long history. The first synthesizer might have been an early ancestor of *Homo sapiens* hitting a hollow log, or perhaps learning to whistle. Singers use a sophisticated synthesizer whose capabilities are often forgotten: the human vocal tract. All musical instruments can be thought of as being 'synthesizers', although few people would think of them in this context. A violin or clarinet is viewed as being 'natural', whereas a synthesizer is seen as 'artificial', although all of these instruments produce sound by essentially synthetic methods.

Although synthesizer can be spelt with a '-zer' or '-ser' ending, the '-zer' ending will be used in this book. Also, the single word 'synthesizer' is used here to imply 'sound synthesizer', rather than a generic synthesizer.

Recently, the word 'synthesizer' has come to mean only an electronic instrument that is capable of producing a wide range of different sounds. The actual categories of sounds that qualify for this label of synthesizer are also very specific: purely imitative sounds are frequently regarded as nothing other than recordings of the actual instrument, in which case the synthesizer is seen as little more than a replay device. In other words, the general public seems to expect synthesizers to produce 'synthetic' sounds. This can be readily seen in many low-cost keyboard instruments which are intended for home usage: they typically have a number of familiar instrument sounds with names such as 'piano', 'strings' and 'guitar'. But they also have sounds labeled 'synth' for sounds that do not fit into the 'naturalistic' description scheme. As synthesizers become better at fusing elements of real and synthetic sounds, the boundaries of what is regarded as 'synthetic' and what is 'real' are becoming increasingly diffuse.

Note that the production of a wide range of sounds by a synthesizer can be very significant. An 'electronic musical instrument' that produces a restricted range of sounds can often be viewed as being more musically acceptable.

This blurred perception has resulted in broad acceptance of a number of 'hyper-real' instrument sounds, where the distinctive characteristics of an instrument are exaggerated. Fret buzz and finger noise on an acoustic guitar and breath noise on a flute are just two examples. Drum sounds are frequently enhanced and altered considerably, and yet, unless they cross that boundary line between 'real' and 'synthetic', their generation is not questioned – it is assumed to be 'real' and 'natural'. This can cause considerable difficulties for performers who are expected to reproduce the same sound as the compact disk (CD) in a live environment. The actual sound of many live instruments may be very different from the sound that is 'expected' from the familiar recording that was painstakingly created in a studio. Drummers are an example: they may have a physical drum kit where many parts of the kit are present merely to give a visual cue or 'home' to the electronically generated sounds that are being controlled via triggers connected to the drums, and where the true sound of the real drums is an unwanted distraction.

The electronic piano is an example, where the same synthesis capability could be packaged in two different ways, and would consequently be sold separately to synthesists and piano players.

Forms

Synthesizers come in several different varieties, although many of the constituent parts are common to all of the types. Most synthesizers have one or more audio outputs; one or more control inputs; some sort of display; and buttons or dials to select and control the operation of the unit. The significant difference between performance and modular forms are as follows:

- Performance synthesizers have a standard interconnection of their internal synthesis modules already built-in. It is usually not possible to change this significantly, and so the signal flow always follows a set path through the synthesizer. This enables the rapid patching of commonly used configurations, but does limit the flexibility. Performance synthesizers form the vast majority of commercial synthesizer products.
- Conversely, modular synthesizers have no fixed interconnections, and the synthesis modules can be connected together in any way. Changes can be made to the connections whilst the synthesizer is making a sound, although the usual practice is to set up and test the interconnections in advance. Because more connections need to be made, modular synthesizers are harder and more time-consuming to set up, but they do have much greater flexibility. Modular synthesizers are much rarer than performance synthesizers, and are often used for academic or research purposes.

Non-ideal interfaces are actually very common. The 'qwerty' typewriter keyboard was originally intended to slow down typing speeds and thus help prevent the jamming of early mechanical typewriters. It has become dominant (and commercially, virtually essential!) despite many attempts to replace it with more ergonomically efficient alternatives. The music keyboard has also seen several carefully human engineered improvements which have also failed to gain widespread acceptance. It is also significant that both the qwerty and music keyboards have become well-accepted metaphors for computers/information and music in general.

Both performance and modular synthesizers can come with or without a music keyboard. The keyboard has become the most dominant method of controlling the performance aspect of a synthesizer, although it is not necessarily the ideal controller. Synthesizers that do not have a keyboard (or any other type of controller device) are often referred to as expanders or modules, and these can be controlled either by a synthesizer, which does have a keyboard, or from a variety of other controllers. It has been said that the choice of a keyboard as the controller was probably the biggest setback to the wide acceptance of synthesizers as a musical instrument. Chapter 7 describes some of the alternatives to a keyboard.

1.1.4 Sounds

Synthesized sounds can be split into simple categories such as 'imitative' or 'synthetic'. Some sounds will not be easy to place in a definite category, and this is especially true for sounds, which contain elements of both real and synthetic sounds.

Imitative sounds often sound like real instruments, and they might be familiar orchestral or band instruments. In addition, imitative sounds may be more literal in nature, the sound effects. In contrast, synthetic sounds will often be unfamiliar to anyone who is used to hearing only real instruments, but over time a number of clichés have been developed: the 'string synth' and 'synth brass' are just two examples.

Synthetic sounds, depending on their purpose, can be divided into various types.

'Imitations' and 'emulations'

'Imitations' and 'emulations' are intended to provide many of the characteristics of real instruments, but in a sympathetic way where the synthesis is frequently providing additional control or emphasis on significant features of the sound. Sometimes an emulation may be used because of tuning problems, or difficulties in locating a rare instrument. The many 'electronic' piano sounds are examples of an emulated sound.

'Suggestions' and 'hints'

'Suggestions' and 'hints' are sounds where the resulting sound has only a slight connection with any real instrument. The 'synth brass' sound produced by analogue polyphonic synthesizers in the 1970s is an example of a sound where just enough of the characteristics of the real instrument are present and thus strongly suggest a 'brass'-type instrument to an uncritical listener, but where a detailed comparison immediately highlights the difference to a real brass instrument.

'Alien' and 'off-the-wall'

'Alien' and 'off-the-wall' sounds are usually entirely synthetic in nature. The cues which enable a listener to determine if a sound is synthetic are complex, but are often related to characteristics that are connected with the physics of real instruments: unusual or unfamiliar harmonic structures and their changes over time; constancy of timbre over a wide range; and pitch change without timbre change. By deliberately moving outside of the physical limitations of conventional instruments is noise-like.

Noise-like

Of course, most synthesizers can also produce variations on 'noise', of which 'white noise' is perhaps the most un-pitched and unusual sound of all, since it has the same sound energy in linear frequency bands across the entire audible range. Any frequency-dependent variation of the harmonic content of a noise-like sound can give it a perceivable 'pitch', and it thus becomes playable.

All of these types of synthetic sounds can be used to make real sounds more interesting by combining the sounds into a hybrid composite (see Chapter 6).

Factory presets

One final category of sound is found only in commercial sound synthesizers: the factory sounds that are intended to be used as demonstrations of the broad capabilities of the instrument when it is being auditioned by a potential purchaser. These sounds are typically produced rapidly at a later stage in the production process, and are not always a good guide to the true potential sound-making capabilities of the synthesizer. They also frequently suffer from a number of problems which are directly related to their design specification; they can be buried underneath excessive amounts of reverberation, they may

Naming sounds is not as straightforward as it might appear at first. For example, if you have more than two or three piano sounds, then manufacturer's name or other adjectives tend to be used: 'Steinway piano' or 'Detuned pub piano' are simple examples. For sounds that are more synthetic in nature, the adjectives become more dense, or are abandoned altogether in favor of names which suggest the type of sound rather than try and describe it: 'crystal spheres' and 'thudblock' are two examples.

use carefully timed cyclic variations and echo effects for special effects, and they are rarely organized in category groupings, favoring instead contrast and variation. Some techniques for making use of these sounds are described in Chapter 6.

In marked contrast, the factory sounds for samplers and sample-based instruments are intended for use in performance and are the result of careful recording and editing. So a multi-sampled grand piano 'preset' in a digital piano is almost the opposite of a synthesizer factory preset: it is intended to produce as accurate a playable reproduction of that one specific sound source as possible.

1.1.5 Synthesis methods

There are many techniques that can be used to synthesize sound. Many of them use a 'source and modifier' model as a metaphor for the process which produces the sound: a raw sound source produces the basic tone, which is then modified in some way to create the final sound. Another name for this model is the 'excitation and filter' model, as used in speech synthesis. The use of this model can be seen most clearly in analogue subtractive synthesizers, but it can also be applied to other methods of synthesis, for example, sample and synthesis (S&S) or physical modeling. Some methods of synthesis are more complex: frequency modulation (FM), harmonic synthesis, Fonctions d'Onde Formantiques (FOF) (see Section 5.5) and granular synthesis. For these methods, the metaphors of a model can be more mathematical or abstracted, and thus may be more difficult to comprehend. This may be one of the reasons why the 'easier to understand' methods such as subtractive synthesis and its derived variant called S&S have been so commercially successful.

1.1.6 Analogue synthesis

Understanding how a synthesis technique works is essential for the adjustment (tweaking) of sounds to suit a musical context, and also knowing how the sound can be controlled in performance. This is just as much a part of the synthesists toolkit as playing ability.

'Analogue' refers to the use of audio signals, which can be produced using elements such as oscillators, filters and amplifiers. Analogue synthesis methods can be divided into three basic areas, although there are crossovers between them. The basic types are as follows:

1. subtractive
2. additive
3. wavetable.

Subtractive synthesis takes a 'raw' sound, which is usually rich in harmonics, and filters it to remove some of the harmonic content. The raw sounds are traditionally simple mathematical waveshapes: square, sawtooth, triangle and sine, although modern subtractive synthesizers tend to provide longer 'samples' instead of single cycles of waveforms. The filtering tends to be a resonant low-pass filter, and changing the cut-off frequency of this filter produces the characteristic (and clichéd) 'filter sweep' sound, which is strongly associated with subtractive synthesis.

Additive

Additive synthesis adds together lots of sine waves with different frequencies to produce the final timbre. The main problem with this method is the complexity of controlling large numbers of sine waves, but see also the section 'Additive'in Section 1.1.7.

The word 'analogue' can also be spelt without the '-ue' ending. In this book, the longer version will be used.

Wavetable

Wavetable synthesis extends the ideas of subtractive synthesis by providing much more sophisticated waveshapes as the raw starting point for subsequent filtering and shaping. More than one cycle of a waveform can be stored, or many waveforms can be arranged so that they can be dynamically selected in real time – this produces a characteristic 'swept' sound which can be subtle, rough, metallic or even glassy in timbre.

1.1.7 Digital synthesis

Digital technology replaces signals with numerical representations, and uses computers to process those numbers. Digital methods of synthesizing sounds are more varied than analogue methods, and research is still continuing to find new ways of making sounds. Some of the types that may be encountered include the following:

- FM
- wavetable
- sample replay
- additive
- S&S
- physical modeling
- software synthesis.

FM

FM is the technical term for the way that FM radio works, where the audio signal of music or speech is used to modulate a high-frequency carrier signal which then broadcasts the audio as part of a radio signal. In audio FM, both signals are at audio frequencies, and complex frequency mirroring, phase inversions and cancellations happen that can produce a wide range of timbres. The main problem with FM is that it is not possible to program it 'intuitively' without a lot of practice, but its major advantage in the 1970s was that it required very little memory to store a large number of sounds. With huge falls in the cost of storage, this is no longer as crucially important in the 2000s. FM was used in some sound cards and portable keyboards, and like many synthesis techniques, its marketability seems to be influenced by the cycles of fashionability.

In fact, most of the effects that audio FM uses are exactly the sort of distortions and problems that you try to avoid in radio FM!

Wavetable

Wavetable synthesis uses the same idea as the analogue version, but extends the basic idea into more complex areas. The waveshapes are usually complete but short segments of real samples, and these can be looped to provide sustained sections of sound, or several segments of sound can be joined together to produce a composite 'sample'. Often this is used to 'splice' the start of one sound onto the sustained part of another. Because complete samples are not used, this method makes very efficient use of available memory space, but this results in a loss of quality. Wavetable synthesis is used in low-cost, mass-market sound cards and MIDI instruments.

Sample replay

Sample replay is the ultimate version of wavetable. Instead of looping short samples and splicing them together, sample replay does just that: it replays complete samples of sounds, with a loop for the sustained section of the sound. Sample replay uses lots of memory, and was thus initially used in more expensive sound cards and MIDI instruments only. Falling prices for memory (allegedly driven strongly downwards by huge sales of cartridges for video games consoles in the 1980s and 1990s) have led to sample replay becoming very widespread. Sample replay is often referred to by other names: AWM (Advanced Wave Memory), AWM2, RS-PCM etc.

Additive

Digital techniques make the task of coping with lots of sine waves much easier, and digital additive synthesizers have been more successful than analogue versions, but they are still a very specialised field. There are very few synthesizers that use only additive synthesis, but additive is often an element within another type of synthesis, or can be part of a palette of techniques.

S&S

S&S is an acronym for 'samples and synthesis', and uses the techniques of wavetable and sample replay, but adds in the filtering and shaping of subtractive synthesis in a digital form. This method is widely used in MIDI instruments, sound cards and professional electronic musical instruments, although it is rarely referred to as 'S&S'. Instead, the marketing departments at synthesizer manufacturers will create a term that suggests the properties of innovation and differentiation: Hyper Integrated (HI), Expanded Articulation (XA), AI2 and VX are some examples.

The term 'physical modeling' is still used where a mathematical model of an instrument is produced from the physics of that instrument, but the word 'modeling' has become a generic term for any mathematical modeling technique that can be applied to synthesis.

Physical modeling

Physical modeling uses mathematical equations which attempt to describe how an instrument works. The results can be stunningly realistic, very synthetic or a mixture of both. The most important feature is the way the model responds in much the same way as a real instrument; hence the playing techniques of the real instrument can often be employed by a performer.

Initially the 'real' instruments chosen were exactly that, and then plucked, hit and blown instruments were modeled to varying degrees of accuracy; but once these were established, then models of analogue synthesizers and even valve amplifiers and effects units began to develop. The high processing demands of modeling meant that it was only found in professional equipment in the mid-1990s. But it rapidly became more widely adopted, and by the start of the twenty-first century it could be found, albeit in a simplified form, in low-cost keyboards intended for home usage, as well as computer sound cards, although in professional equipment, highly developed models are used to produce an increasingly wide range of 'modeled' sounds, instruments, amplifiers, effects, environments and loudspeakers. Physical modeling is another term that is rarely used by manufacturers. Instead, terms such as Virtual Circuit Modeling (VCM), VariOS and Multi Modeling Technology (MMT) are used.

Software synthesis

In the 2000s, the use of powerful general-purpose computers as audio processing and synthesis devices has given physical modeling a new role: software synthesis. Here, the computer replaces almost all of the 'traditional' equipment that might be expected by a time traveler from the 1970s. The computer can now integrate the functions of a sequencer for the notes, a synthesizer or sample-replay device to produce the sounds, a mixer to combine the sounds from several synthesizers or sample-replay devices, and process the mixed audio through effects-processing units, hard disk recording to capture the audio and CD 'burning' software to produce finished CDs. The synthesizers and effects often use physical modeling techniques to emulate an analogue synthesizer, an analogue reverb line and more. All of these functions are carried out on digital signals, entirely within the computer – conversion to analogue audio is needed only for monitoring playback, and in the case of the CD, the audio signal output of the CD player is typically the first time that the audio signal has ever been in an analogue format. Chapters 6 and 9 explores this topic in more detail.

1.2 Beginnings

The beginnings of sound synthesis lie with the origins of the human, Homo sapiens, species. Many animals have an excellent hearing sense, and this serves a diverse variety of purposes: advance warning of danger, tracking prey and communication. In order to be effective, hearing needs to monitor the essential parts of the audio spectrum. This can involve very low frequencies in some underwater animals or ultrasonic frequencies for echo location purposes in bats; the dynamic range required can be very large.

Human hearing is more limited. Sounds from 15 Hz to 18 kHz can be heard, although this varies with the sound level, and the upper limit reduces with age. The dynamic range is more than 120 dB, which is a ratio of 1012:1. With two ears and the complex processing carried out by the brain, the average human being can accurately locate sounds and resolve frequency to fractions

Using distance as an analogy, a ratio of 1012:1 is equivalent to the ratio between one million kilometers and one millimeter.

of a hertz, although this performance is dependent on the frequency, level and other factors.

Human beings also have a sophisticated means of producing sound: the vocal tract. The combination of vocal cords, throat, tongue, teeth, mouth cavity and lips provides a versatile way of making a wide variety of sounds: a biological sound synthesizer. The development of this particular instrument is long and still ongoing – it is probably the oldest and most important musical instrument (Figure 1.2.1).

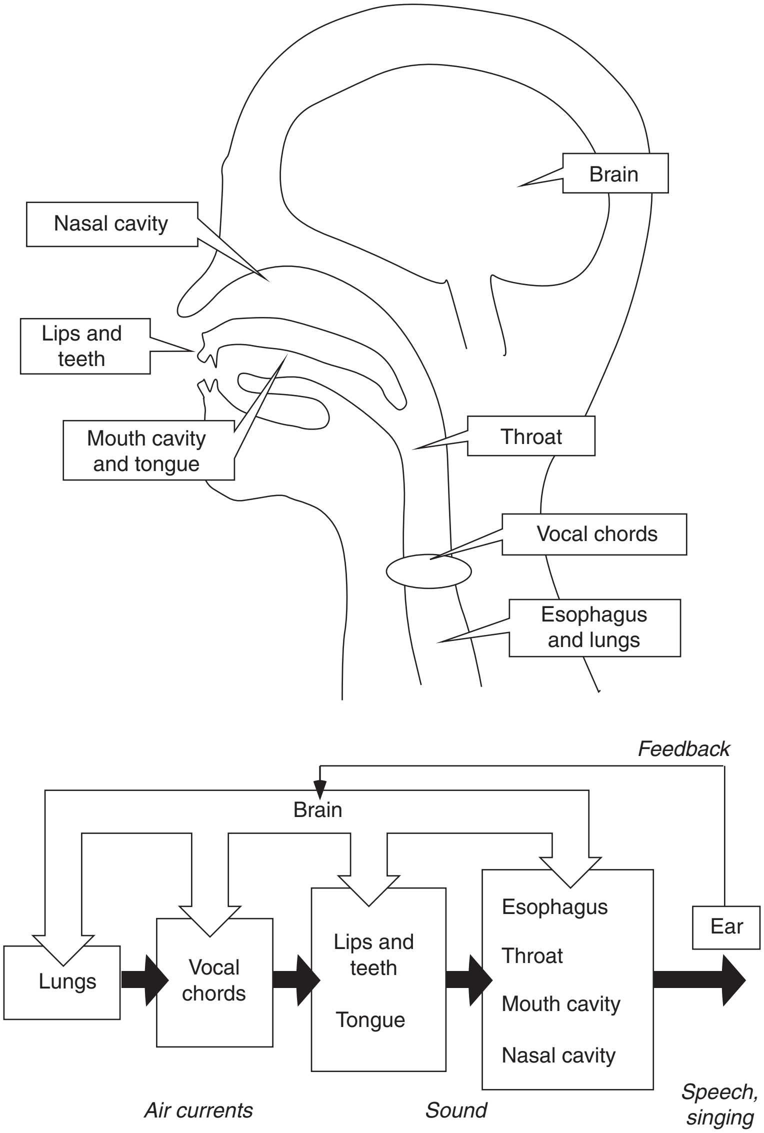


FIGURE 1.2.1
The human voice is a complex and sophisticated synthesizer capable of producing both speech and singing sounds. The main sound source is the vocal cords, although some sounds are produced by the interactions between the lips, tongue and teeth with air currents. The throat, nose, mouth, esophagus and lungs form a set of resonant cavities that filter the sounds, and the mouth shape is dynamically variable.

The mixture of sophisticated hearing and an inbuilt sound synthesizer, plus the everyday usage via speech, singing or whistling, makes the human being a perceptive and interactive listener. The human voice is part of a feedback loop created by the ears and brain. The brain not only controls the vocal tract to make the sounds, but also listens to the sounds created and adjusts the vocal tract dynamically. This analysis–synthesis approach is also used in resynthesis, as described in Section 5.6. The combination of sound production and analysis forms a powerful feedback mechanism; and it seems that knowing how to make sounds is an essential part of inferring the intended meaning when someone else makes the sounds.

Making sounds and listening to sounds is a fundamental part of human interactivity. Using the example of a human being from conception onwards, it is possible to see the range of possibilities:

- *Listening*: Pregnant mothers are often aware that sudden noises can startle a baby in the womb.
- *Mouth*: Most parents will confirm that babies are capable of making a vocal noise from just after birth!
- *Shaking*: Once control over hands and feet is possible, a baby will investigate objects by interacting with them. The rattle is specifically designed to provide an audible feedback when it is shaken.
- *Singing*: Part of the process of learning to speak involves long periods of experimentation by the infant, where the range of possible sounds is explored.
- *Speech*: The 'singing' sounds are then reduced down to the set of sounds which are heard from the parent's own speech.
- *Blowing*: Blowing (and spitting!) is part of the learning process for making speech sounds. Blowing into tubes and whistles may lead to playing real musical instruments.
- *Percussive pitching*: 'Open mouth' techniques for making sounds include slapping the cheek or top of the head, or tapping the teeth. The throat and mouth cavity are then altered to provide the pitching of the resulting sound.
- *Whistling*: Whistling requires the mastering of a musical instrument which is created by the lips.

These have been arranged in an approximate chronological order, although the development of every human being is different. The important information here is the wide range of possible ways that sounds can be made, and the degree of control which is possible. Singing and whistling are both highly expressive musical instruments, and it is no accident that many musical instruments also use the mouth as part of their control mechanism. With such a broad collection of sounds, humankind has developed a rich and diverse repertoire of musical and spoken sounds.

The 'electric' guitar is analogous in some ways to the electric piano, and the method of extracting the sound with a coil-based pickup system is very similar (the piano's rods are replaced by metal strings). Note that in an electric guitar, the sound production system is unchanged; the pickups produce an electrical signal that represents the vibration of the strings, whereas an electric piano has replaced the strings with metal rods which are held at only one end, and so are slightly different to the strings of a conventional piano. Contact microphones placed on the frame of the piano itself, or even just microphones placed near the piano, rather than coil-based pickups, are often used when a conventional strung piano requires amplification.

Beyond this human-oriented synthesis, there are many possibilities for making other sounds. Striking a log or other resonant object will produce a musical tone, and blowing across and through tubes can make a variety of sounds. A bow and arrow may be useful for hunting, but it can also produce an interesting twanging sound as well. From these, and a large number of other ordinary objects, human beings have produced a number of different families of musical instruments and this process is still continuing.

In the twentieth century, a number of new instruments have been developed: the electric piano is an example. The word 'electric' is almost a misnomer for this particular instrument, because the actual sound is produced by metal rods vibrating near coils of wire, and thus electromagnetically inducing a voltage in the coils. No electricity is used to produce the sound – it is merely that the output of an electric piano is primarily an electrical signal rather than an acoustic one, and so it needs to be amplified in order to be heard. Naturally, such an instrument could not have been produced before electricity came into common usage, since it depends on amplifiers and loudspeakers.

The synthesizer is even more dependent on technology. Advances in electronics have accelerated its development, and so the transition from simple valve-based oscillators to sophisticated digital tone generators using custom silicon chips has taken less than 100 years. If semiconductors are taken as the starting point of electronics, then the major developments in the electronic music synthesizer have actually occurred in the last half of the twentieth century. If mass-market synthesis hardware is the criteria, then the major developments have taken place in the last quarter of the twentieth century. If software synthesis is taken as the enabler for truly flexible sound creation, then this has only been widely available in the past 10 years of the century and the first years of the twenty-first century.

1.3 Telecoms research

Much of the research effort expended by the telecommunications industry in the last century has been focused on sound, since the transmission of the human voice has been the major source of revenue. With the advent of reliable digital transmission techniques, communications are becoming increasingly computer oriented. But the human voice is likely to remain one of the major sources of traffic for the foreseeable future.

Although the invention of the telephone showed that it was possible to transmit the human voice from one location to another by electrical means, this was not the only reason for commercialization of telephone. Familiarity now with the telephone makes it difficult to appreciate how strange the concept of talking to someone at a distance was at the time: why not go and talk to them face-to-face instead? But one of the major driving forces behind the adoption of the telephone was actually musical – the telephone made it possible to broadcast a musical performance to many people. Again, long usage of radio and television has removed any sense of wonder about being able to hear a concert without actually being there. But at the turn of the century this was amazing!

Thaddeus Cahill's Teleharmonium is an example of how telecommunications was used to provide musical entertainment. Developed from prototypes in the 1890s, the 1906 commercial version in New York was essentially a large set of power generators, which produced electrical signals at various frequencies, and these could be distributed along telephone lines for the subscribers to listen to. The teleharmonium can be thought of as a 200-ton organ connected to lots of telephones rather than just one loudspeaker. As microphone and instrument technology developed, live performances by musicians could also be distributed in the same way. Without competition from radio, the ability to be able to talk to someone by telephone might have been seen as nothing more than a curious side effect of this musical distribution system.

Violins fitted with diaphragms and conical horns to provide mechanical amplification, notably expressed in John Matthias Augustus Stroh's patent of 1899, were used to overcome the limited sensitivity of early microphones. These instruments are often misinterpreted as musical curiosities because they look like a mixture of a violin and a record player.

Telecommunication approaches sound from a technical viewpoint, and thus a great deal of research was put into developing improved performance microphones and loudspeakers, as well as increasing the distance over which the sound could be carried. Speech is intelligible with levels of distortion that would make music almost impossible to listen to. Thus as the telephone began to be used more and more for speech communications, the research tended to concentrate on the speech transmission. This is one of the reasons that the telephone of today has a restricted bandwidth and dynamic range: it is designed to produce an acceptable level of speech intelligibility, but in as small a bandwidth as possible. The bandwidth of 300 Hz to 3.4 kHz is still the underlying standard for basic fixed-line telephony, but the experience of mobile telephony shows that sound quality can be lowered even further, whilst still retaining acceptability, if there is a perceived gain in functionality.

One example of the way that telecommunications research can be used for electronic musical purposes is the invention of the vocoder. Bell Telephone Laboratories invented the vocoder in the 1930s as a way of trying to process audio signals. The word comes from 'VOice enCODER', and the idea was to try and split the sound into separate frequency bands and then transmit these more efficiently. It was not successful at the time, although many modern military communication systems use digital descendants of vocoder technology. But the vocoder was rediscovered and adopted by electronic music composers in the 1950s. By the 1950s, telephones were in wide use for speech, and the researchers turned back to the musical opportunities offered by telephony. Lord Rayleigh's influential work *The Theory of Sound* had laid the foundations for the science of acoustics back in 1878, and Lee de Forest's triode amplifier of 1906 provided the electronics basis for controlling sound. E. C. Wente's condenser microphone in 1915 provided the first high-quality audio microphone, and the tape recorder provided the means to store sounds.

Jacquard used holes punched in cards to control weaving machines in the early 1800s.

At the German Radio station NWDR in Cologne in 1951, Herbert Eimert began to use the studio's audio oscillators and tape recorders to produce electronically generated sounds. Rather than assembling test gear, researchers at the Radio Corporation of America (RCA) Laboratories in the United States produced a dedicated modular synthesizer in 1955, which was designed to simplify the tedious production process of creating sounds by using automation. A mark II model followed in 1957, and this was used at the Columbia-Princeton Electronic Music Centre for some years. Although the use of punched holes in paper tape to control the functions now appears primitive, the RCA synthesizer was one of the first integrated systems for producing electronic music.

Work at Bell Telephone Laboratories in the 1950s and 1960s led to the development of pulse code modulation (PCM), a technique for digitizing or sampling sound and thus converting it into digital form. As is usual in telecommunications technology, the description and its acronym, pulse code modulation and PCM, are in a technical language that conveys little to the non-engineer. What PCM does is actually very straightforward. An audio signal is 'sampled' at regular intervals, and this gives a series of voltage values: one for each interval. These voltages represent the value of the audio signal at the instant that the sample was made. These voltages are converted into numbers, and these numbers are then converted into a series of electrical pulses, where the number and organization of the pulses represent the size of the voltage. PCM thus refers to the coding scheme used to represent the numbers as pulses. (You may like to compare PCM with the pulse width modulation (PWM) as described in Chapter 3.)

PCM forms the basis of sampling. A great deal of work was done to formalize the theory and practice of converting audio signals into digital numbers. The concept of sampling at twice the highest wanted frequency is called the Nyquist criterion after the work published by Nyquist and others. The filtering required to prevent unwanted frequencies being heard (they are a consequence of the sampling process) was also developed as a result of telecommunications research. Further work in the 1970s led to the invention of the digital signal processor (DSP), a specialised microprocessor chip which was produced in order to carry out the complex numerical calculations which were needed to enable audio coding algorithms to be developed. DSPs have since then been used to produce many types of digital synthesizer. It is interesting to note that the wide availability, powerful processing capability and low cost of more general-purpose processors have gradually reduced the need to use DSPs for audio processing in personal computers (PCs). This means that a typical PC of the 2000s will contain a variety of audio processing software 'codecs' for use in telecommunications as well as for audio and music playback. The word 'codec' is derived from COder and DECoder. Voice over IP (VOIP) codecs, which allow telephone quality audio to be transmitted over an IP network, are a major telecommunications use. Highly compressed but almost CD-quality music via the MPEG codec colloquially known as 'MP3' has also made music transmission over telecommunications networks accessible to all.

Current telecommunications research continues to explore the outer most limits of acoustics, physics and electronics, although since telecommunications is now almost solely concerned with computers, the emphasis is increasingly on data communications between computers, but human communication still forms much of that data.

1.4 Tape techniques

1.4.1 The analogue tape recorder

The analogue tape recorder has been a major part of electronic music synthesis almost from the very beginning. It enables the user to splice together small sections of magnetic tape which represent audio, and then replay the results. This has the important elements of 'building up from small parts' that is the basis of the definition of synthesis.

The principle of the tape recorder is not new. The audio signal is converted into a changing magnetic field, which is then stored onto iron. Early examples recorded onto iron wire, then onto steel ribbon. There were also experiments with the use of paperbacked tape, but the most significant breakthrough was the use of plastic coated with magnetic material, which was developed in Germany in 1935. But it was not until after the end of World War II in 1945 that tape recording started to become widely available as a way of storing and replaying audio material. The tape that was used consisted of a thin acetate plastic tape coated with iron oxide, and polyester film still forms the backing of magnetic tape. More details of the technique of magnetic recording are given in Chapter 4.

Before recording, all music was live!

Tape recorders are very useful for synthesizing sounds because they allow permanent records to be kept of a performance, or they allow a performance to be 'time-shifted': recorded for subsequent playback later. This may appear to be obvious to the modern reader, but before tape recording, the only way to record sound for later playback was literally to make a record! It is not feasible to break up and reassemble records (see Section 1.4.6), and so when it was first introduced, the tape recorder was a genuinely new and exciting musical tool.

Pitch and speed

A tape recording of a sound ties together two main aspects of sound: the pitch and the duration. Because it captures the waveform in a physical form (on tape), it also ties things such as the distinctive tonal characteristics (formants) of any sounds to the speed of playback. So, for example, a recording of a reverberant room will sound much larger when you slow down the speed of playback tape. Sounds with strong formants will be significantly affected by changing the speed of playback: a triangle only sounds correct at the original speed.

If you record the sound at 15 inches per second (ips), then playing back at twice the speed, 30 ips, will double the pitch and so it will be transposed up by one octave. But the duration will be halved, and so a 1-second sound will only last for half a second when played back at twice the recording speed. This means that the decay on a plucked sound will happen twice as quickly as normal, which may sound correct in some contexts, but wrong in others. Breaking this interdependence is not at all easy using tape recorder technology, although it is relatively simple using digital processing techniques.

'Accuracy' and 'Precision' are often used as synonyms, but have distinct meanings. 'Accuracy' refers to repeatability and consistency, whereas 'precision' refers to the detail of one instance. So a person who can play a note every quarter of a second repeatedly would be accurate, whereas a precise measurement might show that the timing interval between two specific notes was exactly 0.250,000 of a second.

Being able to change the pitch of sounds once they are recorded can simplify the process of producing electronic sounds using oscillators. By changing the speed at which the sound is recorded, the same oscillators can be used to produce tape segments, which contain the same sound, but shifted in pitch by one or more octaves. This avoids some of the problems of continuously retuning oscillators, although it does depend on the tape speeds of the tape recorders being accurate.

Unfortunately, the tape speed of early tape recorders was not very accurate. Long-term drift of the speed affects the pitch of anything recorded, and so required careful monitoring. Short-term variations in tape speed are called wow and flutter. Wow implies a slow cyclic variation in pitch, whereas flutter implies a faster and more irregular variation in pitch. Depending on the type of sound, the ear can be very sensitive to pitch changes. Wow and flutter can be very obvious in solo piano playing, although some orchestral or vocal music can actually sound better!

Of course, changing the tape speed can be used as a creative tool: adjusting the tape speed whilst recording will permanently store the pitch changes in the recording, whereas changing the replay tape speed will only affect playback (but the speed changes are probably not as easy to reproduce on demand). Deliberately introducing wow and flutter can also be used to introduce vibrato and other pitch-shifting effects.

Splicing

Once a sound has been recorded onto tape, it is then in a physical form which can be manipulated in ways which would be difficult or impossible for the actual sound itself. Cutting the tape into sections and then splicing them together allows the joining, insertion, and juxtaposition of sounds. The main limits to this technique are the accuracy of finding the right place on the tape, and the length of the shortest section of tape that can be spliced together. Each joint in a piece of tape produces a potential weak spot, and so an edited tape may need to be recorded onto another tape recorder. Every time a tape is copied, the quality is degraded slightly, and so there is a need to compromise between the complexity of the editing and the fidelity of the final sound.

Reversing

Reversing the direction of playback of tape makes the sound play backwards. Unfortunately, because domestic tape recorders are designed to record in stereo on two sides (known as quarter-track format) merely turning the tape around does not work. Playing the back of the tape (the side which has the backing, rather than the oxide visible) does allow reversing of quarter-track tapes, but there is significant loss of audio quality. Professional tape recorders use the mono full-track and stereo half-track formats, where the tape direction is unidirectional, and these can be used to produce reversed audio (although the channels are swapped on a half-track tape!).

Playing sounds backwards has two main audible effects:

1. Most naturally occurring musical instrument sounds have a sharp attack and a slow release or decay time, and this is reversed. This produces a characteristic 'rushing' sound or 'squashed' feel, since the main rhythmic information is on the beats and these are now at the end of the notes.
2. Any reverberation becomes part of the start of the sound, whilst the end of the sound is 'dry'. Echoes precede the notes which produced them. Both of these serve to reinforce the crescendo effect of the notes.

Tape loops

Splicing the end of a section of tape back onto the start produces a loop of tape, and the sound will thus play back continuously. This can be used to produce repeated phrases, patterns and rhythms. Several tape loops of different lengths played back simultaneously can produce complex polyrhythmic sequences of sounds.

Sound on sound

Normally, a tape recorder will erase any pre-existing magnetic fields on the tape before recording onto it using its erase head. By turning this off, any new audio that is recorded will be mixed with the already-existing audio. This is called 'sound on sound' because it literally allows sounds to be layered on top of each other. As with many tape manipulation techniques, there is a loss in the quality each time this technique is used, specifically for the pre-existing audio in this case.

Delays, echoes and reverberation

By using two tape recorders, where one records audio onto the tape, and the second plays back the same tape, it is possible to produce time delays. The time delay can be controlled by altering the tape speed (which should be the same on both tape recorders) and the physical separation of the two recorders. Some tape recorders have additional playback heads, and these can be used to provide short time delays. Dedicated machines with one record head and several playback heads have been used to produce artificial echoes for example, the Watkins CopyCat and Roland Space Echo. The use of echo and time delays has been part of the performance technique of many performers, from guitarists to synthesists, as well as bathroom vocalists.

By taking the time-delayed signal and mixing it into the recorded signal, it is possible to produce multiple echoes from only one playback head (or the playback tape recorder). With multiple playback heads spaced irregularly, it is possible to use this feedback to partially simulate reverberation. With too much feedback, the system may break into oscillation, and this can be used as an additional method of synthesizing sounds, where a stimulus signal is used to initiate the oscillation.

Multi-tracking

Although early tape recorders had only one or two tracks of audio, experiments were carried out on producing tape recorders with more tracks. Linking two tape recorders together to give additional tracks was very awkward. Quarter-track tape recorders used four tracks, although usually only two of these could be played at once. Modified heads produced tape recorders with four separate tracks, and these 'multi-track' tape recorders were used to produce recordings where each of the tracks was recorded at a different time, with the complete performance only being heard when all four tracks were replayed simultaneously. This allowed the production of complete pieces of complex music using just one performer. Eight-track recorders followed, then 16-track machines, then 24, and additional tracks could be added by synchronizing two or more machines together to produce 48 and even 96-track tape recorders.

1.4.2 Found sounds

Found sounds are ones which are not pre-prepared. They are literally recorded as they are 'found' in situ. Trains, cars, animals, factories and many other locations can be used as sources of found sounds. The term prepared sound is used for sounds which are specially set up, initiated and then recorded, rather than spontaneously occurring.

1.4.3 Collages

Just as with paper collages, multiple sounds can be combined to produce a composite sound. Loops can be very useful in providing a rhythmic basis, whereas found sounds, transposed sounds and reversed sounds can be used to add additional timbres and interest.

1.4.4 Musique concrète

Musique concrète is a French word that has come to be used as a description of music produced from ordinary sounds which are modified using the tape techniques described earlier. Pierre Schaeffer coined the term in 1948 as music made 'from … existing sonic fragments'.

1.4.5 Optical methods

Although magnetic tape provides a versatile method of recording and reproducing sounds using a physical medium, it is not the only way. The optical technique used for the soundtrack on film projectors has also been used. The sound is produced by controlling the amount of light that through the film to a detector. Conventional film uses a 'slot', which varies in width, although it is also possible to vary in transparency or opacity of the film. Optical systems suffer from problems of dynamic range, and physical degradation due to scratches, dust and other foreign objects.

Chapters 2 and 4 deal with optical techniques in more depth.

1.4.6 Disk manipulation

Before the wax cylinder, shellac disk, vinyl record or the tape recorder, all music happened live. Although the tape recorder was the obvious tool for manipulating music, it was not the only one. Records with more than one lead-in groove, leading to several different versions of the same audio being selected at random, have been used for diversions such as horse racing games, and in the 1970s, a 'Monty Python' LP was deliberately crafted with a looped section which played: 'Sorry, squire: I've scratched your record!' continuously until the 'stylus' was lifted from the record.

Disk manipulation was overlooked for many years because vinyl discs were perceived as playback-only devices. It is quite possible to produce many of the effects of tape using a turntable or disk-cutting lathe. For example:

- Large pitch changes can be produced by using turntables with large speed ranges.
- Some pickups can be used in reverse play to reverse the play back of sounds.
- Multiple pickups can be arranged on a disk so that echo effects can be produced.
- 'Scratching' involves using a turntable with a slipmat under the disk, a bidirectional pickup cartridge and considerable improvisational and cueing skill from the operator, who controls the playing of fragments of music from standard (or custom-cut) LP discs by playing them forwards and backwards, repeating phrases and mixing between two (or more) discs at once.

The live user manipulation of vinyl discs has become so successful that interfaces that attempt to emulate the same twin disk format have been produced for use with CDs. Software emulations of the technique are also available on a number of computer platforms for use with MP3 files or other digital audio files.

1.4.7 Digital tape recorders

The basic tape recorder is an analogue device: the audio signal is converted directly into the magnetic field and stored on the tape. As with many analogue devices, the last 20 years of the twentieth century have seen a gradual replacement of analogue techniques with digital technology, and this has also happened with the tape recorder. In the case of the tape recorder, much of the tape handling remained the same, although open reels of tape were gradually replaced with enclosed designs of cassettes, notably in the domestic environment with the 'compact cassette', which was the MP3 of the second half of the twentieth century. In terms of tape manipulation techniques, the cassette made access to the tape difficult, although for the ordinary domestic user of cassettes, it definitely made tape recording more convenient. And for playback, Sony's Walkman personal portable cassette player was the equivalent of a modern Apple iPod MP3 player.

The method of recording audio in a digital tape recorder involves converting the audio signal into a digital representation for subsequent storage on the tape. The data formats used to store the information on the tape are normally not designed to be edited physically, although there have been some attempts to produce formats that can be cut and spliced conventionally. In general, editing is done digitally rather than physically in a digital tape recorder, and so their creative uses are limited to storing the output of a performance or a session, rather than being a mechanism for manipulating sound.

One of the early formats for digital audio tape (DAT) recording, the Sony F1 system, used Betamax video tape cassettes as the storage medium, and although intended for domestic and semi-pro usage, it was rapidly adopted by the professional music business in the 1980s. It was followed by DAT, which was not successful as a domestic format, where the compact cassette, then the MiniDisc, and later, the CD-recordable (CD-R) optical disks have dominated in turn, but DAT was widely accepted by the professional music industry in the 1990s.

Hard disk recording replaces the tape storage with a hard disk drive, although these are normally backed up to a tape backup device or an optical drive like one of the variations on recordable digital versatile disk (DVD) or Blu-Ray (BD) technology. The early twenty-first century has seen the rise of flash chip-based memory as a replacement for hard drives, with a rapid drop in cost and equally fast rise in capacity, and so the term 'hard disk' recording may not survive for much longer.

The cost of hard disk capacity (and now flash memory, also known as 'flash drives') appears to follow a permanently descending curve. Hard disks and flash memory also have rapid access time. Tape-based or optical storage is cheaper per byte, but has slower access time.

1.5 Experimental versus popular musical uses of synthesis

There is a broad spectrum of possible applications for synthesis. At one extreme is the experimental research into the nature of sound, timbre and synthesis itself, whereas at the opposite extreme is the use of synthesizers in making popular music. In between these two, there is a huge scope for using synthesis as a useful and creative tool.

1.5.1 Research

Research into music, sound and acoustics is a huge field. Ongoing research is being carried out into a wide range of topics. For example some of these include the following:

- alternative scalings
- alternative timbres
- processing of sounds
- rhythm, beats, timbre, scales, etc.
- understanding of how instruments work.

Much of this work involves multi-disciplinary research. For example, trying to work out how instruments work can require knowledge of physics, music, acoustics, electronics, computing and more. Some of the results of this research work can find application in commercial products: Yamaha's DX series of FM synthesizers and modeling-based software synthesis are just two of the many examples of the conversion of academic theory into practical reality. This is covered in more detail in Section 1.7.

1.5.2 Music

Music encompasses a huge variety of styles, sounds, rhythms and techniques. Some of the types of music in which a strong synthesized content may be found are as follows:

- *Pop music*: Popular music has some marked preferences – it frequently uses a 4/4 time signature, and preferentially uses a strongly clichéd set of timbres and song forms (especially verse/chorus structures, and key changes to mark the end of a song). It often has a strong rhythmic element, which reflects one of its purposes: music to dance. Pop music is also designed to be sung, with the vocal or instrumental hook often being a key part of the production effort.
- *Dance music*: Dance music has one purpose – music to dance to. Simplicity and repetition are therefore key elements. There are a large and evolving number of variants to describe the specific sub-genre: acid, melodic trance; house; drum and bass; jungle; garage; but the basic formula is one of a continuous 4/4 time signature with a solid bass and rhythm. Much dance music is remixed versions of pop and other types of music, or even remixed dance music.
- *New Age music*: New Age music mixes both natural and synthetic instruments into a form which concentrates on slower tempos than most popular music, and is more concerned with atmosphere.
- *Classical music*: Although much of classical music uses a standard palette of timbres, which can be readily produced by an orchestra, the augmentation by synthesizers is known in some genres (particularly music intended for film, television and other media purposes).
- *Musique concrète*: Although musique concrète uses natural sounds as the source of its raw material, the techniques that it uses to modify those sounds are often the same as those used by synthesizers.
- *Electronic music*: Electronic music need not be produced by synthesizers, although this is often assumed to be the case. As with popular music, a number of clichés are commonly found: the 8- or 16-beat sequence and the resonant filter sweep are two examples from the 1970s.

Crossovers There are some occasions when the boundaries between experimental uses of synthesizers crossover into more popular music areas, and vice versa. The use of synthesizers in orchestras typically occurs when conventional instrumentation is not suitable, or when a specific rare instrument cannot be hired. Music, which is produced for use in many areas, often requires to have elements of orchestral and non-orchestral instrumentation – adding synthesizer parts can enhance and extend the timbres available to the composer or arranger, and it avoids any need for the synthesizer to attempt to emulate a real orchestra. The use of orchestral scores for both movies and video games has produced a mass-market outlet for orchestration which is often augmented with synthesized instrumentation. Conversely, the use of orchestral instruments in experimental works also happens.

1.6 Electro-acoustic music

The study of the conversions between electrical energy and acoustic energy is called electro-acoustics. Unlike previous centuries, where the development of mechanical-based musical instruments had dominated the study of musical acoustics, in the twentieth century, innovation has largely concentrated on instruments that are electronic in nature. It is thus logical that the term electro-acoustics should also be used by musicians to describe music that is made using electronic musical instruments and other electronic techniques.

Unfortunately, the term 'electro-acoustic music' is not always used consistently, and it can also apply to music where acoustic instruments are amplified electronically. The term 'electronic music' implies a completely electronic method of generating the sound, and thus represents a very different way of making music. In practice, both terms are now widely used to mean music that utilizes electronics as an integral part of the creative process, and thus it covers such diverse areas as amplified acoustic instruments (where the instruments are not merely made louder), music created by synthesizers and computers, and popular music from a wide range of genres (pop, dance, techno, etc.). Even classical music performed by an orchestra, but with an additional electronic instrumentation, or even post-processing of the recorded orchestral sound, could be considered to be 'electronic'.

1.6.1 Electro-acoustics

Electro-acoustics is a science tempered by human interaction and art. In fact, the close linking between the human being and most musical instruments, as well as the space in which they operate, can be a very emotional one. The electronic nature of many synthesizers does not fundamentally alter this relationship between human and instrument, although the details of the interface are still very clumsy. As synthesizers develop, they would gradually become performer oriented, and less technological, which should make their electro-acoustic nature less and less important. Many conventional instruments have histories of many hundreds of years, whereas electro-acoustic music is less than a century old, and synthesizers are less than 50 years old. Electro-acoustics is comparatively new.

1.7 The 'Produce, Mix, Record, Reproduce' sound cycle

Making musical sounds requires a complete process in order to successfully transfer from the performer to the end consumer. Understanding the detail of this process will clarify the way in which the performance environment has evolved over time.

In a live performance, the process seems to be very straightforward. The performer makes the sounds, and those sounds, plus perhaps the visual experience of the performer making them, are seen by the listeners or viewers. The only obvious element of the performance that is not apparent to the listener or viewer is the time taken by the performer to prepare, and it might have taken a considerable effort to learn the instrument or the piece of music. In the case of multiple performers, each individual performer in an orchestra, band, choir or other gathering of performers, may have prepared individually as well as a group.

But there are a number of other processes that have led to the performance being possible. The sounds produced by a performer are based on either the memory of what those sounds are, or else the conversion of printed musical symbols from a score into those sounds. The score itself is the result of a composer putting together sounds to achieve an effect, and capturing the instructions in a written form. The sounds that a performer makes may be based on the score, but the timbre and performance details may be based on a long-term education gained from many other performers and teachers. The composer must have also spent time learning about sounds and how they can be used.

When a performance is captured by a recording device, the process has all the live elements, but more are added because the recording can subsequently be reproduced. Recording devices such as scores, tape recorders, or electronic captures of physical performance all convert sound into a stored form, and video camcorders can record the visual part of a performance too. Playing back the stored performance can be done immediately after it has been stored, or it can be in a different place, or a different time

Making sounds is thus far from a straightforward process. There are three stages to the process, although they actually loop around in a cycle, and the cycle may be repeated several times in order to complete the transfer from the creator of the sounds to the end consumer of the performance.

The stages are as follows:

- Produce, the making of sound
- Mix, the combining or alteration of sound
- Record, the storing of sound
- Reproduce, which is the 'produce' start of another cycle.

Understanding that this cycle is present is important when considering the way that technology has integrated cycles into devices and made them largely invisible. Whilst the score and practice elements of a performance seem obvious when explained in the context of forming the essential preparation of a live performance, when a synthesizer is used to replay a pre-recorded sequence of sounds, or an MP3 player uses a playlist to replay a sequence of songs made up of pre-recorded sounds, the details of the cycles may well be hidden. Listeners to music produced by a Trautonium may not have known the mechanism which was used to produce those sounds, and when a computer makes sounds, then just about all of the many cycles used are not apparent at all.

This book is aimed at making the 'produce, mix, record, reproduce' cycle not only visible, but understandable. Knowing how sounds are made is just one part of a complex set of nested cycles, and understanding this can be a valuable tool to making the most of devices, performers, and their processes.

1.8 From academic research to commercial production …

Synthesizers can be thought of as coming in two forms: academic and commercial. Academic research produces prototypes which are typically innovative, fragile and relatively extravagant in their use of resources. Commercial synthesizers are often cynically viewed as being almost the exact opposite: minor variations on existing technology that are often renamed to make them sound new and different; and very careful to maximize the use of available resources. Previous editions of this book also added 'robust, perhaps even over-engineered' to this list, but the dependence on software in many modern synthesizers has reduced their robustness, commercial pressure has reduced any over-engineering, and there is now an increasing dependency on ongoing updates or 'continuous beta' approaches to product support. Production development of research prototypes is often required to enable successful exploitation in the marketplace, although this is a difficult and exacting process, and there have been both successes and failures.

In order to be a success, there are a number of criteria that need to be met. Moving from a prototype to a product can require a complex exchange of information from the inventor to the manufacturer, and may often need an additional development work. Custom chips or software may need to be produced, and this can introduce long delays into the time-scales, as well as a difficult testing requirement. Management tasks such as organizing contracts, temporary secondment of personnel, patents and licensing issues, all need to be monitored and controlled.

Robert Moog and Bob Oberheim both used their names for companies and products, and then left that company. Dave Smith has ended up working in a company with his name.

Even when the product has been produced, it needs to be promoted and marketed. This requires a different set of skills, and in fact, many successful companies split their operations into 'research and development' and 'sales and marketing' parts. The synthesizer business has seen many companies with ability in one of these fields, but the failures have often been a result of a weakness in the other field. Success depends on talent in both areas, and the interchange of information between them. Very few of the companies who started out in the 1960s and 1970s are still active; however, the creative driving force behind these companies, which is frequently only one person, is often still working in the field, albeit sometimes in a different company.

Apart from the development issues, the other main difference between academic research and commercial synthesizer products is the motivation behind them. Academic research is aimed at exploring and expanding of knowledge, whereas commercial manufacturers are more concerned with selling products. Unfortunately, this often means that products need to have a wide appeal, simple user interfaces, and easy application in the popular music industry. The main end market for electronic musical instruments is where they are used to make the music that is heard on television, radio, films, DVDs and CDs and the development process is aimed at this area.

What follows are some brief notes on some of these 'developed' products.

1.8.1 Analogue modular

Analogue synthesizers were initially modular, and were probably aimed at academic and educational users. The market for 'popular' music users literally did not exist at the time. The design and approach used by early modular synthesizers was similar to those of analogue computers. Analogue computers were used in academic, military and commercial research institutions for much the same types of calculations that are now carried out by digital computers.

Pioneering work by electronic music composers using early modular synthesizers was more or less ignored by the media until Walter Carlos released some recordings of classical Bach by using a Moog modular synthesizer. The subsequent release of this material as the *Switched On Bach* album quickly became a major success with the public, and the album became one of the best-selling classical music records ever. This success led to enquiries from the popular music business, with the Beatles and the Rolling Stones being early purchasers of Moog modular synthesizers.

'Moog' is pronounced to rhyme with 'vogue'.

By the beginning of the twenty-first century, software synthesis allowed the creation of emulated analogue modular synthesizers (and other electronic music instruments) on general-purpose computers. The comparatively low cost of the software means that musicians who could never afford a real modular synthesizer are able to explore sound-making, whilst the use of software means that patches can easily be stored and recalled: a huge advantage over analogue modular synthesizers.

1.8.2 FM

FM as a means of producing audio sounds was first comprehensively described by John Chowning, in a paper entitled. *The Synthesis of Complex Audio Spectra by Mean of Frequency Modulation*, published in 1973. At the time, the only way that this type of FM could be realized was by using digital computers, which were expensive and not widely available to the general public.

As digital technology advanced, some synthesizer manufacturers began to look into ways of producing sounds digitally, and Yamaha bought the rights to use Chowning's 1977-patented FM ideas. Early prototypes used large numbers of simple transistor–transistor logic (TTL) chips, but these were quickly replaced by custom-designed chips which compressed these onto just a few more complex chips. The first functional all-digital FM synthesizer designed for consumer use was the Yamaha GS1, which was a pathfinder product designed to show expertise and competence, as well as test the market. Simple preset machines designed for the home market followed. Although the implementation of FM was very simple, the response from musicians and players was very favorable.

Perhaps as a consequence of the DX7, other synthesizers in the 1980s (and beyond) tended to concentrate on a similar price point and a two-model strategy: the basic 'mass market' model and a more expensive 'pro' model, often with more notes on the keyboard (or a weighted keyboard), and sometimes with extended functionality.

The DX1, DX7 and DX9 were released in late 1982, with the DX1 apparently intended as the professional player's instrument, the DX7 a mid-range, cut-down DX1, and the DX9 as the low-cost, large-volume 'best seller'. What actually happened is very interesting. The DX9 was so restricted in terms of functionality and sound that it did not sell at all, whereas the DX7 was hugely in demand amongst both professional and semiprofessional musicians, and the DX1 was interpreted as being a 'super' DX7 for a huge increase in price. Inevitably, it took Yamaha some time to increase the production of the DX7 to meet the demand, and this scarcity only served to make it all the more sought-after! By the time that the mark II DX7 was released, about a quarter of a million DX7s had been sold, which at the time was a record for a synthesizer.

The popularity of the DX7 was responsible for the release of the mark II instrument, which was a major redesign, not a new instrument, a very rare approach, and one which shows how important the DX7, and FM, had become. For several years, between 1983 and 1986, Yamaha and FM enjoyed a popularity that ushered in the transition from analogue to digital technology. It also began the trend away from user programming, and towards the selling of pre-prepared sounds or patches. The complexity of programming FM meant that many users did not want to learn, and so purchased sounds from specialist companies that marketed the results of a small number of 'expert' FM programmers.

In the late 1990s, Yamaha released a new FM synthesizer module, the FS1R, which extended and enhanced the FM synthesis technique of the previous generation, and this was accompanied by a resurgence of interest in FM as a 'retro' method of making sounds. Fashion in synthesis is cyclic. In 2001, a software synthesizer version of the original DX series FM synthesizers was released, and DX-style FM joined the sonic palette of commercial software synthesis.

1.8.3 Sampling

Sampling is a musical reuse of technology which was originally developed for telephony applications. The principles behind the technique were worked out in the twentieth century, but it was not until the invention of the transistor in the 1950s that it became practical to convert continuous audio signals into discrete digital samples using PCM.

Commercial exploitation of sampling began with the Fairlight Computer Musical Instrument (CMI) in 1979, although this began as a wavetable synthesizer as the size of the wavetables increased, it rapidly evolved into an expensive and fashionable professional sampling instrument, initially with only 8-bit sample resolution. Another 8-bit instrument the Ensoniq Mirage was the first instrument to make sampling affordable. E-mu released the Emulator in 1979, drum machines such as the LinnDrum were released in 1979, and sampling even began to appear on low-cost 'fun' keyboards designed for consumer use at home, during the mid-1980s.

The 8-bit resolution was replaced by 12 bits in the late 1980s, and 16 bits became widely adopted in the early 1990s. Before the end of the twentieth century the CD standards of 16-bit resolution and 44.1 kHz had become widely adopted for samplers, with lower sampling rates only being used because of memory constraints. The twenty-first century has seen a wide adoption of software sample playback as an alternative to hardware: either as plug-ins to software MIDI and audio sequencers, or as stand-alone 'sample' sequencers.

CD-R read-only-memory compact discs (CD-ROMs) of pre-prepared samples have replaced do it yourself (DIY) sampling for the vast majority of users. Samplers have mostly become replay-only devices, with only a few creative individuals and companies producing samples on CD-ROMs, and many musicians using them. The use of music CDs as source material has become formalized, with royalty payments on this usage being 'business as usual' for many often-used artists of previous generations.

The wide availability of pre prepared sounds for samplers is analogous to the patches that are available for software-based analogue modular synthesizers (see Section 1.8.1). In both cases, most users make only minor changes to the sounds which have been created by a few highly skilled individuals.

1.8.4 Modeling

Mathematical techniques such as physical modeling seem to have made the transition from research to product in a number of parallel paths. There have been several speech coding schemes based on modeling the way that the human voice works, but these have been restricted to mainly telecommunications and military applications; only a few of these have found musical uses (see Chapter 5). Research results that have been reporting the gradual refinement of modeling techniques for musical instruments have been released, notably by Julius Smith (Julius O. Smith III), and commercial devices based on these began to appear in the mid-1990s. Yamaha's VL1 was the first major commercial synthesizer to use physical modeling based on blown or bowed tubes and strings, and many other manufacturers have followed, including many emulations of analogue synthesizers.

The development of electronic musical instruments is still continuing. The role of research is as strong as ever, although the pace of development is accelerating. Digital technology is driving synthesis towards general-purpose computing engines with customized audio output chips, and this means that the software is increasingly responsible for the operation and facilities that are offered, not the hardware. By the turn of the century, many companies had products that used general-purpose DSPs to synthesize their sounds, and had produced products that used mixtures of synthesis technologies to produce those sounds: FM, additive, emulations of analogue synthesis, physical modeling and more. But despite the flexibility and power of these systems, the popular choice has continued to be a combination of sample replay and synthesis; and perhaps the simplicity and familiarity of the metaphor used is a key part of this.

The twenty-first century has seen modeling technology become a standard tool to produce digital versions of both analogue electronic and natural instruments. Fast powerful processors have also greatly reduced dependence of the DSP as the processing engine, and opened up the desktop or laptop computer as a means of synthesizing using modeling techniques. But the complex metaphor and interface demands of physical modeling and other advanced techniques have seen them pushed into niche roles, with only the analogue emulations enjoying wide commercial success.

1.9 Synthesis in context

One of the major forces, which popularised the first use of synthesizers in popular music, was using synthesizers to produce recorded performances of classical music. Because these could be assembled onto tape with great precision, the timing control and pitch accuracy, which were used, were on a par with the best of human realizations, and so the results could be described as 'virtuoso' performances. In the 1950s, this suited the mood of the time, and so a large number of electronically produced versions of popular classical music were produced. This has continued to the present day, although it has become increasingly rare and uncommercial; perhaps the wide range of musical genres that co-exist in the 2000s means that it is no longer seen as relevant.

There is no 'correct' way to use synthesizers to create music, although there are a number of distinct 'styles'. Individual synthesists have their own preferences, although some have less fixed boundaries than others, and can move from one style to another within a piece of music. I do not know of any formal means of categorizing such styles, and thus propose the following divisions:

- *Imitative*: Imitative synthesis attempts to use electronic means to realize a performance which is as close as possible to a recording of a conventional orchestra, band or group of musicians. The timbres and control techniques, which are used, are intended to mimic the real-world sounds and limitations of the instrumentation. Many film soundtracks fall into this category.

- *Suggestive*: This style does not necessarily use imitative instrumental sounds, but rather, aims to produce an overall end result, which is still suggestive of a conventional performance.

- *Sympathetic*: Although using instrumental sounds and timbres which may be well removed from those used in a normal performance, a 'sympathetic' realization of a piece of music aims to choose sounds which are in keeping with some elements of the conventional performance.

- *Synthetic*: Electronic music aims to free the performer from the constraints of conventional instrumentation, and so this category includes music where there is little that would be familiar in tone or rhythm to a casual listener.

1.9.1 'Synthetic' versus 'real'

Sound synthesis does not exist in isolation. It is one of the many methods of producing sound and music. There are a large number of non-synthetic, non-electronic methods of producing musical sounds. All musical instruments synthesize sounds, although most people would probably use the word 'make' rather than 'synthesize' in this context. In fact, the word synthesize has come to mean something which is unnatural; synthetic implies something that is similar to, but inferior to 'the real thing'. The ultimate example of this view is the sound synthesizer, which is often described as being capable of emulating any type of sound, but with the proviso that the emulation is usually not perfect.

As with anything new or different, there is a certain amount of prejudice against the use of electronic musical instruments in some con texts. This is often expressed in words such as: 'What is wrong with real instruments?', and is frequently used to advocate the use of orchestral instruments rather than an electronic realization using synthesizers.

There are two major elements to this prejudice: unfamiliarity and fear of technology. Many people are very much used to the sounds and timbres of conventional instrumentation, especially the orchestra. In contrast, the wider palette of synthetic sound is probably very unfamiliar to many casual listeners. Thus an unsympathetic rendering of a pseudo-classical piece of music, produced using sounds which are harsh, unsubtle, and obviously synthetic in origin, is almost certain to elicit an unfavorable response in many listeners. In contrast, careful use of synthesis can result in musical performances, which are acceptable even to a critical listener, especially if no clues are given as to the synthetic origins.

The technological aspect is more complex. Although the piano-forté was once considered too new and innovative to be considered for serious musical uses, it has now become accepted by familiarity. The concept of assembling together a large number of musicians into an orchestra was a more gradual process, but the same transition from 'new' to 'accepted' still occurred. It seems that there may be an inbuilt 'fear' of anything new or which is not understood. This extends far beyond the synthesizer: computers and most other technological inventions can suffer from the same aversion.

Attempting to draw a line between what is acceptable technologically and what is not can be very difficult. It also changes with time. Arguably, the only 'natural' musical instrument is the human voice, and anything that produces sounds by any other method is inherently 'synthetic'. This includes all musical instruments from simple tubes through to complex computer-based synthesizers. There seems to be a gradual acceptance of technological innovation over time, which results in the current wide acceptability of musical instruments which may well have been invented more than 500 years ago.

1.9.2 Film scoring

Film scoring is an excellent example of the way sound synthesis has become integrated into conventional music production. The soundtracks of many films are a complex mixture of conventional orchestration combined with synthesis, but a large number of films have soundtracks that have been produced entirely electronically, with no actual orchestral content at all, although the result sounds like a performance by real performers. In some cases, although the music may sound 'realistic' to a casual listener, the performance techniques may be well beyond the capability of human performers, and some of the timbres used can be outside of the repertoire of an orchestra.

There are advantages and disadvantages to the 'all-synthetic' approach. The performer who creates the music synthetically has complete control over all aspects of the final music, which means that changes to the score can be made very rapidly, and this flexibility suits the restraints and demands which can result from film production schedules. But giving the music a human 'feel' can be more difficult, and arguably more time consuming, than asking an orchestra to interpret the music in a slightly different way. The electronic equivalent of the conductor is still some way in the future, although there is considerable academic research into this aspect of controlling music synthesizers. One notable example that illustrates some of the possibilities are the violin bows which have been fitted with accelerometers by Todd Machover's team at the MIT Media Lab in Boston, USA (United States of America), which can measure movement in three dimensions and which are not dissimilar to the sort of measurements which would be required for a conductor's baton.

Mixing conventional instrumentation with synthesizers is also used for a great deal of recorded music. This has the advantage that the orchestral instruments can be used to provide a basic sound, and additional timbres can be added into this to add atmosphere or evoke a specific feel. Many of the sounds used in this context are clichés of the particular time when the music was recorded. For example, soundtracks from the late 1970s often have a characteristic 'drum synthesizer' sound with a marked pitch sweep downwards – the height of fashion at the time, but quaint and hackneyed to people 10 or 20 years afterwards. But recycling of sounds (and tunes) does occur, and 'retro' fashionability is always ready to rediscover and reuse yesterday's clichés.

1.9.3 Sound effects

The real world is noisy, but the noises are often unwanted. For film and television work, background noise, wind noise and other extraneous sounds often mean that it is impossible to record the actual sound whilst recording the pictures, and so sound effects need to be added later. Everyday sounds such as doors opening, shoes crunching on gravel paths, switches being turned on or off, cans of carbonated drink being opened and more are often required. Producing these sounds can be a complex and difficult process, especially since many sounds are very difficult to produce convincingly.

Years of exposure to film and television have produced a set of clichéd sounds which are often very different from reality. For example, does a real computer produce the typical busy whirring and bleeping sounds that are often used for anything in the context of computers or electronics? Sliding doors on spaceships always seem to open and close with a whoosh of air, which would seem to suggest a serious design fault. The guns in Western films suffer from a large number of ricochets and fight scenes often contain the noises of large numbers of bones being broken, although the combatants seem relatively uninjured.

Many of the sounds that you hear on film or television are dubbed on afterwards. Some of these are 'synthesized' live by humans using props on a 'Foley' stage, although often the prop is not what you might expect: rain can be emulated by dropping rice onto a piece of cardboard, for example. But many sounds are produced synthetically, especially when the real-world sound does not match expectations. An example is the noise made when a piece of electronic equipment fails catastrophically: often nothing is heard apart from a slight clunk or lack of hum, which is completely unsuitable for dramatic use. A loud and spectacular sound is needed to accompany the unrealistic shower of sparks and smoke which billow from the equipment.

This use of sounds to enhance the real world can also be used to extremes, especially in comedy. A commonly used set of 'cod' or comic sounds has become as much a part of the film or television medium as the 'fade to black'. Laboratory equipment blips and bloops, and elastic bands twang in an unrealistic but amusing way – the exaggeration is the key to making the sound effect funny. Many of these sounds are produced using synthesizers, or a combination of prop and subsequent processing in a synthesizer. Samplers are often used in order to reproduce these sound effects 'on cue', and the user inter face to these can vary from a music keyboard to the large wooden pads connected to drum sensors that are used to add the fighting noises to Hong Kong 'Kung Fu' movies.

Given this mix of cliché, artificial recreation and exaggeration, it is not surprising that there is a wide variety of pre-prepared sound effect material in the form of sound effects libraries. As with all such 'canned' sample materials, the key to using it effectively is to become a 'chef' and to 'synthesize' something original using the library contents as raw materials. Discovering that the key 'weapon firing' sound is the same as the characteristic noise made by the lead robot in another television program can have a serious effect on reputations of all concerned.

As the DVD (where 'video' is often mistakenly taken to be the middle 'V' of the acronym, when it was originally stated to be 'versatile', although it is now said that DVD is not an acronym at all) became the fastest-selling consumer item ever in the early twenty-first century, so surround sound has become more widely used, initially in films, but musicians are always keen to exploit any new technology. In film and television usage, the front channels are typically used for the speech and the music, with the rear channels being used for sound effects, atmosphere and special effects. In music, several systems that used variations of dummy-head recordings were experimented with in the 1970s, 1980s and 1990s, and some commercial recordings were made using them and released on stereo CDs. But although these systems can enhance the stereo image by adding sounds which appear to be behind the listener, their limitations did mean that they tended to be used for special effects or for back ground ambience: a vocal performance that moved around your head, or raindrops surrounding you With no clear single contender for replacing stereo music on CD with a surround-based medium, exploiting the possibilities of surround music is not straightforward. Manufacturers are, of course, keen to see the replacement of recording equipment with surround-oriented new purchases.

1.9.4 Synthesis and making sounds

The rough timeline in Figure 1.9.1 shows the historical progress of music-making.

When synthesizers first became available as sound-making instruments, they were new and unusual, particularly in the sounds they made. The Moog 'bass' sound quickly became a cliche. As synthesizers developed, they were used by skilled musicians to replace some conventional instruments, particularly where the instruments being replaced were time consuming to record. Some types of brass and string backing sounds were particularly prone to this replacement technique.

The recording and reproducing of sound digitally has gone through several stages. MIDI was used as a digital alternative to multi-track tape recording for some musical arrangements using synthesizers, but it was not until digital samplers matured that full digital sampled arrangements became widely used. Samplers were initially used as replacements for conventional instruments such as pianos and strings, but over time, synthesizer sounds were sampled and put into sample ROMs and either S&S sample replayers or samplers became replacements for cliched synthesizer sounds too.

The timeline shows a gradual removal of the physical: for example, many computer software programs use the qwerty keyboard to enter notes into the internal sequencer, which produces constant velocity notes, with the dynamics left for the user to add in later, if at all required. This is also symptomatic of a gradual removal of the need for accurate performance: the computer can be used to correct notes, add velocity, after-touch, etc., after the user has entered

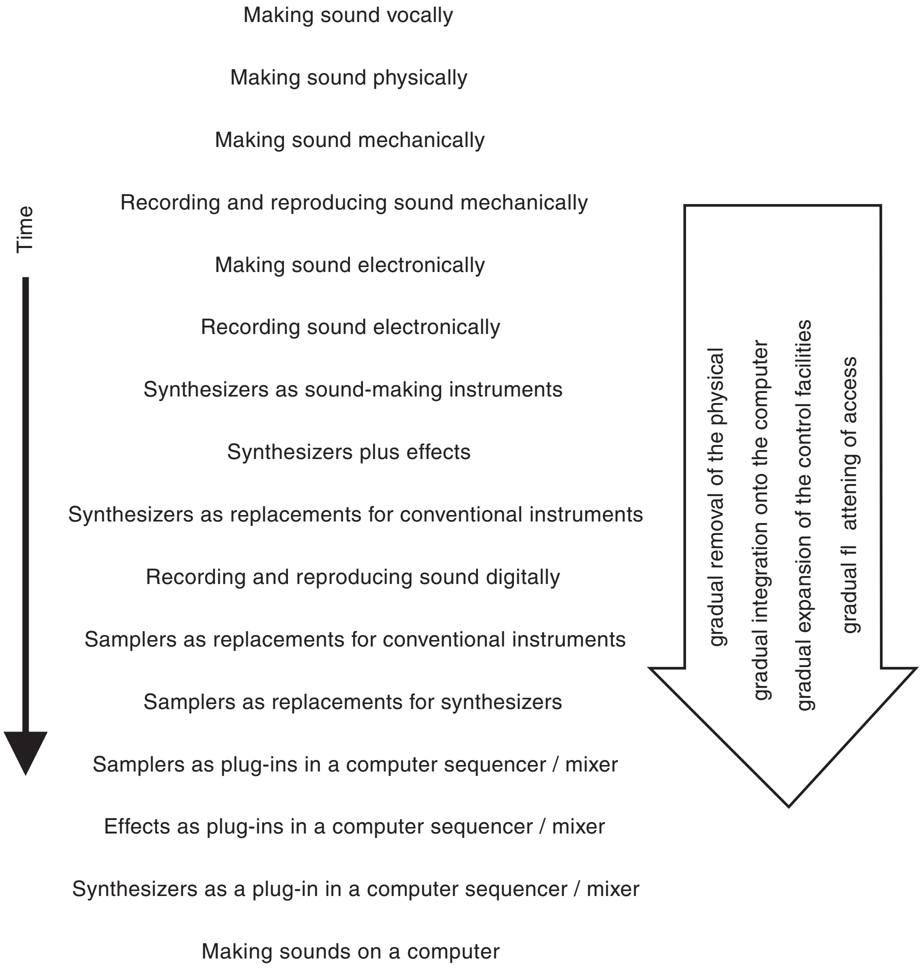

Making sound vocally
Making sound physically
Making sound mechanically
Recording and reproducing sound mechanically
Making sound electronically
Recording sound electronically
Synthesizers as sound-making instruments
Synthesizers plus effects
Synthesizers as replacements for conventional instruments
Recording and reproducing sound digitally
Samplers as replacements for conventional instruments
Samplers as replacements for synthesizers
Samplers as plug-ins in a computer sequencer / mixer
Effects as plug-ins in a computer sequencer / mixer
Synthesizers as a plug-in in a computer sequencer / mixer
Making sounds on a computer
Time
gradual removal of the physical
gradual integration onto the computer
gradual expansion of the control facilities
gradual flattening of access

FIGURE 1.9.1 Synthesis in the context of making sound – a rough timeline.

the music using the qwerty keyboard. There is also a flattening of access much of the marketing effort for computer software seems to suggest that anyone can now make a best-selling album by using the same tools as the professionals.

An alternative interpretation of these changes can be seen as being more positive. Removing the focus on the music keyboard and performance opens up music making to more people, whilst still enabling the capable performer to produce music quickly and efficiently. Although it sometimes may seem that musical ability is no longer essential, talent continues to shine through, and sophisticated sound-making is now easier than ever before, but only a few special people have the ability to explore the limits and still make music that connects at an emotional level. In a world where software is very accessible and affordable, the best way of rising above the crowd is to have the knowledge and ability to go beyond the basics and presets, to make the most of what is provided, to work around limitations, and to make music that connects with people.

Most intriguingly of all, the opening up of sound-making to people means that it is no longer necessary to spend lots of time learning about the interworking limitations of various pieces of equipment, or specific peculiarities, and so synthesis is increasingly a world of 'you can do that' instead of 'you can't do that'. The author started out in a world where arcane knowledge, hardware incompatibility and carefully guarded techniques were the norm, and wishes that he had a time machine to go back and reveal a different way.

1.10 Acoustics and electronics: fundamental principles

Knowledge of the electronic aspects of acoustics can be very useful when working with synthesizers, because synthesizers are just one of the many tools that can be used to assist in the creation of music. Thus this section provides some background information on acoustics and electronics. Because some of the terminologies used in this section use scientific unit symbols, some additional information on the use of units is also provided.

1.10.1 Acoustics

Acoustics is the science of sound. Sound is concerned with what happens when something vibrates. The vibration can be produced by vibrating vocal cords, wind whistling through a hole, a guitar string being plucked, a gong being struck, a loudspeaker being driven back and forth by an amplified signal, and more. Although most people think of sound as being carried only through the air, sound can also be transmitted through water, metals, wood, plastics and many other materials.

Although it is often easy to observe an object vibrating, sound waves are less tangible. The vibrations pass through air, but the actual process of pressure changes is hard to visualize. One effective analogy is the stretched spring, which is vibrated at one end – the actual compressions and rarefactions (the opposite of compression) of the 'waves' can then be seen traveling along the spring (Figure 1.10.1). Trying to amalgamate this idea of pressure waves on springs with the ripples spreading out on a pond is more difficult. As with light, the idea of spreading out from a source is hard to reconcile with what happens – people see and hear things, and waves and beams seem like very abstract notions. In real life, the only way to interact with sound waves is with your ears, or for very low frequencies, your body.

When an object vibrates, it moves between two limits; a vibrating string provides a good example, where the eye tends to see the limits (where the string is momentarily stopped whilst it changes direction) rather than where

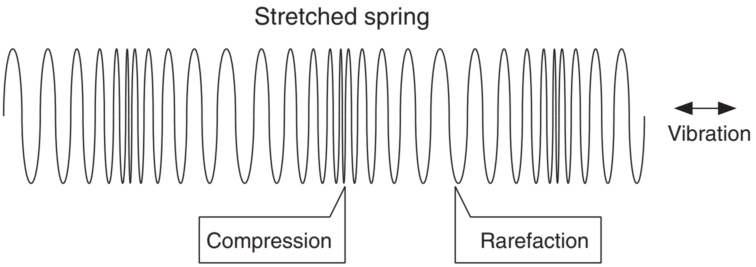


FIGURE 1.10.1 Pressure changes in the air can be thought of as being similar to a stretched spring which is vibrated at one end. The resulting pressure 'waves' can be seen traveling along the spring.

it is moving. This movement is coupled to the air (or another transmission medium) as pressure changes. The rate at which these pressure changes happen is called the frequency. The number of cycles of pressure change, which happen in 1 second is measured in a unit called hertz (Hz) (cycles per second is an alternative unit). The time for one complete cycle of pressure change is called the period, and is measured in seconds.

Pitch and frequency

Frequency is also related to musical pitch. In many cases, the two are synonymous, but there are some circumstances in which they are different. Frequency can be measured, whereas pitch can sometimes be subjective to the listener. In this book frequency will be used in a technical context, whereas pitch will be used when the subject is musical in nature.

The frequency usually used for the note A just above middle C is 440 Hz. There are local variations on this 'A-440 standard', but most electronic musical equipment can be tuned to compensate. Human hearing starts at about 20 Hz, although this depends on the loudness and listening conditions. Frequencies lower than this are called subsonic or, more rarely, infrasonic. For high frequencies human hearing varies with age and other physiological effects (such as damage caused by over exposure to very loud sounds or ear infections). For a normal teenager, frequencies of up to 18 kHz (18,000 Hz) can be heard; the 15,625 Hz line whistle from a 625-line Phase Alternation Line (PAL) television is an useful indicator. The ageing process means that an average middle-aged person will probably only be able to hear frequencies of perhaps 12 or 13 kHz. The 'hi-fi' range of 20 Hz to 20 kHz is thus well in excess of most listeners' ability to hear, although there is some debate about the ability of the ear to discern higher frequencies in the presence of other sounds (most hearing tests are made with isolated tones in quiet conditions).

Apparently 'middle C' is so called because it is written in the middle: between the bass and treble staves.

Notes

The fundamentals of most musical notes are in the lower part of 20 Hz to 20 kHz range. The fundamental is the name given to the lowest major frequency which is present in a sound. The fundamental is the pitch, which most people would whistle when attempting to reproduce a given note. Harmonics, overtones or partials are the names for any additional frequencies that are present in a sound. Harmonics are those frequencies that are integer multiples of the fundamental – they form a series called the harmonic series for that note. Overtones or partials are not related to the fundamental frequency. The upper part of the human hearing range contains these additional harmonics and partials. Table 1.10.1 shows the fundamental frequencies of the musical notes. An 88-note piano keyboard will span A0 to C8, with the top C having a frequency of just over 4 kHz.

The study of musical scales is a complex subject. For further information see Pierce, (1992).

Musical pitch is divided into octaves, and each octave represents a doubling of frequency. Thus A4 has a frequency of 440 Hz, whereas A5 has twice this frequency: 880 Hz. A3 has half the frequency: 220 Hz. Octaves are normally split into 12 parts, and the intervals are called semitones. The relationship between the individual semitones in an octave is called the scale. The table shows the equal tempered scale, where the intervals between the semitones are all the same: many other scalings are possible. Since there are 12 tones, and the frequency doubles in an octave interval, the semitone intervals in an equal tempered scale are each related by the 12th root of two, which is approximately 1.059,463. Semitones are split up into 100 cents, but most human beings can only detect changes in pitch of 5 cents or more. Cent intervals are related by the 1200th root of 2, which is approximately 1.00,057,779. As an example of what this represents in terms of frequency: for a A5 note of 880 Hz, a cent is just below 0.51 Hz, and thus 5 cents represent only 2.5 Hz!

Phase

When an object is vibrating, it repeatedly passes through the same position as it moves. A complete movement back and forth is called a cycle or an oscillation, which is why anything that produces a continuous vibration is called an oscillator. The particular point in a cycle at any instant is called the phase: the cycle is divided up into 360°, rather like a circle in geometry. Phase is thus measured in degrees, and zero is normally associated with either the start of the cycle, or where it crosses the resting position. The word 'zero crossing' is used to indicate when the position of the object passes through the rest position. A complete cycle conventionally starts at a zero crossing, passes through a second one, and then ends at the third zero crossing (Figure 1.10.2).

The change of position with time as an object vibrates is called the waveform. A simple oscillation will produce a sine wave, which looks like a smooth curve. More complex vibrations will produce more complex waveforms – a guitar string has a complex waveform because it produces a number of harmonics at once. If two identical waveforms are mixed together, the phase can determine what happens to the resulting waveform. If the two are in phase, that is, they both have the same position in the cycle at the same instant, then they

Table 1.10.1 Note frequencies in Hz

C	C+	D	D+	E	F	F+	G	G+	A	A+	B	Octave
16.35	17.32391444	18.354048	19.44543649	20.60172231	21.82676447	23.12465142	24.49971475	25.9565436	27.5	29.13523509	30.86770633	0
32.70319566	34.64782889	36.708096	38.89087298	41.20344463	43.65352894	46.24930285	48.9994295	51.9130872	55	58.27047019	61.73541265	1
65.40639131	69.29565778	73.41619201	77.78174596	82.40688925	87.30705788	92.49860569	97.99885901	103.8261744	110	116.5409404	123.4708253	2
130.8127826	138.5913156	146.832384	155.5634919	164.8137785	174.6141158	184.9972114	195.997718	207.6523488	220	233.0818807	246.9416506	3
261.6255653	277.1826311	293.664768	311.1269838	329.627557	349.2282315	369.9944228	391.995436	415.3046976	440	466.1637615	493.8833012	4
523.2511305	554.3652622	587.3295361	622.2539677	659.255114	698.456463	739.9888455	783.9908721	830.6093952	880	932.327523	987.7666024	5
1046.502261	1108.730524	1174.659072	1244.507935	1318.510228	1396.912926	1479.977691	1567.981744	1661.21879	1760	1864.655046	1975.533205	6
2093.004522	2217.461049	2349.318144	2489.015871	2637.020456	2793.825852	2959.955382	3135.963488	3322.437581	3520	3729.310092	3951.06641	7
4186.009044	4434.922098	4698.636289	4978.031741	5274.040912	5587.651704	5919.910764	6271.926976	6644.875162	7040	7458.620184	7902.132819	8
8372.018088	8869.844195	9397.272577	9956.063482	10548.08182	11175.30341	11839.82153	12543.85395	13289.75032	14080	14917.24037	15804.26564	9
16744.03618	17739.68839	18794.54515	19912.12696	21096.16365	22350.60682	23679.64306	25087.70791	26579.50065	28160	29834.48074	31608.53128	10
C	C+	D	D+	E	F	F+	G	G+	A	A+	B	Octave

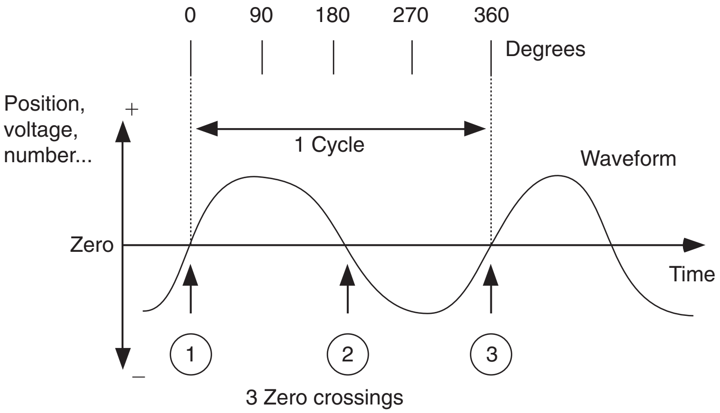


FIGURE 1.10.2 A complete cycle starts on the zero axis, crosses the zero axis and ends just as it is about to cross the zero axis for the second time. This can be simplified to '1 cycle = 3 zero crossings'.

will be added together; this is called 'constructive interference', because the two waveforms add together as they 'interfere' with each other. Conversely, if the two waveforms are 180°out of phase, then the phases will be equal and opposite, and the two waveforms will tend to cancel each other out this is called 'destructive interference'.

Beats

Slight differences of frequency between two waveforms can produce a different effect. Assuming that the two waveforms start at the same zero crossing, and with the same phase, then the waveform with the higher frequency will gradually move ahead of the slower waveform, and its phase will be ahead. This means that from an initial state of constructive interference, the waveforms will pass through destructive interference and then back to constructive interference repeatedly. The rate of passing through these adding and cancellation stages is determined by the difference in frequency. For a difference of one-tenth of a hertz, it will take 10 seconds for the cycle of constructive, destructive and constructive interference to occur. This cyclic variation in level of the mixed waveforms is called 'beating', and sounds like a sound that 'wobbles' in level. This beating is often used in analogue synthesizers to provide a 'lively' or 'interesting' sound.

If the difference in frequency between the two waveforms is increased, then the speed of the beats will increase. When the frequency of the beats is above 20 Hz, then the mixed sound begins to sound like two separate frequencies. As the difference increases, the two frequencies will pass through a series of ratios of frequency, some of them sounding pleasant to the ear, and others sounding unpleasant. The ratio between the two frequencies is called an interval; the easiest and most 'pleasing' interval is a ratio of 2:1, an octave.

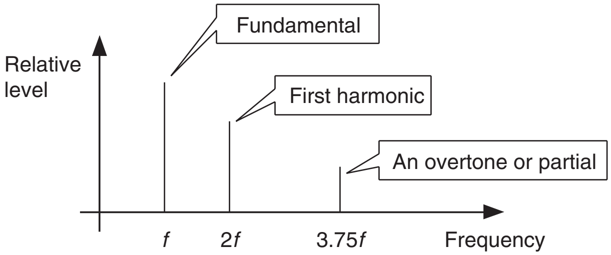


FIGURE 1.10.3 Timbre is set by the frequency content of a sound. In this example, the fundamental frequency of the sound is at frequency *f*, whilst there is a harmonic at twice this frequency, 2*f*. There is also an overtone or partial frequency at 3.75*f*. (Figure 2.3.7 provides an overview of spectrum plots like this one.)

Timbre

Timbre is a description of the contents of a sound. The timbre of a sound is determined by the harmonic content: the relationship between the level of the fundamental, the levels of the harmonics or overtones, and their evolution in time (see section 'Envelopes' in Section 1.10.1). Pure sounds tend to have only a few harmonics at low levels, whereas bright sounds tend to have many harmonics at high levels. Missing harmonics can also be important, and can produce 'hollow' sounding timbres. If the ratios of the frequencies between the fundamental and the other frequencies are not integers, then the timbre can sound bell-like or even like noise. The ability of the human ear to perceive timbre is related to the frequency. At low frequencies, the ear can detect phase differences and can follow changes in a large number of harmonics. As the frequency increases, the phase discrimination ability of the ear diminishes above A4 (440 Hz), and the number of harmonics that can be heard decreases because of the response of the ear.

For example, a sound that has a fundamental of 100 Hz has harmonics at 100 Hz intervals, and so the 150th harmonic is at 15 kHz. But a sound with a fundamental of 1 kHz has a 15th harmonic at 15 kHz. The number of audible harmonics are thus restricted as the fundamental frequency rises. Synthesizers provide comprehensive control over the frequency, phase and level of harmonics, and thus give the user control of the timbre (Figure 1.10.3).

Timbre (tahm-brer) is derived from a French word. 'Tone color' and 'tonal quality' are commonly used as synonyms for timbre.

Loudness

When a string vibrates, the size of the string and the amount of movement determine how much energy is transferred to the surrounding medium (usually air). The larger the amount of energy that is turned into changes in air pressure, the louder the sound will be. This can be demonstrated by using a tuning fork: it becomes much louder when it is placed on a tabletop, because it moves a much larger amount of air. The amount of movement of a vibrating object is called the amplitude of the vibration, whereas the amount of energy in the sound, which is produced by the vibrating object is called the power or

Table 1.10.2 Decibels

Sound pressure level (dB)	Sound pressure (microbars)	Power (Watts per square meter)	Power (Watts per square meter)	Equivalent	Musical dynamic
130	632	10 W	10	Threshold of pain	
120	200	1 W	1	Aircraft taking off	
110	63	100 mW	0.1	Loud amplified music	
100	20	10 mW	0.01	Circular saw	fff
90	6	1 mW	0.001	Train	ff
80	2	100 μW	0.0001	Motorway	f
70	0.6	10 μW	0.00001	Factory workshop	mf/mp
60	0.2	1 μW	0.000001	Street noise	p
50	0.06	100 nW	0.0000001	Noisy office	pp
40	0.02	10 nW	0.00000001	Conversation	ppp
30	0.06	1 nW	0.000000001	Quiet room	
20	0.002	100 pW	1E-10	Library	
10	0.006	10 pW	1E-11	Leaves rustling	
0	0.0002	1 pW	1E-12	Threshold of hearing	

intensity of the sound. Power is measured in watts, but a relative logarithmic scale is commonly used to avoid large changes in units: dB or decibels. Named after Alexander Graham Bell, the pioneer of telephony, decibels are used to indicate the relative difference between sound intensities or sound pressure levels (Table 1.10.2). The perception of sound power or level by humans is subjective: a change in sound power of 1 dB is 'just audible', whereas for something to sound 'twice as loud', the change is approximately 10 dB. The entire scale of sound intensity, from silence to painful, is just 12 doublings of sound power!

Musicians use an alternative relative measure for sound level. The 'dynamics marks' used on musical scores provide guidance about the loudness of a specific note. These range from ppp (pianississimo, softest) to fff (fortississimo, loudest), although this tends to be a subjective measure, and is also dependent on the instrument producing the sound. On average, the range covered by dynamics marks is approximately 50 or 60 dB, which represents a ratio of about a million to one in sound intensity (Table 1.10.3).

'Loudness' is a specific term, which means the subjective intensity of a sound, as opposed to intensity, which can be objectively measured by a sound intensity meter. The human hearing response to different frequencies is not flat: sounds between 3 and 5 kHz will sound louder than lower or higher pitched sounds, and a graph can be plotted showing this response, called an

Table 1.10.3 Dynamics

Musical dynamic	Name	Description	dB (approx.)
fff	Fortissississimo	Loudest	100
ff	Fortissimo	Very loud	93
f	Forte	Loudly	85
mf	Mezzo-forte	Moderately loud	78
mv	Mezza-voce	Medium tone	70
mp	Mezzo-piano	Moderately soft	62
p	Piano	Softly	55
pp	Pianissimo	Very softly	47
ppp	Pianississimo	Softest	40

equal loudness contour. This topic is covered by the science of psychoacoustics, which is the study of the inter-relationship between sound and its perception. Loudness is commonly used (incorrectly from a technical viewpoint) as a synonym for sound intensity.

Since sound is just pressure waves moving through a transmission medium like air, it can be measured in terms of the pressure changes which are caused. The unit for such pressure changes is the bar, although in common with many scientific units, smaller subdivisions such as millibars or microbars are more likely to be encountered in normal acoustics measurements. Since sound loudness is dependent on the response of the ear, it is measured in phons, where the phon is based on a subjective measure of the apparent loudness of sounds at different frequencies and intensities.

Envelopes

Sounds do not start and stop instantaneously. It takes a finite time for a string to start vibrating, and time for it to reduce to a stationary state. The time from when an object is initiated into a vibrating state is called the attack time, whereas the time for the vibration to decay to a stationary state again is called the decay time. For instruments that can produce a continuous sound, like an organ, the decay time is defined as the time for the sound to decay to the steady-state 'sustain' level, whereas the end of the vibration is called the release time (Figure 1.10.4).

The human sensory system seems to have a time resolution limit of about 10 ms, and thus, sounds that appear to start 'instantaneously' typically have attack times of less than approximately 10 ms.

Some instruments have long attack, decay and release times: for example bowed stringed instruments. Plucked stringed instruments have shorter attack times. Some instruments have very fast attack times: for example pianos, percussion. Very short times are often called transients. The combination of all the stages of a sound is called an envelope. It shows the change in volume of the sound plotted against time. The word envelope can also be used in a more

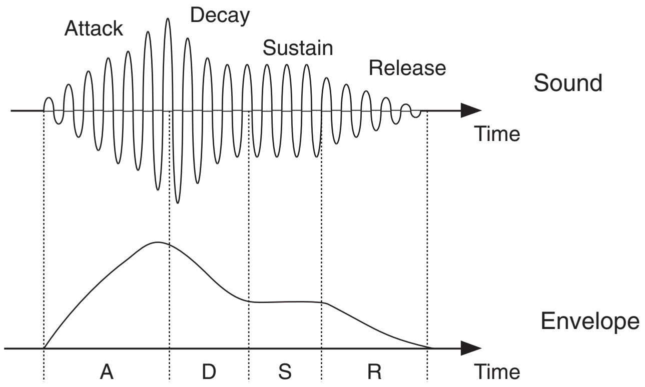


FIGURE 1.10.4
An envelope is the change in volume with time.

generic sense: it then refers to any complex time function. A typical example might be the envelope of a harmonic within a sound, which you would find in an additive synthesizer (see Chapter 3).

Gain and attenuation

The amplitude of a sound is a measurement of the extremes of its waveform: the most positive and negative voltages. If the amplitude changes, then the ratio between the original and the changed amplitudes is called the gain. Gains can be positive or negative, and can refer to amplitude or power, and are usually measured in dB. Gains of less than one are called attenuation, thus large attenuations mean that the audio signal can become very small, whereas large gains mean that the signal can become very large.

1.10.2 Electronics

Electronics is concerned with the study and design of devices that use electricity. Specifically it is concerned with the movement of electrons – tiny particles that carry a minute electrical charge and so produce electric currents when they move around circuits.

Voltage

Electrons flow through a conducting medium if there is a difference in the distribution of electrons, which means that there is an excess of electrons in one location, and too few electrons in another location. Such a difference is called a potential difference, or a voltage. Voltage is measured with a unit called the volt. The higher the voltage, the greater the potential difference, and the more electrons that want to move from one location to another. If the potential difference gets large enough, then the electrons will jump through air (which is what a spark is: electrons flowing through air). Normally electrons only flow through metals and other conducting materials in a more controlled manner.

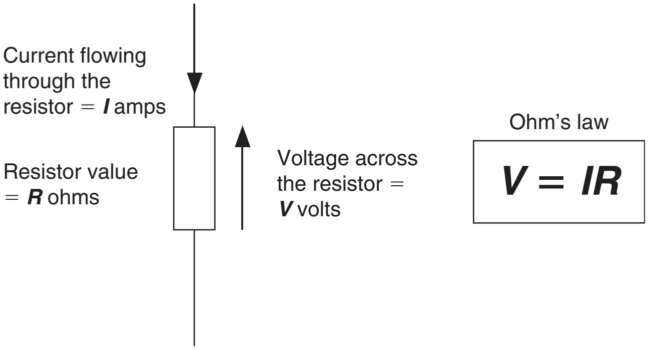


FIGURE 1.10.5 Another way of looking at the relationships between voltage, current and resistance is by considering the voltage across a resistor. If current *I* is flowing through a resistance of *R* ohms, then the voltage which will be present across the resistor will be *V* volts, where $V = IR$.

Current

Current is the name given for the flow of electrons. Using water as an analogy, the current is the flow of water, whereas the potential difference is the height of the water tower above the tap. The higher the water tower, the greater the pressure and the larger the flow when the tap is opened. To put things into some sort of perspective: a current of 1 ampere ('ampere' is normally shortened to 'amp' in common usage amongst electronics engineers) represents the movement of about 6000 million electrons per second.

Resistors are materials that impede the progress of electrons. Most metals will allow electrons to pass with almost no resistance, although very few materials present no resistance to the flow of electrons. Materials that allow electrons to pass through with no resistance are called superconductors: conductors because they 'conduct' electrons along, and super because they have no resistance to the flow of electrons. The word 'resistance' is actually used in electronics, but with a refined meaning: the resistance of a material is a measure of how hard it is for electrons to flow through it. Materials that do not allow electrons to flow are called insulators, whereas, materials that do allow the flow of electrons are called conductors.

Resistors

Electronic components are made that have specific resistances, and these are called resistors. Resistance is measured in ohms, and is the voltage divided by the current (Figure 1.10.5). If a current of 1 amp is flowing through a resistor, and there is a voltage of 1 volt across the resistor, then the resistance is 1 ohm:

$$R = V/I$$

where R is the resistance, V, the voltage and I, the current.

Resistors can range in value from very low resistances (fractions of ohms) for short lengths of metal wire, through to very high resistances (millions of

Sound conductance is the reciprocal of resistance, and it uses units called mhos, which shows that electronics engineers have a sense of humour!

ohms) for some materials which are on the borders of being insulators. For very high resistances, an alternative measurement is used: conductance.

When the current flows through the resistor, it produces heat. The amount of heat is determined by the product of the voltage across the resistor and the current. This is called the power, which is given off by the resistor, and it is measured in watts.

$$\text{Power} = V \times I$$

If 1amp flows through a resistor with a resistance of 1 ohm, then the voltage across that resistor will be 1 volt, and 1 watt of power will be dissipated as heat by the resistor. The small resistors that are found in most domestic electronic equipment, such as radios and hi-fi, will be 1/4 or 1/8 watt, and will be just less than a centimeter long and a couple of millimeters across. A 'typical' value would be 10,000 ohms.

Capacitors

Having said that electrons carry a charge, and that the flow of charge is called a current, what happens if no current flows? Charge can be stored by having a device, which stores electrons, and this is called a capacitor; since it has a 'capacity' for holding charge. You 'charge up' a capacitor by applying a voltage to it. Once it has stored a charge you can remove the voltage and the charge will stay in the capacitor (although it will gradually decay away in time). The size of a capacitor is measured in farads (named after Michael Faraday, a major pioneer in early electricity and magnetism experiments) and has the symbol F. This is a very large unit, so large that 1F capacitors are very rare. Capacitors are normally measured in smaller sub-units of farads: F, nF or pF are the most common units. Large capacitors are often quoted in tens of thousands of F, which represents a few hundredths of a farad.

Inductors do not like change. If you pass a current through an inductor, it initially tries to prevent the current flowing as the magnetic field is produced. When you try to stop the current flowing, the magnetic field is converted back into current to try and maintain the current flow. This is why you sometimes get arcing at the contacts of devices that have lots of coils inside: like electric motors. The current is trying to keep flowing, even across gaps in the circuit!

Inductors

Inductors are almost the opposite of capacitors – instead of storing charge, they temporarily store current. An inductor is often made from a coil of wire, and the action of current flowing causes a magnetic field to be produced. The energy from the current flow is thus stored as a magnetic field. If the current is removed, then the magnetic field will collapse and produce a current as it does so. The energy is thus converted from current in to magnetic field and back again. The 'size' of an inductor is measured in henrys (H), and again, this is such a large unit that hundredths and thousandths of henrys are much more likely to be found in common use.

Transistors

The transistor is a device that uses special materials called semiconductors (Figure 1.10.6). Silicon and germanium are two examples. A semiconductor is a material whose resistance is normally very high, but to which the addition

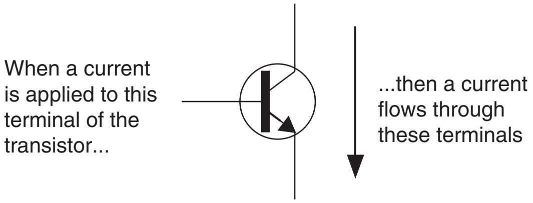

When a current is applied to this terminal of the transistor...
...then a current flows through these terminals

FIGURE 1.10.6 A transistor uses one current to control another. It can be used as an amplifier, a voltage-to-current converter or a switch.

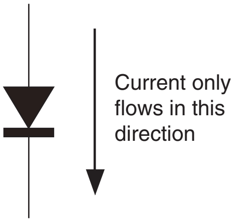


FIGURE 1.10.7 Current only flows through a diode in one direction.

of tiny amounts of other elements can alter the resistance in useful ways. By controlling exactly how these other elements are placed in the semiconductor material it is possible to produce devices that can control the flow of currents. A transistor is one such device. It has three terminals: current flows between two of these only when the third has a small current flow too. The control current is much smaller than the main current flow, and so the device can be used as an amplifier. If the control current turns on and off, then the main current turns on and off too, thus the transistor can also be used as a switching device. Transistors are the basis of almost all electronics.

Transistors that use current as the control are called bipolar or junction transistors, although there are other types which use electric fields to control the main current, and these are called field effect transistors, abbreviated to field effect transistors (FETs). FETs use very small currents indeed, and are widely used in electronics, particularly in making chips (see later).

Diodes

Diodes are simple semiconducting devices which allow a current to flow only one way. Inside there is a barrier which prevents the flow of electrons in one direction, but which breaks down and lets the electrons flow past in the other direction. When the current flows, the diode then behaves like a low value resistor, and so some heat is produced. If the barrier is made from a special material, then the effective resistance is higher, but instead of heat, light is produced, and these are called light-emitting diodes, or LEDs (Figure 1.10.7).

The functions of both diodes and transistors used to be produced by using valves. Valves were small evacuated glass tubes which had a small heating element that was used to excite a special material so that it emitted electrons which traveled across the valve to a collecting plate. Current could only flow from the emitter (called a cathode) to the collecting plate (called the anode), and thus a diode was produced. By putting a grid in between the cathode and the anode, the flow of current could be controlled in much the same way as a transistor.

Integrated circuits

Integrated circuits, or ICs for short, are an extension of the process that is used to make transistors. Instead of putting just one transistor onto a piece of silicon, the first ICs 'integrated' a complete two transistor circuit onto one piece of silicon. As the technology developed, resistors and capacitors were added, and the number of transistors increased rapidly. By the mid-1990s, ICs made from hundreds of thousands of transistors had become common.

The development of sophisticated stand-alone computer ICs from very humble cash register origins has produced the microprocessor. Commonly known as a 'chip', microprocessors carry out a vast range of processing tasks: a typical item of consumer electronic equipment will contain several: a video cassette recorder (VCR) could have a 'chip' dedicated to dealing with the front panel and IR remote control commands. Another would keep track of the programming and time functions, whereas another might handle the tape transport mechanism.

The use of microprocessor chips has had a major effect on the evolution of synthesizers: most notably in the change from analogue to digital methods of producing sounds. In a wider context, chips have become ubiquitous in most items of electronic equipment, but their function is completely unknown to all but a very small number of users of the equipment. One illustration of this change is hobby electronics. In the 1970s it was possible to buy kits of parts to build your own analogue synthesizer, and large numbers of people, including the author, actually built or adapted these kits and constructed synthesizers. In the process of building the synthesizer, the constructor would learn a lot about how it worked, and so would be able to repair it if it went wrong. In the 2000s, such kits are considerably rarer, and fewer people have the time or skill to build them – few items of electronics fail, and those that do are often replaced rather than repaired. The twenty-first century equivalent of the synthesizer kit is the PC, and although it is certainly possible to program your own synthesizer, the complexities are such that few people do. However, many people utilize software written by those few to make music.

Environment: form and function

An end product that uses digital and analogue electronics is often defined by its functions. The user does not need to know what type of storage is used in a dictation machine as long as it captures and plays back speech. In simple products the functionality is expressed in the 'form' of the device. A user could guess what the function of a dictation machine was by observing the microphone and the tiny cassette or flash memory card slot, plus the control labeling. And a few investigatory presses of buttons would quickly reveal how to use it.

But PCs are intended to be generic devices. The function or operation of a dictation program on a computer might not be obvious at all, and for complex programs training might be required to be able to do anything! In many ways, a synthesizer is a generic instrument in much the same way. You do not need to know in detail how it works inside, but you do need to have a usable model for how it makes the sounds, how to control them and the environment in which you would use it. Playing a synthesizer might not be obvious, and training might be required to make any noises at all!

Because it is so vital to know the environment in which a synthesizer is used, each chapter of this book has an 'Environment' section, where this topic is discussed.

The preceding sections are not intended to be complete guides to either electronics or acoustics. Instead they aim to give an overview of some of the major concepts and terms which are used in these subjects. Further information can be found by following the references in the bibliography.

1.10.3 Units

Technical literature is full of units, and these units are often prefixed with any of the symbols which show the relative size of the unit. A familiar example is the use of the meter for measuring the dimensions of a room, but kilometers are used for measuring the dimensions of a country. A kilometer is 1000 meters, and this is shown by the 'kilo' prefixed to the basic unit: the meter.

Table 1.10.4 gives some conversions between units and prefixes.

Table 1.10.4 Units

Name	Symbol	Ratio	Ratio
Peta	P	1E + 15	1 thousand million million times
Tera	T	1E + 12	1 million million times
Giga	G	1,000,000,000	1 billion times (1 thousand million times)
Mega	M	1,048,576	1,048,576 times
Mega	M	1,000,000	1 million times
K	K	1,024	1 thousand and twenty-four times
Kilo	k	1,000	1 thousand times
Milli	m	1/1,000	1 thousandth
Micro	μ	1/1,000,000	1 millionth
Nano	n	1/1,000,000,000	1 billionth (1 thousand millionth)
Pico	p	1/1,000,000,000,000	1 million millionth

So one microsecond (μs) is one millionth of a second, whereas one megahertz (MHz) is one million hertz.

When the size of computer memory is described, prefixes are often used that refer to powers of two instead of powers of ten. A Kbyte of memory does not mean 1000 bytes of memory, instead it refers to 1024 bytes of memory. Kbytes are sometimes mistakenly called kilobytes. A similar confusion can arise over the use of the prefix M. One Megabyte (MB) of memory is 1,048,576 bytes of memory, and not a million! In this case, although the use of the word megabyte in this context is technically wrong, it has entered into common usage and become widely accepted.

The same warning about ambiguity applies for the prefix Giga: it can mean 1000 cubed, or 1024 cubed (1,073,741,824 bytes). The International Electrotechnical Commission has tried to promote the use of a different term: Gibi, for 1024 cubed, but popular usage continues to use Gigabytes rather than Gibibytes (GiB). Of course, whenever specifications are used, the decimal (1000-based) value is used, since it appears to be larger. 1,048,576 bytes is almost 5% larger than 1 million bytes! Unfortunately, a 500 thousand million byte hard drive (500 GB) actually only has a capacity of approximately 465 × (1024 cubed) bytes (450 GiB) as far as the computer is concerned, because computers always use the 1024-based figure.

For the next prefix, Tera, the same problem applies, and the discrepancy between the two versions is larger. For a Terabyte, 1000 to the fourth power is one million million, whilst 1024 to the fourth power is 1,099,511,627,776 bytes, which is nearly 10% different, meaning that a 1 Terabyte drive is actually only approximately 910 thousand million bytes of storage.

1.11 Analogue electronics

Analogue electronics is concerned with signals: audio, video, instrumentation or control signals. These are usually direct representations of the real-world value, but converted in to an electrical signal by some sort of transducer or converter, analogue signals indicate the value by voltage or current. For example, a device for measuring the level of a liquid in a tank might produce a voltage and by connecting this voltage to a calibrated indicator or meter, the level can be monitored remotely. Being able to connect a meter across an analogue circuit and directly measure a voltage is typical of analogue circuitry, and is rarely possible with digital circuits, which normally require more complex equipment to monitor what is going on.

Analogue electronics is not always about voltages. Some signals are carried along cables as currents (and current waveforms) rather than voltages, with Ohm's law describing the relationship between the currents and voltages. Analogue electronics covers a wide range of voltages and currents. A cathode-ray tube (CRT) television has voltages of several tens of thousands of volts inside, and a Public Address (PA) amplifier might be delivering tens of amps into the speaker cabinets, whilst the voltage along the wire to a pair of stereo headphones will be only fractions of volts for quiet sounds, and a low-power op-amp might be consuming only fractions of amps of current. In contrast, digital circuits tend to use 5 volts or less, and individual currents flowing in digital circuits tend to be very small, but the total current can be a few amps because there are lots of circuits.

In many cases, analogue electronics is used for the input and output parts of a device, although the majority of the device is digital. CD, DVD and MP3 players have analogue outputs for audio or video signals, and may have power supply circuits with a lot of analogue circuitry, but the remainder is digital.

Signals in analogue electronics are often shown as plots of the value against time. These waveforms are often interpreted and drawn as if they were centered on a value of zero. So the use of the term 'zero crossing' does not necessarily mean that the waveform actually passes through a zero position, but merely that it passes an arbitrary line, which is approximately mid-way between the highest and lowest points of the oscillation.

Because analogue electronics works with direct representations of values in an electrical form, any distortion or interference can affect the quality of the signals. Thus if a signal that is supposed to be 4 volts is changed to 4.1 volts then this could change the tuning of an oscillator, or the cut-off frequency of a filter quite drastically. Digital circuits use voltage and current in a different way: the numbers 1 and 0 are represented as a high and low voltage or current, and so if anything above 3 volts is considered to be a '1', a change from 4 to 4.1 volts has no effect at all on the number. Because of this fundamental difference, analogue electronics is very concerned with the quality of circuits, the components used in those circuits, and the interconnections between circuits.

Operational amplifiers

Operational amplifiers, or op-amps, are one of the basic building blocks of analogue electronics (Figure 1.11.1). Although individual transistors can be used

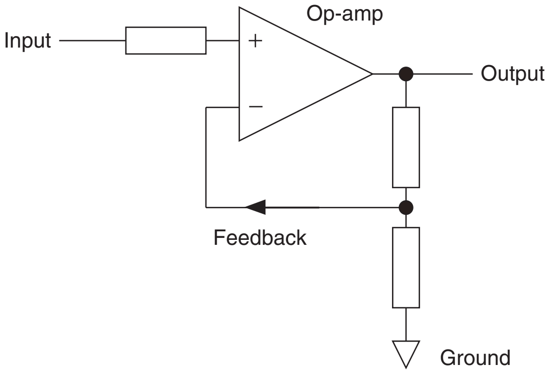


FIGURE 1.11.1
An op-amp has a very large gain and so it needs feedback to be applied in order to reduce the gain to a known amount.

as amplifiers, they are not perfect and have limitations of distortion, gain. Op-amps are built from several transistors, and provide idealized, near-perfect gain blocks which are easy to control and use in circuits. Op-amps have a very large gain (amount of amplification), and in normal use this is deliberately reduced by feeding some of the output back into the input, rather like the way that people tend to shout if they cannot hear themselves.

The integrator is one example of an analogue processing element which can be created using an op-amp. By connecting the output of an op-amp to the input through a capacitor, the resulting circuit can only change its output slowly – at a rate set by the time that the capacitor takes to charge. An integrator can be used to convert a sudden change into a smooth transition, and is a simple filter circuit which 'filters' out rapid changes.

The oscillator is a variation on the integrator. If a capacitor is arranged so that it acts as a timing element for an op-amp circuit, the circuit will repeat the cycle of charging and discharging the capacitor continuously. This produces a repetitive output at a frequency set by the time it takes for the capacitor to charge and discharge.

Filters are sophisticated versions of integrators. They come in many forms, and most have a gain, which is dependent on frequency, exactly the opposite of a hi-fi amplifier that aims to produce a 'flat' or consistent gain for any input frequency. Filters have a wide range of use in synthesizers.

Connections

Analogue electronics tends to be connected together with separate cables for each function. The two phono connectors used to connect stereo audio hi-fi equipment together carry the left (white) and right (red) signals. Yellow phono connectors are probably video signals. Analogue synthesizers are typically connected together using two sets of cables: one for the voltage representing the pitch of the note being played, and another for a voltage or current that indicates when the note is being played. Digital synthesizers and computers are connected together with MIDI (or USB) cables, where many different signals are carried along a single cable.

Analogue connectors are often round and have radial symmetry: Jacks, phonos/RCA, 4mm/banana and XLR/Cannon connectors all meet these criteria, but there are plenty of exceptions.

The gender of a plug or socket is often important when connecting equipment together, and there are a number of conventions that are useful to know.

- A plug is normally a connector at the end of a cable.
- Plugs are normally found in pairs: one at each end of a cable.
- Plugs often allow the metal that carries the voltage or current to be seen and touched by fingers, but in some circumstances, particularly for high voltages or currents, the plugs may be designed so that the metal cannot be touched if there are voltages of current present.

- A socket is normally on a panel or on the back of a piece of equipment.
- Sockets are often sources of voltage or current, and so the metal that carries the voltage or current is often not visible or touchable by fingers.
- A 'male' connector is one where the metal that carries the voltage or current is visible and touchable.
- A 'female' connector is one where the metal that carries the voltage or current is not visible and not touchable.

A good illustration of all of these points is 'mains' alternating current (AC) power cabling, which ranges from 100 to 240 volts, at frequencies of 50 or 60 Hz, around the world. The female sockets mean that it is difficult to touch the high voltage, and power cables normally have two different plugs at the two ends: one male plug that connects into the socket, and a female plug at the other end, so that it is not possible to touch the metal with the high voltage. The piece of equipment that is powered by mains power has a male plug on it, because it does not contain any source of voltage, thus touching the metal is not dangerous. Once the female plug is connected into the male socket on the piece of equipment, the metal carrying the voltage is protected from fingers touching it.

Audio and video cables normally work with lower voltages than mains power, and so many audio cables have male plugs at each end, and pieces of equipment have female sockets on them for inputs and outputs. But it is a good practice not to touch the metal that carries the voltage or current on a plug, and to touch only the case of the plug body. Pulling cables of any type by the cable rather than the plug body is not recommended under any circumstances, since it can either break the wires inside the cable, or expose potentially dangerous voltages if the plug or cable breaks.

Because analogue audio connections can require lots of cables, it is a good idea to have different colors or cable, or to put markings on the cables so that they can be easily identified. Rings of heat-shrink sleeving are a way to do this. Noting down the colors associated with specific connections can be very useful several months or years later when the connections need to be changed. Without such a record, it may be necessary to remove all the cables and put them back again, just to change one pair of connections.

The role of electronics

The introduction, in 1969, of performance-oriented 'extra', electronic keyboards which were intended to be used in conjunction with another 'main', more traditional, mechanical keyboard, is very significant. From this point onwards, musical performance involving electronics has changed and evolved more or less continuously through to the present day. The keyboard gradually moved from being an unseen accompaniment instrument to somewhere much closer to center stage in the 1970s and 1980s, and has since then moved back into the shadows as the guitar has returned to popularity, and as the two decks plus DJ has become a synonym for the use of sampling and sequencing. Even guitar-like 'keyboard controllers' with shoulder straps did not succeed in reversing this trend. Sometimes deliberate misdirection can be successful, as in the use of a guitar synthesizer to play drum sounds which was used in the 1990s and 2000s by Roy Wilfred Wooten ('Futureman'), the percussionist in Bela Fleck and the Flecktones.

Keyboards are not the only instruments that can appear static and disconnected when used live. Drum machines played by hitting drum pads are very visual, but programming live on stage is less visually interesting. In fact, the connection between pressing buttons on a control panel and the drumming sound which is then produced will not be immediately apparent to many people in an audience.

The guitar has also been changed by electronics. The electric guitar is much like the electric piano; a passive electromagnetic pickup (essentially a microphone) connected to an acoustic musical instrument. Just as the sound of electric pianos could be altered by phaser and flanger 'effects' boxes, so could the guitar, although the fuzz box and wah-wah pedal also extended the tonal range and performance possibilities of the guitar considerably in ways that do not work as well on keyboards. Distorted keyboard sounds tend to sound unwanted, although distortion on a guitar sound can be essential in some genres of music. But the tactile user interface of the guitar was less suited to replacement by electronics than the keyboard, and so the evolution of the guitar-controlled synthesizer has been slower and less far-reaching than the keyboard-controlled synthesizer. But the limitations of pressing keys on a keyboard, as compared to plucking, hammering on and damping of strings on a guitar, suggest that the guitar is a more expressive musical controller and so may be a key part of the ultimate musical controller (see Chapter 9).

Drums are another example where electronics has taken the physical instrument and changed it beyond all recognition, but in this case, the original physical drum has still survived, albeit augmented and often replaced by its electronic offspring. In many ways, the twenty-first century has seen the descendants of the drum take over almost all of the roles that the accompaniment section of drums, bass and rhythm backing used to occupy, leaving just lead vocals and solo instruments. The electrification of the drum is therefore very significant.

1.12 Digital and sampling

This section brings together the background principles behind the two major technologies used in digital musical instruments: digital and sampling.

1.12.1 Digital

The word 'digital' can be applied to any technology where sound is created and manipulated in a discrete or quantised way, as samples (numbers which represent the sounds) rather than continuous values. This tends to imply the use of computers and sophisticated electronics, although an emphasis on the technology is often a marketing ploy rather than a result of using digital methods to make sounds. Physical modeling synthesizers are an excellent counter example where much of the complexity of the digital processing is deliberately hidden from the user, and as a result, the synthesizer is perceived by the performer as merely a very flexible and responsive 'instrument'. Perhaps in the future we will see digital 'instruments' where the synthetic method of sound production is not apparent from the external appearance. (Although a power-supply cable might be a useful clue!)

1.12.2 Digital electronics

Digital electronics uses signals that represent real-world values as numbers. The numbers are held in binary form: voltages or current which can have only two values or states, on and off or one and zero. By using groups of these two-valued voltages, any number can be stored in digital form. One familiar digital circuit is a light switch: assuming that there is no dimmer, a light is either on or off.

Gates are simple electronic circuits which take one or more of these digital inputs and produce an output which is a logical function of them. For example, an output might only occur if both inputs are the same, or an output might be the opposite of the input. The rules for determining how these interactions take place is called Boolean algebra, and this is the branch of mathematics that is used to solve problems of the form: 'John does not cycle to work at the weekend. Bill travels to work, but only at the weekend. Simon has a car and he gives a lift to a colleague on Saturday. Who does Simon give the lift to?' (It could be John or Bill: more information is needed to provide a more definitive answer).

Registers are simple circuits which can store a binary value. Sets of registers can be used to hold whole numbers, and are known as memory. Real-world values that can change are represented as sequences of numbers, and this can occupy large amounts of memory. Audio signals require high precision and the frequent measuring of the value, and synthesizers, and especially samplers, may need to contain lots of memory chips in order to store audio signals.

Memory comes in two forms. Permanent storage is called read-only memory (ROM) and is used to store the instructions, which control how a piece of equipment works called the operating system. Digital information (data) is stored in ROM by physically breaking links inside the ROM with short bursts of high current. Temporary storage is called random access memory (RAM), since any of the data it contains can be directly accessed as it is required; in contrast to a serial memory such as a tape, where you need to wind through the tape to get to the required data. Some variants of ROM can be erased and rewritten: instead of using a permanent break in a wire, they store the data as charges on capacitors. These reprogrammable ROMs are called erasable programmable ROMs (EPROMs) or flash EPROMs, although this is commonly becoming abbreviated to just flash memory or flash drives.

Microprocessors are stand-alone general-purpose computers which are designed to carry out lots of logical operations very quickly and efficiently.

They do this by having memory stores and registers for values, a special arithmetic section which can carry out logical and mathematical functions on the values, and a way to control the movement and processing of the data: usually a sequence of instructions called a program.

DSPs are microprocessors which have been optimized to deal with manipulating signals: often audio signals, although video and other types of signal are also possible. DSPs have a streamlined architecture and special circuitry to carry out functions rapidly and efficiently.

An understanding of binary numbers would be essential for an understanding of computers in the 1970s or 1980s. In the twenty-first century, the underlying electronics is much less important, and an under standing of the operating system (Windows, MacOS, Linux, etc.) is essential.

1.12.3 Digital numbers

In order for digital techniques to work with sound, there needs to be a way to represent sounds and values as numbers. Digital systems use binary digits, or bits, as their basic way of storing and manipulating numbers. Bits tend to be organized into groups of eight, for various historical and mathematical reasons. A single bit can have one of two values: on or off, usually given the values 1 and 0, respectively. Eight bits can represent any of 256 values, from 0 to 255, or %0000 0000 to %1111 1111 in binary notation. The '%' is often used to indicate a binary number, and the binary digits (bits) are grouped into blocks of four to aid reading. A collection of 8 bits is known as a byte, and the blocks of 4 are called nibbles(!). Sixteen bits can be used to represent numbers from 0 to 65,535, and more bits can provide larger ranges of numbers. The 2 bytes that make up a 16-bit 'word' are called the most significant and the least significant bytes, are normally abbreviated to MSB and LSB, respectively. Note that the numbers are integers – only whole numbers can be represented with this method.

Larger numbers, and especially decimal numbers, require a different method of representing them. Floating point numbers split the number into two parts: a decimal number part from 0 to 9.9 and a multiplier or exponent part, which is a power of ten. The value 2312 could thus be regarded as being 2.31,21,000, and this would be stored as 2.312×10^3 in floating point representation. For binary numbers, a power of two is used instead of a power of ten, but the principle of splitting the number into a decimal number and a multiplier is the same. Floating point numbers can be processed using either a microprocessor or with special-purpose arithmetic chips called DSPs, which are optimized for the carrying out of complicated mathematical operations on numbers (some are designed for integers, whereas others are intended for floating point numbers); these are typically used for filtering, equalization and 'effects' such as echo, reverb, phasing and flanging.

Sampling always produces numbers that are an incomplete representation of the analogue original. But the amount of incompleteness can be made insignificant and unimportant with careful design.

1.12.4 Sampling

Sampling is the process of conversion from an analogue to a digital representation. An audio signal is a continuous series of values, which can be displayed on an oscilloscope as a waveform, whereas a digital 'signal' is a series of numbers. The numbers represent the value (the size, or magnitude) of the audio signal at specific points in time and these are called samples. The sampling process has three stages, which are repeated at a rate determined by a sample clock:

1. The audio signal is 'sampled'.
2. The sample value is converted in to a number.
3. The number is presented at an output port.

Samples are thus just numbers which represent the value, size or magnitude (measured in volts) of an audio waveform at a specific instant of time. These numbers are taken at the rate of the sample clock, and so a CD with a sample clock rate of 44.1 kHz processes 44,100 stereo samples per second. The opposite of sampling is the conversion from digital to analogue. This is called 'sample replay'. Replaying samples has three stages, which again are repeated at a rate set by the sample clock:

1. The number is presented to an input port.
2. The number is converted in to an analogue value.
3. The analogue value forms part of an audio signal.

Sample replay is the basis of almost all digital synthesizers. Regardless of how the digital sample is produced, the conversion from digital to audio is what produces the sound that is heard. Chapter 3 shows how sample replay has progressed from single cycle waveforms to complex looped sample replay.

1.12.5 Conversion

The conversion process from analogue in to digital and back again is at the heart of sampling technology. A complete digital audio conversion system, as used in a sampler, a direct-to-disk recording system, or a digital effect processor, typically consists of two sections. An 'analogue-to-digital' (ADC) section converts the audio signal into digital form and temporarily stores it in the sample RAM. The stages in the process are as follows:

- audio signal
- anti-aliasing filters
- sample and hold
- ADC conversion chip
- sample RAM containing digital sample values.

However, the 'digital-to-analogue' (DAC) section reverses the process and converts the digital representation of the audio back into an analogue audio signal. The stages in the process are as follows:

- sample RAM containing digital sample values
- DAC chip
- deglitcher
- reconstruction filter
- audio signal.

The majority of the actual conversion is achieved by two chips: ADC conversion is carried out by an ADC chip, whereas the reverse process of DAC conversion is done by a DAC chip. DACs are also commonly used inside ADCs (see Figure 1.12.3). In each case, there are two distinct and very different parts to the circuitry: the analogue audio part; and the digital sampled part. The analogue audio circuitry contains audio signals, whereas the digital sampled circuitry contains numbers that change at the sample clock rate.

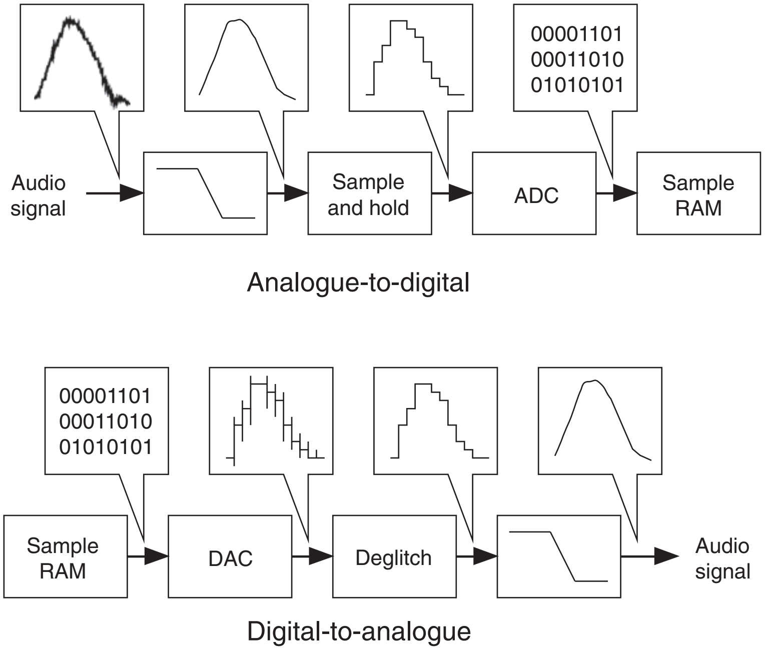
Although the names are different, the circuits which make up the two parts on either side of the sample RAM have very similar functions. The anti-aliasing filters prevent any unwanted audio frequencies from being converted by the ADC, whilst the reconstruction filter prevents any additional frequencies produced by the DAC's stepped waveform output from being heard at the audio output. The sample and hold circuit improves the quality of the conversion by presenting a constant level whilst the conversion is taking place, and the deglitcher prevents any momentary unwanted outputs from the DAC from being converted back into audio clicks (Figure 1.12.1).

PCM and pulse code modulation

The abbreviation PCM, meaning 'pulse code modulation', is often used in marketing material for digital synthesizers and samplers. It is taken from the terminology used in telecommunications and audio signal processing, where a 'modulation' is a conversion from one form in to another (analogue audio in to digital samples in this case), the 'pulse' refers to the regular timing between

FIGURE 1.12.1

An overview of a complete sampling system. Note that the filter at the input permanently removes the higher frequencies, and that the filter at the output reconstructs just the filtered version of the original audio signal.



samples, and 'code' refers to the conversion of the value or size of the audio signal into numbers. PCM sounds like a technical description, but its meaning is obscured because it was named at a time when several other earlier methods were widely useed:

- PAM, pulse amplitude modulation, where the output is not numbers, but pulses with different heights.
- PWM, pulse width modulation, where the output is not numbers, but pulses with different widths.
- PPM, pulse position modulation, where the output is not numbers, but pulses with different positions.

Because it converts signals in to a completely digital form, rather than just changing them into pulses where the size, duration or position are still analogue, PCM has become widely adopted, although some special purpose applications still use PAM, PWM and PPM. For example, 100BASE-T2 Ethernet cables use PAM, PWM is used in Class D audio amplifiers, while PPM is used for the radio signals that control the servos in many radio controlled cars, boats and planes.

The PCM used in telecommunications compresses the audio and is called G.711. The PCM used in digital audio is not compressed, and so is called Linear PCM.

Digital to analogue

A typical simple DAC has three parts:

- A latch to hold the digital numbers.
- A network of resistors to convert the number in to a voltage.
- An output buffer amplifier.

The latch holds the digital number which represents the sample value, and each number is held in the latch until the next sample value is available. In a CD player, the sample clock rate is 44.1 kHz and the samples change every 22 micro-seconds. The network of resistors is arranged so that the bits in the digital number produce voltage, which are proportional to their position in the number. Large value bits produce big voltages, and small value bits produce small voltages. These voltages are added together by the output amplifier, whose output is thus an analogue voltage, which represents the value of the digital number (Figure 1.12.2).

Analogue to digital

In a typical ADC, the audio waveform is examined at regular intervals of time (the sample clock rate: every 22 micro-seconds for a 44.1-kHz sample clock) and the value is held in an analogue memory circuit called a sample and hold. The sample and hold circuit is the point in the conversion circuitry where the audio signal is actually 'sampled', and it is designed to capture the instantaneous value of the voltage at that point in the waveform and hold it whilst the conversion process proceeds. If the sample and hold circuit takes too long to

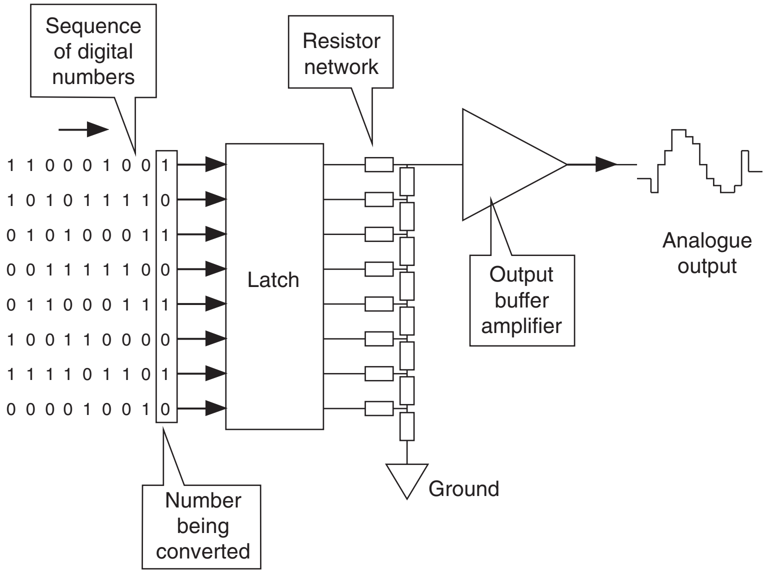


FIGURE 1.12.2
A DAC converts digital numbers to analogue voltages by using a network of resistors. The network is arranged so that the bits in the latch change the output voltage depending on their value, so the most significant bits have the largest effect.

capture the audio waveform, or if the held value changes whilst the conversion is taking place, then this too can degrade the quality of the conversion.

Once the sample and hold circuit has captured the value of the audio waveform, the held value is compared with a value which is produced by a counter and a DAC – in much the same type of circuit as that found in a wavetable synthesizer producing a sawtooth waveform (see Chapter 4). The counter counts up from zero, and as it does so, the ascending count of numbers is converted into a rising voltage by the DAC. A comparator circuit looks at the held value from the sample and hold circuit and the output of the DAC, and when they are the same, the comparator indicates that the two are equal, and the counter output is conveyed to the output of the ADC and held in a latch. The output of the ADC now holds a number that represents the value of the sample. The counter then resets and the ADC can begin to process the next value from the sample and hold circuit (Figure 1.12.3).

The detailed operation of some ADCs may differ from this, but the principle is the same – the audio signal is sampled, the sample value is converted in to a numerical representation of the value and the number appears at the output port of the ADC. This process is repeated at the sample clock rate. Some ADCs achieve the conversion process in different ways, and the output format of the number can be either serial (one stream of bits) or parallel (several streams carrying a complete sample).

1.12.6 Sampling theory

In order for the process of taking a sample of the audio signal and transforming it into a number to work correctly, a number of criteria have to be met.

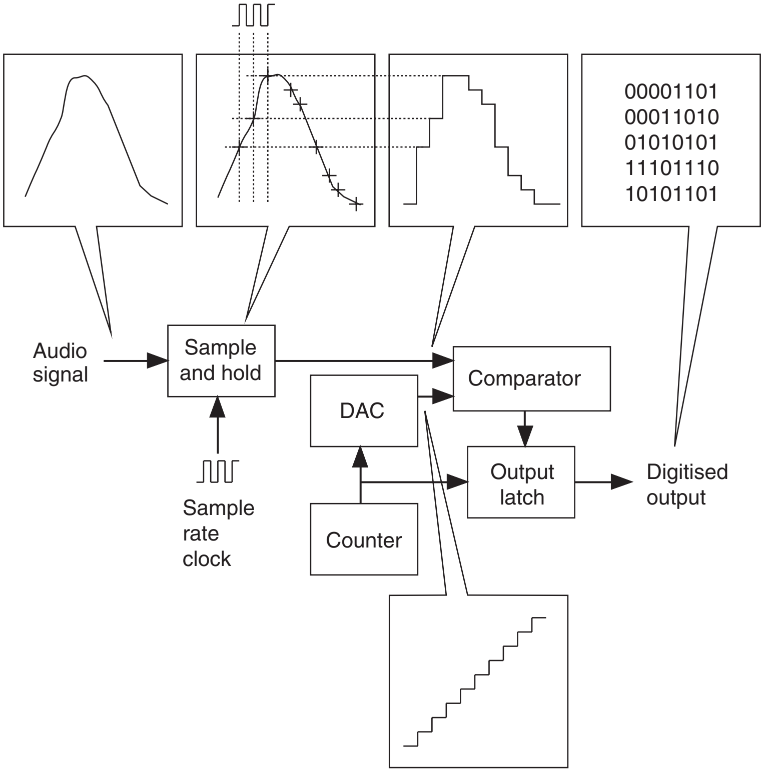


FIGURE 1.12.3 An ADC takes an audio signal, samples it and then compares the value with the output of a DAC driven by a counter. When the two values are the same, the comparator latches the output of the counter into the ADC output latch. The analogue signal sample has then been converted into an equivalent digital value. This process repeats at the sampling frequency.

First, the rate at which the samples are taken must be at least twice the highest frequency, which is required to be converted, this in practice means that the input is normally filtered so that the highest frequency which can be present is known. Secondly, the samples must be taken at regular intervals – any jitter or uncertainty in the timing can significantly degrade the conversion quality.

Finally, the numbers used to represent the signal must have enough resolution to adequately represent its dynamic range.

1.12.7 Sample rate

The simplest representation of a waveform at a given frequency is two sample values: ideally the top and bottom peaks. The time between these two peaks represents half of the period of the waveform drawn through them, which assumes that it is a sine wave and if this is the highest frequency, it must be a sine wave. Because two points are needed, the sampling needs to be at least twice as fast as the frequency which is being sampled. This requirement that 'the sampling frequency is at least twice the highest frequency in the signal' is called the Nyquist criterion. Note that if the sampling rate was exactly twice the frequency of the audio signal, the two points would always be at exactly the same values on the waveform, which could include zero, and there might be no output at all. Sampling is not normally done synchronously, and so a sampling rate, which is at least twice that of the highest frequency in the audio signal, will enable the same waveform to be reconstructed at the output of a subsequent DAC section.

If the audio signal is sampled at a rate, that is higher than twice the highest frequency which is present in the audio signal, no additional information is provided by using a higher sampling rate. The Nyquist criterion thus represents the most efficient rate at which to sample a given audio signal with a specific highest frequency component. However, sampling at higher frequencies can simplify the design and implementation of the filtering and some other parts of the circuitry. In the limiting case, some ADCs sample at several hundred times the Nyquist rate and then process the resulting 1-bit representation to produce the equivalent of more bits sampled at a lower rate. But the basic amount of information, which is required in order to be able to reconstruct the audio signal, still remains constant (Figure 1.12.4).

1.12.8 Filtering and aliasing

If any frequencies are present in the audio signal which are above the 'half-sampling' frequency, they will still be sampled, but the effect will be to make them appear to be lower in frequency; this process is called aliasing. It can be likened to a security camera, which looks at a room for a few seconds every 5 minutes. If the room has someone inside the first time that the camera is active, but not the second time, then there are several scenarios for what has happened. The obvious case is that the person was present for the first 5 minutes

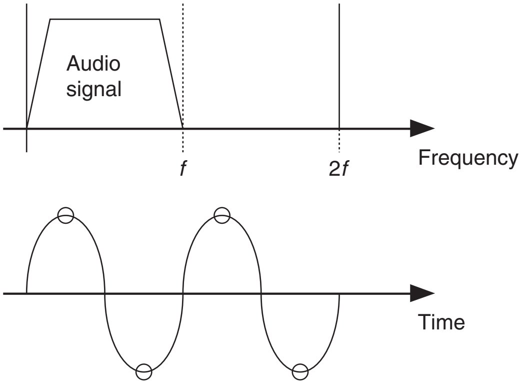


FIGURE 1.12.4 The Nyquist rate is twice the highest frequency which is present in the signal to be sampled. At least two samples are needed in order to provide a single cycle of a waveform.

and so was observed when the camera was active, but who then left before the camera was active the second time. Alternatively, the person could have been in and out of the room several times, and just happened to be present first time, but not the second. The important point is that the two cases appear the same from the viewpoint of the camera. Aliasing behaves in the same way – an aliased high frequency appears as a lower frequency in the digital representation, and it is not possible to reconstruct the original higher frequency. It is a 'one-way' process where information is lost or becomes ambiguous.

To prevent information from getting lost by the sampling process, anti-aliasing filters are used to constrain the audio signal to below the half-sampling frequency. This ensures that only frequencies that can be reproduced are sampled, and guarantees that the DAC will be able to output the same audio signal. The design of these filters affects the quality of the conversion, since they need to pass all frequencies below the half-sampling rate, but completely reject all frequencies above the half-sampling rate. 'Brick wall' filters with flat pass-bands and high stop-band rejection are difficult to design and fabricate, and in practice, the cut-off frequency of the filter is set to slightly lower than the half-sampling rate, and the stop-band rejection is chosen so that any frequency which passes through to the conversion process will be so small that they will be lost in the inherent noise of the converter (Figure 1.12.5).

The half-sampling frequency sets the highest frequency which was present in the original audio signal before it was sampled, and this is the highest frequency which will be reproduced when the sample is replayed. Thus for a sample rate of 44 kHz, the highest frequency which can be reproduced by the replay circuitry will just be less than 22 kHz. But reproducing sample values from a memory device also produces unwanted additional frequencies. Consider again the limiting case of two adjacent sample values, which represent a sine wave at just under the half-sampling frequency: when these are read out from a memory device, they will form the equivalent of a square-shaped waveform. Additional filtering is required to remove these extra frequencies, and a sharp low-pass filter with a cut-off frequency set to near the half-sampling rate is normally used. This filter is often called a reconstruction filter, and it limits the output spectrum of the sample replay to those frequencies which are below the half-sampling frequency. Any unwanted frequencies which are not removed by this filter are called aliasing frequencies.

Domestic CD players with a 44.1 kHz sample rate, and thus a half-sampling rate is 22.05 kHz, normally quote the upper limit of the audio signal frequency response as being 20 kHz. Professional DAT recorders typically use 48 kHz sampling, and also quote a 20 kHz upper frequency response. The filter is thus not as sharp and can be of higher quality. This 44.1/48 kHz sample rate/20 kHz bandwidth has become a 'de facto' standard for samplers and digital synthesizers.

Sample rates of 96 and 192 kHz began to appear at the close of the 20th century, and have increasingly been used for converting audio at the start of an otherwise all-digital processing chain based on computers and hard disk

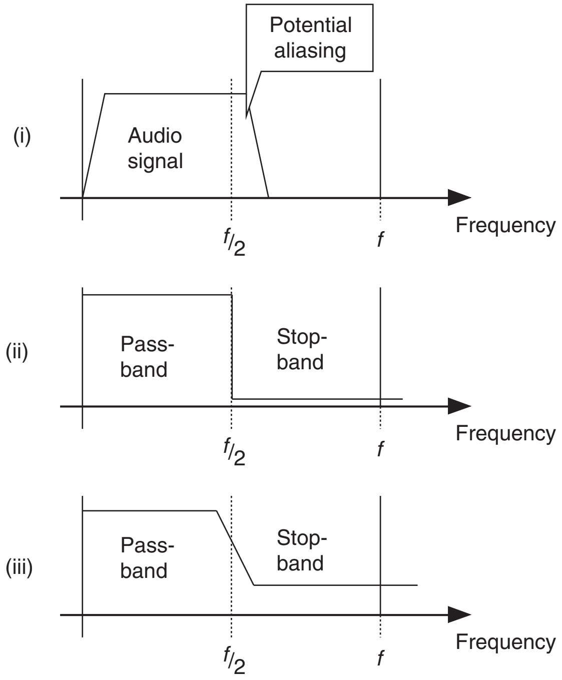


FIGURE 1.12.5 (i) If an audio signal contains some frequencies which are higher than half of the sampling frequency, then aliasing can occur. (ii) An anti-aliasing filter prevents this by having a pass-band which is set so that frequencies above the half-sampling frequency are in the stop-band of the filter. Theoretically this filter should pass everything below the half-sampling frequency and stop everything above. (iii) In practice, filters with a practically realizable cut-off slope and sufficient stop-band attenuation to prevent audible aliasing are used.

storage. Since CDs and DATs are designed around the 44.1/48 kHz, a number of alternative enhanced CD formats, plus DVD Audio, that use higher sampling rates have been produced, but these have not seen wide public acceptance.

1.12.9 Resolution

The size of the numbers that are used to represent the sample values determine the fidelity with which the audio signal can be reproduced. In digital circuitry, the number of bits, which are used to represent the sample value, limits the range of available numbers. In the simplest case, a 1-bit number can have only two values: 1 and 0. For each additional bit which is used, the number of available values doubles: thus for 2 bits, four values are available. Three bits provide eight values, 4 bits sixteen values and so on. In general, the number of available sample value numbers is given by:

$$D = 2^n$$

where D is the number of available values and n is the number of bits used. The number of available numbers to represent the sample values affects the precision of the digital version of the original audio signal. If only one bit is available, then only a very crude version of the audio signal is possible. As more bits are used to represent the sample values, the ratio between the largest and the smallest number which can be represented increases and it is the size of the smallest change which determines how good the resolution is. As the number of bits increases, the detail which can be represented by the numbers improves. This reduces the distortion, and for a typical 16-bit conversion system the distortion will be more than 90 dB below the maximum output signal.

The number of bits, which are used to represent a sample is important because it sets the limiting value on the output quality of the signal. The relationship can be approximated by the simple formula: $S \times 6n$ dB, where S is the signal-to-noise (and distortion) ratio (SNR) – the ratio between the loudest audio signal and the inherent noise and distortion of the system, often called the dynamic range, measured in dB – and n is the number of bits. This is the performance of a perfect system, and represents the 'ideal' case: real-world digital audio systems will only approach these figures (Table 1.12.1).

Table 1.12.1 shows the number of bits versus the 'ideal' dynamic range. As the table shows, a 'CD quality' output should have a dynamic range of nearly 96 dB: 'better than 90 dB' is frequently quoted in manufacturer's specifications. Note that the entire audible range, from silence to painful, can be covered by 20 bits. It thus appears that using between 16 and 20 bits should be adequate for almost all purposes. Unfortunately, this is not the case, and the simple example of volume control illustrates the problem.

Suppose, a digital synthesizer design uses 16-bit numbers to represent the audio samples, and the volume control is implemented by manipulating the digital audio signal. Thus, for a maximum output volume (0 dB), all of the 16 bits in the audio samples will be used in the replay of the signal. A crude method of reducing the volume could be achieved by using less bits: shifting

Table 1.12.1 Bits and SNR

Number of bits	Dynamic range
8	48
10	60
12	72
14	84
16	96
18	108
20	120
22	132
24	144

the digital numbers to the right. Each bit which is removed reduces the volume by 6dB, so a coarse volume control might work by shifting the digital words to the right so that less bits are used, with zeroes added from the left. So as bits are removed the volume decreases, but there is a corresponding decrease in the dynamic range of the signal. For an audio signal which is at −48dB, only 8 of the original 16 bits are being used to produce the audio signal, which means that the output signal effectively has only an 8-bit resolution – the remainder of the signal has been filled with eight zeroes.

It should be noted that shifting digital numbers to the right is not a very useful way of making changes to the volume of a signal, since the 6dB steps are very coarse. In practice, the numbers are reduced or increased by using a multiplication device: often a special purpose signal processing chip.

Using only half of the available bits for an audio signal has two major effects on the audio. The reduction in dynamic range means that there is a corresponding increase in the background noise level, whilst the release of notes can become distorted, especially if reverb is used. This characteristic 'grainy' distortion is called 'quantisation noise', and is caused by the transition between silence and audio represented by just changing 1 bit: effectively the audio waveform has been converted into a pulse wave.

Reducing the volume by having less bits in the output signal is thus very different from always using all the available bits and changing the volume with an analogue volume control. With the analogue control, the full-bit resolution is always available, and so a −48dB signal would still have the same dynamic range as the original sample; even if some of this is buried in the background electrical noise of the system. Some types of DAC chips allow exactly this type of output. Multiplying DACs and floating point DACs can be used with two inputs: one of which represents the audio signal at the full-bit resolution, whilst the other input represents the volume control bits. This type of 'fixed with scaling' conversion system is in widespread use. For example, telephones do not use linear coding but their basic performance is approximately 8 bit for SNR, with about the equivalent of 12 bits for the dynamic performance; although the restricted bandwidth significantly affects the perceived quality. In synthesizers, the sample resolution is normally 16 bits, whilst the volume control can be 6 bits or more, which is sometimes translated as '24-bit DACs' in manufacturers' literature.

In audio signal terms, with an 8-bit integer numbers at a rate of 8kHz, the sound quality is comparable to telephone quality; 16-bit numbers at a rate of 44.1kHz are often quoted as being of 'CD audio quality', since this is the basic storage used by a CD player for audio.

1.13 MIDI, transports and protocols

MIDI has played a major role in the development of electronic music since 1983. Wherever possible, this book has deliberately avoided making too many explicit references to MIDI in order to prevent it becoming 'Yet another book on MIDI'.

For example, the envelopes described in Chapter 2 are mostly dealt with in terms of control voltages, gate pulses and trigger signals because these are likely to be the native interfacing for many analogue synthesizers although many users will also use a MIDI to control voltage converter box to enable the use of MIDI control.

Since some readers may not be familiar with MIDI, the remainder of this section provides some background information, although as synthesizers are increasingly implemented in software inside computers, detailed knowledge of MIDI is not as important as it was in the late 1980s and 1990s. But what is still important is an understanding of how a musical representation such as MIDI can be used to produce, mix, record and reproduce sounds.

1.13.1 Overview

MIDI provides an interface for the exchange of information between electronic musical instruments and computers. It is based around musical events, except in rare circumstances, musical sounds are not conveyed via MIDI. Instead, MIDI carries information about what is happening and occurrences such as: when a note has been pressed, when a drum is hit and when the sequencer has stopped. A MIDI equipped keyboard will thus output information about what is happening on its own keyboard; thus if some notes are played, it will output MIDI information as a series of 'messages', which describes what notes are played, as they are being played.

MIDI uses a serial digital interface, which means that it sends a series of binary numbers along a single cable, and the numbers represent musical events and values. The transmission of the numbers along the cable is done using current instead of voltage, and in fact the circuit is very simple: current flows from the sending device along the cable to the receiving device, where it lights an LED, and then the current travels back along the cable to the sending device, where the flow of current is controlled to indicate the numbers by flashing the light in patterns. The patterns used are blocks of ten flashes or non-flashes, where current flowing (LED is lit) is defined as 'zero', and no current flowing (LED is not lit) is defined as 'one'. The blocks have a zero at the start, then 8 bits of data, then a one to finish the block of 10. Note that by setting the 'no current' to indicate a one, disconnecting a cable sends only 'end of block' bits.

The light from the LED affects a light-sensitive transistor, which then produces voltages to represent the ones and zeroes which have been transmitted. This use of light in what is called an opto-isolator means that there is no electrical connection between a sending MIDI device and a receiving MIDI device, which helps to avoid problems with hum from AC power supplies.

Looking more generically at what is happening with MIDI, the circuit with the opto-isolator's LED and light-sensitive transistor is the physical part or 'physical layer' of the connection. The organization of the current flow, with an initial zero followed by 8 bits followed by a closing one, is the way that the information is carried from one device to another, and is called the 'transport'. The way that the 8 bits in those blocks are used to carry messages is called a 'protocol'. The words 'physical layer', 'transport' and 'protocol' are often used in computer networking, but they also apply to MIDI.

1.13.2 Ports

The MIDI interface that is present on a piece of hardware is called a MIDI port. There are three types, although only one or two of the types may be present on a given piece of equipment.

- The in port accepts MIDI data.
- The out port transmits MIDI data.
- The 'thru' (American spelling: MIDI was originally specified in the United States) port merely transmits a copy of the MIDI data which arrives at the in port.

All MIDI ports look alike: they consist of 180°5-pin Deutsche Industrie Norm (DIN) sockets, although each port will normally be marked with its function (in, out or thru).

1.13.3 Connections

Connecting MIDI ports together requires just one simple rule:

Always connect an out or a thru to an in.

In a MIDI 'network', information flows from a controller source to an information sink. A keyboard is often used as a source of control information, whilst a synthesizer module is usually an information sink. Thus the out port of the keyboard would be connected to the in port of the synthesizer module. MIDI messages would then flow from the keyboard to the synthesizer module, and the synthesizer could then be 'played' from the keyboard.

1.13.4 Channels

MIDI provides 16 separate channels, which can be thought of as television channels. A piece of MIDI equipment can be 'tuned' so that it receives only one channel, and it will then only respond to MIDI messages which are on that channel. Alternatively, it is possible to set a piece of MIDI equipment so that it will respond to messages on any channel called 'omni'. Some important MIDI messages are not channel-specific and can be received regardless of the channel that the MIDI equipment is tuned to. If more than 16 channels are required, then multiple MIDI ports and cables are used, analogous to getting a second aerial pointed at a different transmitter. Each additional MIDI port or cable provides another 16 separate channels.

1.13.5 Modes

MIDI has several modes of operation. The important ones are as follows:

- Monophonic (one instrument: one note at once).
- Polyphonic (one instrument: several notes at once).
- Multi-timbral (several different instruments at once: several notes at once).

Modes are normally important only to users of guitar controllers or other specialized uses.

1.13.6 Program changes

Continuing with the television analogy, MIDI calls sounds or patches 'programs'. The message that indicates that a program should change is called a 'program change' message. Any of 128 programs can be selected. If more programs are required, then a bank change message allows the selection of banks of 128 programs. For most applications, a program change number does not indicate a specific sound, but specialised mapping called 'General MIDI' (GM) does specify which program change number calls up what sort of sound from a sound module, and more advanced mappings specify a broader range of sounds and controllers: known as XG and GS.

1.13.7 Notes

One of the commonest MIDI messages is the 'note on' message. This indicates that a note has been played on a keyboard, although it could also mean that a sequencer is replaying a stored performance. The note on message contains information on the MIDI channel that is being used, what key has been played and how quickly the key was pressed: this is called the 'velocity'. As a shorthand method of sending messages, a velocity of zero is taken to mean a note off message, although a separate note off message exists. The MIDI note on message does not contain any timing information about when the key was pressed – the message itself is used to indicate that the key has just been pressed.

Other common note specific messages include:

- Pitch-bend message, which transmits any changes in the position of the pitch-bend wheel.
- After-touch messages (polyphonic and monophonic) which transmit information about how hard the keys are being pressed once they have reached the end of their travel. This is intended as an additional control source for introducing vibrato or other modulation by increasing the finger pressure on a key which is being held down.

1.13.8 MIDI controllers

A MIDI controller is something that is used to control part of a performance such as the modulation wheel which is often found on the left hand side of the keyboard on many synthesizers, and which can be used to introduce vibrato or other modulation effects into the sound. Another example might be a foot volume pedal which plugs into a synthesizer – it controls the volume of the synthesizer directly, but it may also cause the synthesizer to transmit MIDI volume messages which indicate the position of the foot pedal. There are a large number of possible controllers, with functions ranging from volume or portamento, through to one which can control the timbre of a sound or set an effect parameter. Only a few of the controllers are defined; many are deliberately left undefined so that manufacturers can allocate them for their own purposes.

1.13.9 System exclusive

Although there are lots of MIDI controller messages, there is an alternative way to provide control over a remote MIDI device. System exclusive (sysex) messages are designed to allow manufacturers to make their own MIDI messages. The sysex messages can be used to edit synthesizer parameters, to store sound data and to transmit samples.

1.13.10 MIDI files

MIDI files are a way to move MIDI sequencer file information between different sequencers; 3.5 inch IBM PC compatible floppy disks were typically used for storage of MIDI files in the 1980s and 1990s, but by 2008, USB flash drives had more or less replaced the floppy disk.

1.13.11 Reference

The Focal Press book on MIDI by Francis Rumsey (1994) gives excellent detailed information on MIDI and is recommended for reading. The official MIDI documentation (The MIDI Specification, published by the MIDI Manufacturers Association (MMA)) is very formal, rather technical and not intended for the general reader, but the MMA also publish more general guides.

1.14 Computers and software

Computers are general-purpose devices that do things based on commands that they are given. Although most computers are now digital, they have also been made by using mechanical technology, as well as analogue electronics. Other possibilities, such as optical computers, molecular computers and quantum computers exploit light, DNA or other assemblies of atoms, and particle physics, but are in the early stages of development.

The ancestors of the computer come from two areas: calculation devices and automation. Early calculation devices such as the abacus provided beads on wires as a mechanical store for numbers, plus a physical way to manipulate those stored numbers and do basic mathematical functions. By the middle of the twentieth century mechanical calculators used gears and cogs in sophisticated pieces of engineering to do much the same. The author used one at school in the early 1970s, just as low-cost electronic ‘four-function’ (add, subtract, multiply, divide) calculators were appearing and the days of mechanical calculators, slide rules and logarithm tables were numbered.

Automation is used when repetitive tasks need to be done without requiring the constant attention or physical effort of a human being. Water powered bellows for organs are a simple example, but weaving devices such as Jacquard’s textile loom from 1801 were sophisticated mechanisms that used punched paper cards to control the weaving to make complex patterns in the woven cloth. These cards were a form of command storage, and by changing the cards the same loom could be used to produce other patterns.

Programmability turns a calculator from an electronic replacement for a mechanical calculating device, into something which can do calculations that are beyond what a human being would attempt on paper or with mechanical aids. Simple programmability just allows a sequence of instructions to be stored and then applied repeatedly to lots of numbers.

But by allowing the instructions to be influenced by the results of the calculations, it was possible to make programs that would do one set of instructions in one set of circumstances, and a different set of instructions in another. So, instead of a sequence of cards being used to control a loom to create the same pattern, again and again, it is like being able to change from one set of cards to another. This 'branching' allows decisions about what instructions to follow, which changes the nature of programming totally. Programs can be written to do many different things, instead of repeating the same thing.

Programmability

Computers provide such a rich variety of functionality because of the deep programmability that is possible. This is a very significant difference between most mechanical devices and computing devices – the mechanical device does a limited set of functions and may be modifiable to do a few more, whereas a computing device is a general-purpose device that can be used for any functions for which programs have been written.

The modern era of personal computing started when computers moved from large, complex, special purpose processing devices used by large companies and universities, to more mundane tasks such as cash registers. The first microprocessors were designed to carry out the simple arithmetic functions required by an electronic replacement for a mechanical cash register. They were called 'micro'-processors because they were small chips that processed numbers (instead of large cabinets). It would have been possible to make dedicated number processing chips to carry out just the functions appropriate for a cash register, but by making them general-purpose number processors they could be used in other applications as well.

Modern computers have taken these early cash register chips and enhanced and improved them in terms of speed, processing power, storage and other parameters. They are now used in a huge range of electronic devices, domestically, commercially and industrially. The processing power and the sophistication of the programming techniques used to harness that power have both shown continuous and ongoing development.

To show the effect of computers, compare a company in the 1960s with one in the 2000s.

1960s

In the 1960s, reports would have been handwritten and passed to the typing pool where typists would use typewriters to produce a typed version, which would then be sent back for corrections, re-typed and eventually issued.

Calculations might be done on a company computer, or time purchased on a large computer off-site, or might be done with mechanical devices such as slide rules, or by hand. Diagrams would be produced by hand in the drawing office using pencils, pens and rulers, and the resulting drawings would be copied chemically as 'blue-prints'. Inter-departmental exchange of information would be done via 'memos': pieces of paper with the message on it, plus a circulation list: names that are crossed off as each person sees it, and it can take weeks for a memo to be seen by everyone. Research is done in the company library by librarians who subscribe to journals, catalog them and place them in order on the shelves.

2000s

In the 2000s, the report is written on a computer using word processing software, edited on a computer and printed out on a laser printer. Calculations would be done in a spreadsheet program on a computer. Drawings would be produced using a drawing program or computer-aided design (CAD) software on a computer and printed out using a large format inkjet printer. Information is exchanged using email and instant messaging on a computer and it will not take hours for it to be seen by everyone. Research is done using the Internet, which can show information from around the world on a computer screen.

(The 'Environment' section in Chapters 3–6 contains sound-making examples that can be used to compare different music creation environments.)

Types

Computers are normally presented in three forms: embedded, servers and desktop/laptop.

Embedded computers are built into other devices and tend to have specialised input and output capabilities. The presence of an embedded computer is often overlooked because the functionality is important, not how it is achieved. Examples include: digital watches, washing machines and vehicle engine management systems, DVD players, satellite navigation devices and video game consoles. Embedded computers tend to have limited amounts of storage, just enough processing power to suit the application, minimalistic user interfaces and simplified user controls. The embedded computer inside a modern digital music workstation has the music keyboard, front panel controls, and MIDI messages as its inputs, and its outputs are the audio outputs and MIDI messages, with perhaps an option to output digital audio or burn a CD-R.

Server computers are used to provide computing power remotely. Typically placed in 19 inch racks very much like the ones used in pro-audio (but usually without the flight-cases), servers are designed to give concentrated computing power without needing lots of keyboards and monitor screens, and so almost all of their operation can be controlled remotely over a network connection. Server computers are often co-located with large amounts of data storage, and are typically placed in secure locations with backup power supplies, flood protection, etc.

Servers provide the processing power, storage and databases for search engines, online banking, commerce and trading systems, and more.

Desktop and laptop computers are 'personal' computers. As the 1960s and 2000s comparison in earlier sections shows, the general-purpose nature of computers means that they get used for a wide variety of functions, and they allow one person to do work which used to require several different sets of skills. So, whilst the 1960s company had different departments with skilled people using specialised equipment, a 2000s company had people using computers running software that suits the task. A time-traveler visiting the 1960s would be able to tell what function an office did by the equipment that was being used, whereas in the 2000s, each office would just have computers and printers. PCs are used by individuals to carry out a range of functions, and they are stand-alone computers that can communicate with each other, and with servers and the Internet, via a network connection. They have limited storage and processing capability, but they can use servers to augment them when required.

PCs are often split into three parts:

1. The main box, which contains the processor, storage and power supply.
2. The monitor, display or screen.
3. Input devices such as a qwerty keyboard and mouse.

Laptops combine most of these into one hinged unit, whereas some desktops combine the display and main box.

1.15 Virtualization and integration

The utilization of general-purpose computers to replace specific machines and functions is also significant because it also reflects what has happened with computers and the software that runs on them. The computer hardware has been incrementally improved over time, with gradual increases in processor speed, number of processors, access to more memory, larger hard disks, and ever faster peripheral connections USB 2.0 is the latest at the time of writing, with USB 3.0 due soon. In the 1980s, specialised computers would be used for word processing, with a monochrome screen and a printer that was little more than a typewriter without a keyboard, or perhaps a leading-edge monochrome laser printer. Most computers had text-based user interfaces, perhaps with simple character graphics, and the mouse was a rare device found on CAD workstations in industry. Diagrams would be drawn using expensive CAD-oriented workstations with color high-resolution monitors, and would be printed in color by using flat-bed printers or x-y plotters. There would be very little commonality, other than the use of microprocessor chips, in these two setups: the hardware, operating system (the software that runs the computer itself), software, printer, monitors and other peripherals would probably be different and not easily inter-workable.

(Many of the features of a 1980s' computer could still be found in embedded computers used during the 1990s and later digital synthesizers and samplers: text-based interfaces, no mouse, proprietary storage formats…)

By the 2000s, similar general-purpose computers, monitors and printers can be used for most purposes. The operating system is likely to be one of just three, and they can all exchange files and provide much the same functionality. Large color monitors and printers are used for most tasks, and the user interface uses a mouse and a graphical display with a window-based operating system. A large number of hardware and software standards have replaced the manufacturer-specific solutions of the 1980s with ubiquitous standards compliance and the ability for computers to inter-work and inter-communicate.

But much has also happened in the software itself. Computer software has evolved much more rapidly, with several major changes to the way that software is written and used. The graphical user interface is an obvious example, but the operating systems, the essential 'internal' software that runs the computer itself, have changed from simple 'one program at once' operation with text-based interfaces, to complex graphical user interfaces which can run many programs simultaneously. Both operating systems and 'application' software have gradually become more sophisticated, and considerably larger and more complex, and a series of innovations have changed the way that software is programmed. Two examples will be covered here: object-oriented programming and virtualization/plug-ins.

Object-oriented programming

To print out a musical score, a piece of computer software will need to know something about how the musical symbols are represented in the score, as well as the capabilities of the printer. One simple way of doing this would be to write software that keeps track of where each symbol needs to be printed, and to know how to make the printer print those symbols. The drawback to this approach is that the software writer needs to know about the symbols, how they are represented in the score and how the printer can print those symbols onto the paper. If a different printer is used, or a different way of representing the symbols in the score is developed, then the whole of the software will need to be reworked.

Object-oriented programming provides a solution by splitting the problem into self-contained units, called 'objects'. Thus instead of one program that does everything, one master control program sends commands to objects that do everything. So the main program does not need to know how to print the score, it merely needs to have an object that knows how to interpret the score, and another object that knows how to print the score, and it then tells the 'interpret' object to print the score. The 'interpret' object sends the information that needs to be printed to the 'print' object, which has information about how to print to several printers, which then sends the appropriate messages to the printer.

What object-oriented programming does is make it so that specific information about how to do something is only used where it is needed. For example, the owner of a concert hall may not understand how to control an orchestra, but he knows that he can ask a conductor to do it, which leaves the owner free to sell tickets, organize publicity, etc.

And the conductor may not know how the radio broadcast technician sends the performance of the orchestra so that it can be heard on radio, but he knows that the technician does, which leaves the conductor free to work on getting the orchestra to perform the music. Object-orientation thus provides an abstraction that lets each level concentrate on just that part of the overall function. This greatly simplifies the programming of complex software, and makes it easier to debug and maintain.

Virtualization and plug-ins

Virtualisation is formally used to refer to the abstraction of physical resources in a computer. In the context of sound-making on computers, it can be used to refer to the way that sound-making computer software is increasingly providing apparently physical resources that are actually nothing more than software. For example, a reverb effects processor might appear to be connected into the effects loop of a mixer, which in turn appears to be fed signals from a number of audio tracks in what appears to be a digital tape recorder, but what is actually happening is that a computer is simulating all of the functionality and presenting a user interface to the end-user that appears to be familiar bits of audio hardware.

Encapsulation is one way that this physicality is emphasized. Plug-ins are software objects that produce sounds or process audio, and there are many different types. By providing one standardized way that plug-ins can be interfaced into sound-making software, it is easy to choose the plug-ins that are required, easy to install them and easy for them to be programmed. Without plug-ins, the software programmers would have to write software for every sound and audio processor they wanted, and the controls for them, in the user interface. But by providing an encapsulated plug-in interface, the programmers of plug-ins only need to concentrate on the sound-making or audio processing. In fact, the use of the term 'plug-in' is a virtualization of the encapsulation, since there is no physical plug or socket in the computer at all!

By virtualizing the controls so that they behave like actual hardware, and encapsulating the interface so that plug-ins can be inserted and removed at will, the end result is a very flexible sound-making environment.

1.16 Questions

This section is designed to act as a brief review of the subject covered in the preceding chapter. The answers are in the text.

1. What is sound synthesis?
2. What is the difference between a modular and a performance synthesizer?
3. Outline the major methods of sound synthesis.
4. What is acoustics?

5. What is electronics?
6. Outline the processes that are required to take a product from laboratory prototype to commercial production.
7. Describe some ways in which synthesizers can be used to make music.
8. Categorize 10 different sounds under the following categories: realistic, synthetic, imitative, suggestive or sympathetic.
9. Give examples of the three types of computer.
10. Compare and contrast an orchestra in the 1960s with a computer-based sound-making software program of the twenty-first century. Are any functions the same, and are any missing or different?

1.17 Timeline

Date	Name	Event	Notes
1500s	Barrel Organ	The barrel organ. Pipe organ driven by barrel covered with metal spikes.	The forerunner of the synthesizer, sequencer and expander module!
1582	Galileo	Galileo conceives the idea of using a pendulum as a means of keeping time.	
1600s	Gottfried Leibniz	Developed the mathematical theories of logic and binary numbers.	
1600	William Gilbert	Electricity is named after the Greek word for Amber.	William Gilbert was the court physician to Elizabeth I.
1612	Francis Bacon	Publishes 'New Atlantis, which describes all sorts of new current sound 'wonders in a passage starting: 'We also have sound houses…'	An essential quote in most books on electronic music.
1642	Blaise Pascal	First mechanical calculator.	Addition or subtraction only.
1657	Christian Huygens	Christian Huygens used the pendulum to regulate the timekeeping of a clock.	
1676	Thomas Mace	Thomas Mace used a thread and a heavy round object to mark musical time.	Also designed a lute with 50 strings in 1672.
1694	Gottfried Leibnitz	Devised a mechanical calculator that could multiply and divide.	
1696	Etienne Loulie	Etienne Loulie invented the 'Chronometer', an improvement on Mace's idea, but with a variable length thread.	
1700	J. C. Denner	Invented the Clarinet.	Single reed woodwind instrument.
1752	Benjamin Franklin	Flies a kite in a thunderstorm to prove that lighting is electrical.	

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1756–1827	Ernst Chladni	Worked out the basis for the mathematics governing the transmission of sound.	The 'Father of Acoustics'.
1768–1830	Jean Baptiste Fourier	French mathematician who showed that any waveform could be expressed as a sum of sine waves.	Basis of Fourier (additive) synthesis and FFT (Fast Fourier Transform).
1801	Valve Trumpet	The modern valve trumpet is invented.	Not all musical instruments are old!
1804	Jacquard	Jacquard punched cards invented.	Basis of stored program control, as used in computers, pianolas, etc.
1807	Jean Baptiste Joseph Fourier	Fourier published details of his theorem, which describes how any periodic waveform can be produced by using a series of sine waves.	The basis of additive synthesis.
1812	D. N. Winkel	Winkel invented a clockwork driven double pendulum timer – very much like a metronome.	
1815	J. N. Maelzel, brother of Leonard Maelzel	Invented the metronome and patents it.	Some dispute about Maelzel versus Winkel as to who actually invented the metronome.
1818	Beethoven	Beethoven started to use metronome marks in scores.	
1820	Oersted	Discovery of electromagnetism.	The basis of electronics.
1821	Michael Faraday	Discovered the dynamo, and formalized link between magnetism, electricity, force and motion.	Used in motors, microphones, solenoids.
1833	Charles Babbage	Invented the Difference Engine – mechanical calculator intended for producing log tables.	The electronic calculator eventually made log tables obsolete!
1837	Samuel Morse	Invented Morse Code	
1844	Samuel Morse	Invented the electric telegraph.	The first telegraph message was 'What hath God wrought?'
1846	Adolphe Sax	Invented the Saxophone.	
1849	Heinrich Steinweg	Steinway pianos founded by Heinrich Steinweg.	
1862	Helmholtz	Published 'On the Sensations of Tone'.	Laid the foundations of musical acoustics.
1866–1941	Dayton Miller	Worked on photographing sound waves and turned musicology into a science.	

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1868–1919	Wallace Sabine	Founded the science of architectural acoustics as the result of a study of reverberation in a lecture room at Harvard where he was a professor of physics.	
1876	Alexander Graham Bell	Invented the telephone.	Start of the marriage between electronics and audio.
1877	Thomas Edison	Thomas Alva Edison invented the cylinder audio recorder – the 'Phonograph'. Playing time was a couple of minutes!	Cylinder was brass with a tin foil surface – replaced with metal cylinder coated with wax for commercial release.
1878	David Hughes	Invented moving coil microphone.	
1878	Lord Rayleigh	Published 'The Theory of Sound'.	Laid the foundations of acoustics.
1887	Heinrich Hertz	Produced radio waves.	
1888	Emile Berliner	First demonstration of a disk-based recording system – the 'Gramophone'.	Disk was made of zinc, and the groove was recorded by removing fat from the surface, and then acid etching the zinc.
1895	Marconi	Invented radio telegraphy.	
1896	Thomas Edison	Invented motion picture.	
1897	Yamaha	Founded Nippon Gakki (Yamaha).	
1898	Valdemar Poulsen	Invented the Telegraphone, which recorded telephone audio onto iron piano wire (also known as the Dynamophone).	Thirty seconds recording time, and poor audio quality.
1899	William Duddell	Turned the noise emitted by a carbon arc lamp into a novelty musical instrument.	Known as 'The Singing Arc'.
1901	Guglielmo Marconi	Marconi sent a radio signal across the Atlantic.	
1901	Harry Partch	Experimented with 13 tones and other microtonal scales.	Mostly self-taught.
1903	Double-sided LP	The Odeon label released the first double-sided LP.	Two single-sided LPs stuck together?
1904–1915	Valve	Development of the Valve.	The first amplifying device – the beginning of electronics.
1906	Lee de Forest	Invented the triode amplifier.	The beginnings of electronics.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1908-	Oliver Messiaen	Serialism, Eastern rhythms and exotic sonorities.	Some of his music uses up to six Ondes Martenot.
1910–1920	Futurists	Futurists.	Category of music.
1912–	John Cage	Pioneer in experimental and electronic music.	Famous for ‘prepared’ pianos, and ‘4 minutes 33 seconds’ – a silent work.
1914	Hornbostel and Sachs	Published a classification of musical instruments based on their method of producing sound.	Idiophones, Membranophones, Chordophones, Aerophones, etc.
1915	E. C. Wente	Produced the first ‘Condenser’ microphone using a metal-plated insulating diaphragm over a metal plate.	Now known as a ‘Capacitor’ microphone.
1915	Lee de Forest	The first Valve-based oscillator.	
1916	Luigi Russolo	Categorizes sounds into six types of noise.	Also invented the Russolophone, which could make seven different noises.
1920s	Cinema organs	Cinema organs, using electrical connection between the console keyboard and the sound generation.	Also start to use real percussion and more: car horns, etc. – mainly to provide effects for silent movie accompaniment.
1920s	Harry Nyquist	Developed the theoretical basis behind sampling theory	Nyquist frequency named after him
1920	Lev Theremin	The Theremin – patented in 1928 in the United States. Originally called the ‘Etherophone’.	Based on interfering radio waves.
1920	Louis Blattner	The first magnetic tape recorder.	Blattner was a US film producer.
1920s	Microphone recordings	First major electrical recordings made using microphones.	Previously, many recordings were ‘acoustic’ – using large horns to capture the sound of the performers.
1920–1950	Musique concrète	Musique concrète.	Tape manipulation.
1923	John Logie Baird.	Began experiments with light sources and disks with holes in them for scanning images.	The beginnings of television and computer monitors.
1924	Moving coil loudspeaker	The modern ‘moving coil’ loudspeaker was patented by Rice and Kellogg.	Superior because of low audio distortion.
1925	John Logie Baird	First television transmission.	Across an attic workshop!
1925–	Pierre Boulez	Pioneer of serialisms and avante-garde music.	

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1928	Maurice Martinot and Ondes	Invented the Ondes Martenot – an early synthesizer.	Controlled by a ring on a wire – finger operated.
1929	Couplet and Givelet	Four voice, paper-tape driven 'Automatically Operating Oscillation Type.	Control was provided for pitch, amplitude, modulation, articulation and timbre.
1930s	Baldwin, Welte, Kimball & others	Opto-electric organ tone generators	
1930s	Bell Telephone Labs	Invented the Vocoder – a device for splitting sound into frequency bands for processing.	More musical uses than telephone uses!
1930s	LP groove direction	Some dictation machines record LPs from the center out instead of edge in.	This pre-empts the CD 'center out' philosophy.
1930s	Ondes	Ondioline – an early synthesizer.	Uses a relaxation oscillator as a sound source.
1930s	Run-in Grooves	Run-in grooves on records invented.	Previously, you put the needle into the 'silence' at the beginning of the track…
1930s	Bell Telephone Labs	Invented the Vocoder – a device for splitting sound into frequency bands for processing.	More musical uses than telephone uses!
1934	John Compton	UK patent for rotating loudspeaker.	
1934	Laurens Hammond	Hammond 'Tone Wheel' Organ uses rotating iron gears and electromagnetic pickups.	Additive sine waves
1935	AEG, Berlin	AEG in Germany used iron oxide backed plastic tapes produced by BASF to record and replay audio.	Previously, wire recorders had used wire instead of tape.
1937	Tape recorder	Magnetophon magnetic tape recorder developed in Germany.	The first true tape recorder.
1940s	Arnold Schoenberg	12-tone technique and atonality.	
1940s	Wire and Ribbon recorders	Major audio recording technology used either steel wire or ribbon.	High speed, heavy and bulky – and dangerous if the wire or ribbon breaks!
1943	Colossus	The world's first electronic calculator.	Built to crack codes and ciphers.
1945	Metronome	First pocket metronome produced in Switzerland.	
1945	Ronald Leslie	Patents rotating speaker system.	
1947	Conn	Independent electromechanical generators used in organ.	

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1948	Baldwin	Blocking divider system used in organ.	
1948	Pierre Schaeffer	Musique concrète.	
1948	Pierre Schaeffer	'Concert of Noises' Futurist movement. Invented music concrete.	Music concrete is made up of pre-existing elements.
1949	Allen	Organs used independent oscillators.	
1949	C. E. Shannon	Published book *The Mathematical Theory of Communications*, which is the basis for the subject of information theory.	Shannon's sampling theorem is basis of sampling theory.
1950s	Charles Wuorinen	Quarter tones.	
1950	John Leslie	Re-introduction of Leslie speakers.	They are a success this time.
1950s	Tape recorder	Magnetic tape recorders gradually replaced wire and ribbon recorders.	There were even domestic wire recorders in the 1950s!
1951	Hammond	Melochord	
1951	Herbert Eimert	Northwest German Radio NWDR in Cologne starts experimenting with sound using studio test gear.	Used oscillators and tape recorders to make electronic sounds.
1954	Milton Babbitt, H. F. Olsen and H. Belar	RCA Music Synthesizer mark I.	Only monophonic.
1955	E. L. Kent	Kent Music Box in Chicago. Inspired RCA mark II synthesizer.	
1955	Louis and Bebe Barron	Soundtrack to 'Forbidden Planet' is a 'tour de force' of music concrete using synthetic sounds.	
1955–1956	Stockhausen	'Gesang der Junglinge' mixed natural sounds with purely synthetic sounds.	
1957	RCA	RCA Music Synthesizer mark II.	Used punched paper tape to provide automation.
1958	Charlie Watkins	Charlie Watkins produced the Copycat tape echo device.	
1958	Edgard Varese	Produced some 'electronic poems' for the Brussels Expo.	
1958	RCA	RCA announces the first 'cassette' tape – a reel of tape in an enclosure.	Not a success.
1960s	Clavioline	Clavioline	British Patent 653340 & 643846.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1960s	Mellotron	The Mellotron, which used tape to reproduce real sounds.	Tape-based sample playback machine.
1960s	Wurlitzer, Korg	Mechanical rhythm units built into home organs by Wurlitzer and Korg.	
1962	Ligetti	Ligetti used the metronome as a musical instrument.	Actually he used 100 of them in concert.
1962	Telstar	The first telecommunications satellite to transmit telephone and television signals.	
1963	Don Buchla	Simple VCO, VCF and VCA-based modular synthesizer: 'The Black Box'.	Not well publicized.
1963	Herb Deutsch	First meeting with Robert Moog. Initial discussions about voltage controlled synthesizers.	
1963	Philips	Philips in Holland announces the 'Compact Cassette' – two reels plus tape in a single case.	A success well beyond the original expectations!
1964	Philips	The Compact Cassette was launched.	Tape made easy by hiding the reels away.
1965	Early Bird	First geo-stationary satellite.	
1965	Paul Ketoff	Built the 'Synket', a live performance analogue synthesizer for composer John Eaton.	Commercial examples such as the Minimoog and ARP Odyssey, soon followed.
1966	Don Buchla	Launched the Buchla Modular Electronic Music System – a solid-state, modular, analogue synthesizer.	Result of collaboration with Morton Subotnick and Ramon Sender.
1966	Rhythm machine	Rhythm machines appear on electronic organs.	Non-programmable and very simple rhythms.
1968	Walter Carlos	*Switched On Bach*, an album of 'electronic realizations' of classical music, became a best seller.	Moog synthesizers suddenly change from obscurity to stardom.
1969	Philips	Digital master oscillator and divider system.	
1970	ARP Instruments	ARP 2600 'Blue Meanie' modular-in-a-box released.	
1970s	Ralph Deutsch	Digital generators followed by Tone-forming circuits.	The popularization of the electronic organ and piano.
1970	Tom Oberheim	Founded Oberheim Electronics.	US company.
1971	ARP Instruments	The 2600, a performance-oriented modular monosynth in a distinctive wedge shaped box.	The 2600 got modulars out of the studio and was hugely influential.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1972	E-mu	E-mu founded by Dave Rossum. Initial products are custom modular synthesizers.	
1972	Hot Butter	*Popcorn* became a hit single.	
1972	Roland	Ikutaro Kakehashi founded Roland in Japan, designed for R&D into electronic musical instruments.	First products are drum machines.
1973	John Chowning	Published paper: *The Synthesis of Complex Audio Spectra by Means of Frequency Modulation*, the definitive work of FM.	FM introduced by Yamaha in the DX series of synthesizers 10 years later.
1973	Oberheim	First digital sequencer.	The first of many.
1974	George McRae	'Rock Your Baby' is first record to completely replace the drummer with a drum machine.	
1974	Kraftwerk	*Autobahn* album was a huge success. A mix of music concrete technique and synthetic sounds.	
1974	Sequential Circuits	Sequential Circuits was founded by Dave Smith.	US company.
1975	Fairlight	Fairlight was founded by Kim Ryrie and Peter Vogel.	Australian company.
1975	Moog	Polymoog was released.	More like a 'master oscillator and divider' organ with added monophonic synthesizer.
1977	Roland	MC-8 Microcomposer launched: the first 'computer music composer' – essentially a sophisticated digital sequencer.	Cassette storage – this was 1977!
1978	Electronic Dream Plant	Wasp Synthesizer launched. Monophonic, all-plastic casing, very low-cost, touch keyboard – but it sounded much more expensive.	Designed by Chris Hugget and Adrian Wagner.
1978	Philips	Philips announced the compact disk (CD)	This was the announcement – getting the technology right took a little longer
1979	First Digital LPs	First LPs produced from digital recordings made in Vienna.	A mix of analogue playback and digital recording technology.
1980	Electronic Dream Plant	Spider Sequencer for Wasp Synthesizer. One of the first low-cost digital sequencers.	252-note memory, and used the Wasp DIN plug interface.
1981	Moog	Robert Moog was presented with the last Minimoog at NAMM in Chicago.	The end of an era.
1981	Roland	Roland Jupiter-8. Analogue 8-note polyphonic synthesizer.	

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1981	Yamaha	Yamaha R&D Studio opened in Glendale, California, USA.	
1982	Moog	Memorymoog – 6-note polyphonic synthesizer with 100 user memories.	Cassette storage! Six Minimoogs in a box!
1982	Philips/Sony	Sony launch CDs in Japan.	First domestic digital audio playback device.
1982	PPG	Wave 2.2, polyphonic hybrid synthesizer, was launched.	German hybrid of digital wavetables with analogue filtering.
1982	Robert Moog/MIDI	First MIDI Specification announced by Robert Moog in his column in Keyboard magazine.	
1982	Roland	Jupiter 6 launched – first Japanese MIDI synthesizer.	Very limited MIDI specification. 6-note polyphonic analogue synth
1982	Sequential	Prophet 600 launched – first US MIDI synthesizer	6-note polyphonic analogue synth – marred by a membrane numeric keypad.
1983	Oxford Synthesizer Company	Chris Huggett launched the Oscar, a sophisticated programmable monophonic synthesizer.	One of the few monosynths to have MIDI as standard.
1983	Philips/Sony	Philips launched CDs in Europe.	Limited catalog of CDs rapidly expanded.
1983	Roland	Roland launched the TR-909, the first MIDI equipped drum machine.	
1983	Sequential Circuit	Sequential Circuit's Prophet 600 is first synthesizer to implement MIDI.	Prophet 600 was marred by awful membrane switch keypad.
1983	Yamaha	Launched 'Clavinova' electronic piano.	
1983	Yamaha	Launched MSX Music Computer: CX-5.	The MSX standard failed to make any real impression in a market already full of 8-bit microprocessors.
1983	Yamaha	Yamaha DX7 was released. First all-digital synthesizer to enjoy huge commercial success. Based on FM synthesis work of John Chowning.	First public test of MIDI is Prophet 600 connected to DX7 at the NAMM show – and it worked (partially!).
1984	Yamaha	Marketing of custom LSIs began.	Yamaha began to market their in-house expertise to the world market.
1985	Akai	The S612 was the first affordable rack-mount sampler, and the first in Akai's range.	12-bit, Quick-Disk storage and only 6-note polyphonic.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1985	Ensoniq	Introduced the 'Mirage', an affordable 8-bit sample recording and replay instrument.	
1985	Korg	Korg announced the DDM-110, the first low-cost digital drum machine.	If was the beginning of a large number of digital drum machines.
1985	Yamaha	Yamaha R&D Studio opened in Tokyo, Japan.	
1986	Sequential	Sequential launched the Prophet VS, a 'Vector' synth which used a joystick to mix sounds in real time.	One of the last Sequential products before the demise of the company.
1986	Steinberg	Steinberg's Pro 16 software for the Commodore C64.	The start of the explosion of MIDI-based music software.
1986	Yamaha	Launched Clavinova CLP series electronic pianos.	CLP pianos were pianos – the CVP series add on auto-accompaniment features.
1986	Yamaha	DX7II was revised DX7 (a mark II).	Optional floppy disk drive.
1987	Casio	Introduced the Casio CZ-101, probably the first low-cost multi-timbral digital synthesizer.	Used Phase Distortion, a variant of waveshaping.
1987	DAT	DAT (Digital Audio Tape) was launched. The first digital audio recording system intended for domestic use.	Worries over piracy severely prevented its mass marketing.
1987	Roland	MT-32 brought multi-timbral S&S synthesis in a module.	If was the start of the 'keyboard' and 'module' duality.
1987	Roland	Roland D-50 combined sample technology with synthesis in a low-cost mass-produced instrument.	S&S synthesis (Sample & Synthesis).
1987	Yamaha	Yamaha DX7II centennial model – second generation DX7, but with extended keyboard (88 notes) and gold plating everywhere.	Limited edition.
1987	Yamaha	Yamaha R&D Studio opened in London, England.	
1988	Korg	Korg M1 was launched. Used digital S&S techniques with an excellent set of ROM sounds.	A runaway best seller. Filter had no resonance.
1988	Korg	Korg M1 workstation was launched. Used digital S&S techniques with an excellent set of ROM sounds.	A runaway best seller. Because it put synthesis, sequencing and mixing/effects into one device. Notably, the filter has no resonance.
1989	Breakaway	The Breakaway Vocaliser 1000 was a pitch-to-MIDI device that translated singing into MIDI messages and sounds via its on-board sampled sounds.	Somewhat marred by a disastrous live demonstration on the BBC's 'Tomorrow's World' program.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1990	Technos	French-Canadian company Technos announced the Axcel – first resynthesizer.	There was no follow up to the announcement.
1991	General MIDI (GM)	First formalisation of synthesizer sounds and drums.	Specified sounds, program change tables and drum note allocation.
1992	MiniDisc	Recordable digital audio disk format released by Sony.	
1995	Yamaha	Launched VL1, world's first Physical Modeling instrument.	Duophonic, and very expensive.
1997	DVD	First DVD video players were released. DVD Audio standard did not appear until 1999.	
1997	Korg	Z1 polyphonic physical modeling synthesizer.	
1998	Yamaha	DJ-X, a dance performance keyboard disguised as a 'fun' keyboard.	Followed by a keyboardless DJ version, the DJXIIB.
1999	MP3	First MP3 audio players for computers appeared.	Internet music downloading began.
2000	Yamaha	mLAN, a FireWire-based, single cable for digital audio and MIDI.	Slow acceptance for a brilliant concept.
2001	Apple	iPod was launched.	Not a runaway success at first: a slow start.
2001	Korg	Karma, a combination of a synthesizer with a powerful set of algorithmic time and timbre processing.	Karma 2 added extra facilities and appeared in the OASYS, Triton and M3 instruments, with a stand-alone software version planned for 2008.
2002	Hartmann Music	Neuron Resynthesizer.	Arguably the first commercially produced resynthesizer.
2003	Yamaha	Vocaloid, mass-market singing synthesis software.	Backing vocals will never be the same again!
2005	Bob Moog	Bob Moog, synthesizer pioneer, died.	1934–2005 (pronounced to rhyme with 'vogue').

PART 2

Techniques

This page intentionally left blank

Chapter 2

Making Sounds Physically

CONTENTS

Sounds and musical instrument

Environment

This chapter deals with sounds that are made by physical methods. This serves two purposes:

- To introduce classification systems for musical instruments and sounds, and thereby, to start the discussion of the analysis and synthesis of sound.
- To introduce the chapter contents with a simple example.

2.1 Sounds and musical instruments

There are many ways to classify musical instruments and sounds.

The simplest division uses the performer's role: using the human vocal tract or interacting with a musical instrument or other objects. This matches the way that music is often described as being vocal, choral, instrumental or orchestral and is supported by descriptions such as 'full orchestra plus choir'. Unfortunately, human beings are also capable of producing sounds that are outside of the normal description of vocal or choral and can be described as a capella or speech effects: clicks, pops, whistling and noise-based sounds.

Sounds made by interacting with a musical instrument or other objects can be classified using the type of instrument itself or the part that is vibrating.

2.1.1 Instrument

- String instruments
- Wind instruments
- Percussion instruments.

This classification scheme uses the material used to make the instrument as the classifier. It is widely used for orchestral instruments in the West.

There are a number of variations and refinements such as brass instruments and keyboard instruments. But it does have limitations, particularly in the context of synthesis, since a synthesizer with a keyboard will be classified as a keyboard, but the same synthesizer controlled by a wind controller will be classified as a wind instrument.

2.1.2 Vibration

This scheme is concerned with what actually makes the sound. There are four basic traditional divisions, with the fifth added more recently.

1. Idiophones, where the sound is produced because the body of the instrument vibrates. Therefore, this group includes percussive instruments such as the marimba, bells and chimes, and wood blocks, as well as less obvious examples such as the triangle and a hand slap on the body of an acoustic guitar.
2. Membranophones, where the sound is produced because a tensioned membrane vibrates. This group includes all the drums with a stretched membrane or skin, plus the kazoo!
3. Chordophones, where the sound is produced because one or more strings vibrate. This group includes the guitar, violin and harp, as well as harpsichords, hammered dulcimers and pianos.
4. Aerophones, where the sound is produced because a column of air vibrates. This group includes the oboe, bagpipes, flutes, horns, trombone and saxophone, as well as the whistle.
5. Electrophones, where the sound is produced because a loudspeaker vibrates. This group includes all electronic instruments, although it generally does not include amplification of another type of instrument, and therefore, the electric guitar is still classified as a chordophone because the vibrating string is the initial source of the vibration.

This classification is easier to understand if you think about genericizing the bit that is vibrating. Aerophones and chordophones both basically vibrate something long and thin in one dimension (1D): a vibrating string or column of air. This produces strong resonances, and therefore, the sounds tend to be pure with a specific pitch. Membranophones are where the membrane can basically vibrate in two dimension (2D), and Idiophones are where the body of the instrument can vibrate in three dimension (3D). As the number of dimensions goes up, the resonances become more complex and weaker, and therefore, the sounds become more complex and with a more diffuse pitch.

This classification scheme is widely used by ethnomusicologists and is known as the Hornbostel–Sachs system. Synthesizers and samplers do not easily fit into this classification scheme, since although easily dismissed as being electrophones, the sound production technique may well be mathematically modeled on any of the other four types, and therefore should be classified appropriately.

In most of the groups mentioned, there are several ways in which the vibration can be caused: hit, scrape, twang and blow. A classification produced by using the vibrating part and the way in which the vibration is caused can also be used.

Classifying sounds rather than musical instruments is required when the sounds are not produced by musical instruments (sirens, wind, gun-shots and explosions are some examples) or are synthetic (bleeps, pips and others) in sound or creation technique. Onomatopoeia (e.g., bang, pop, hiss,…) can be useful for some of the non-instrumental sounds, but pure synthetic sounds can be hard to describe in words ('wee yah oh ooh').

In this book, the instrument type, the way of causing the vibration and onomatopoeia will all be used to describe instruments and sounds.

2.2 Hit, scrape and twang

Hitting things is probably the first interaction that humans made with potential sound-making objects and is the source of percussion instruments. Whereas hitting a hollow log or stone might be accidental at first, producing a drum with a stretched drum-skin requires design and effort. Hitting the stretched string of a bow is not as immediately satisfying as plucking it, and therefore the piano and the guitar hammer-on are relatively recent inventions. Hitting air is not as hard as it might at first appear: the hand-clap is one example. Sonic booms and whip cracks are somewhere in between hitting air and scraping it (Table 2.2.1).

Scraping pieces of wood, especially hollow ones with textured surfaces, needs some skill and preparation, although door hinges that need oiling can make some very distinctive sounds. Jazz brushes on drum-skins can sound like a sophisticated shaker. Scraping tensioned strings requires a lot of deliberation and knowledge about how to make a resonating body.

Twanging tensioned strings is interesting because it leads to trying to make the sound louder, which leads to resonators, and eventually opens the way for scraping of strings. Twanging membranes is very similar to hitting them, and twanging things sees its modern outlet with the ruler and the African thumb piano.

Table 2.2.1 Types of Sound-Making instrument and examples

	Hit	Scrape	Twang	Blow
Idiophones (3D)	Marimba, wood block	Scraper, waterphone, cuica	Jew's harp, thumb piano	Aeolsklavier
Membranophones (2D)	Drums	Jazz brushes		Kazoo
Chordophones (1D string)	Piano, guitar hammer-on	Violin	Guitar, koto	
Aerophones (1D air)				Wind, brass

2.3 Blow into and over

Blowing air over the end of a hollow object probably results in experiments with adding extra holes and trying different sizes of tubes. Blowing between two pieces of grass requires more preparation, and combining it with a tube is an intriguing inventive step. Blowing through the lips to produce whistling is just amazing. The whole process of blowing is interesting because of the way in which energy is transferred, often because of turbulence as the air hits a hard edge producing something not unlike scraping!

2.4 Sequencing

Physical instruments can be controlled by a number of sequencer-like mechanisms. One obvious human mechanism is a conductor, whilst a less obvious and more distributed mechanism is bell ringing, which works with patterns of ordering of the playing of the bells. Orchestras, bell-ringers, conductors and other human performers require energy and, usually, a sense of timing or rhythm.

Mechanical playback devices require some source of energy, either a spring, weights or a water wheel (often used in the past for what were called 'water organs'), a steam engine or other suitable power sources. This powers the musical instrument and the mechanism that converts the stored music from holes into physical controls over the musical instrument through cams and levers. Musical box movements have possibly the simplest arrangement, with pins in a rotating cylinder that twang tuned metal tines, whilst some fairground organs have very complex mechanical linkages to connect to a diverse set of musical instruments ranging from drums to violins. Brass instruments are difficult to control mechanically because the player's lips are essential to the playing technique and cannot be easily replaced with mechanical alternatives. Mechanical timing is often provided by a mechanical governor device, which limits the rotation speed, and as a result maintains the tempo, and which often uses either air resistance or gravity to reduce excessive speed and increase laggardly speed.

2.5 Recording

Physical instruments are played live. Using human memory to store music is certainly possible, but it can be difficult to pass on to other people other than by a physical performance. Writing down music in some sort of notation is a way of capturing the physical events that produced the music in a transferrable form, and it requires information about the note event start, the duration and the pitch.

Mechanical recording captures the sound waves by turning the movement of the air into a movement of a pen, scraper or gouge. This requires a horn to capture as much of the available sound as possible and some sort of recording medium to capture the movement of the pen, scraper or gouge as a mark, scratch or groove: paper, metal, wax,… The rotating cylinder with a spiral groove is a neat way of providing a long length of recording space in a compact design. It is interesting to note that the first sound recordings were made mechanically for the purpose of analysis of the human singing voice, and it was later that the commercial possibilities were exploited.

Mechanical recording of the events rather than the sound is also possible, and it is used in musical box movements, where a cylinder with pins stuck into it provides a visual and mechanical recording of the sound for subsequent playback. Punched wooden or card tablets can be used to control mechanical musical instruments such as pipe organs in much the same way as weaving looms with Jacquard cards. Player pianos use rolls of paper with holes in them to record a player's performance for later playback.

2.6 Performing

Live music requires a venue, one or more performers and an audience. The performers need to know the pieces they are going to perform, and at least one of them (usually the conductor or leader) knows the order in which they will be performed. An indication of pitch and tempo is often used at the beginning of the performance so that the performers can play in tune and time. During the performance, the conductor or leader will provide timing or pitch information as required.

Mechanical performance requires the playback device and an energy source. The audience is not essential: chiming clocks still make a noise even when no one is listening.

2.7 Examples

2.7.1 Hurdy gurdy

The hurdy gurdy is an interesting example of a mechanical instrument that is like a partly-automated violin. The strings are in contact with a rosined wheel instead of a bow, and the rotation of the wheel causes the slightly sticky surface to alternately stick to the string and then break away, thus pulling and releasing the string repeatedly, just as a violin bow does. Instead of guitar-like frets, the strings are adjusted for length with wedges that anchor the string, much like the fingers on the fretboard of a violin. Drone strings are also present, which gives the hurdy gurdy a similar musical repertoire to, and sometimes interchangeability with, bagpipes in some European folk cultures.

2.7.2 Barrel organ

Barrel organs take their name from the main storage device: a barrel or cylinder that contains pins that operate valves to direct air from a set of bellows through the appropriate tuned pipes of a pipe organ. Barrel organs were often human powered, and the barrels were frequently programmed by skilled individuals rather than mass produced. The human operative (the 'organ grinder') turned the barrel and operated the bellows but had no influence over the performance other than the tempo of the music.

2.7.3 Player piano

Player pianos, or pianolas, use a punched paper roll to control the playing of the keys of the piano. Unlike a pipe organ–based barrel organ, the dynamics need to be recorded and reproduced so that the punched paper rolls contain information about the pitch, the start and duration of notes, and the dynamics. Mechanical recording of actual performances using hydraulics or levers had a tendency to distract the performer, and it was not until electrical methods were used to capture playing that recording fidelity improved. But carefully punched transcripted rolls were adequate for most purposes, and these were mass produced in the latter decades of the nineteenth century and in the first decades of the twentieth century.

2.7.4 Phonautograph

The first conversion of sound into a visible form was in 1857. A Frenchman Édouard-Léon Scott de Martinville used a horn to capture sound that moved a bristle that pressed onto a blackened glass plate – later versions used a rotating cylinder. The oldest known sound recording has been recovered from one of these visible records by converting them back into sound using a computer to scan the image. It is of a 435 Hz tuning fork recorded in 1859. The first human voice recording using the same technique was made in 1860 and is a 10-second recording of the French Folk song 'Au Clair de Lune'.

2.8 Questions

1. Describe two alternative ways to classify musical instruments.
2. How would you classify an electric guitar?
3. How would you classify non-musical sounds?
4. Why would a mechanical brass instrument be difficult to make?
5. Why are dynamics important to a piano performance?

2.9 Timeline

Date	Name	Event	Notes
1949	Harry Chamberlin	Rhythmate 40.	A tape loop-based ancestor of tape replay units such as the Chamberlin and Mellotron, but this one played rhythms. Housed in a plain wooden box, with controls on the top.
1959	Wurlitzer	The 'Sideman' Rhythm Unit.	A wooden box by the side of the organ that produced drum sounds. Electromechanical design used rotating disk and contacts to time the 12 rhythms and valve-based circuits to filter and shape the 10 drum sounds.

(*Continued*)

Date	Name	Event	Notes
1963	Korg	Donca-Matic DA-20 Rhythm Unit.	The Keio Organ (Korg) company's first major product – designed as an improvement on the Sideman.
1972	Roland	TR33 Rhythm Unit – early transistor drum machine.	Drum pattern selection was through Dance Style – Bossa Nova, Beguine, Samba,…
1972	Technics	SL-1200 hi-fi turntable. Direct drive.	A hi-fi turntable for the serious enthusiast.
1977	Roland	Roland launched the MC8 – the first 'computer music composer'. A digital 8-part (track) sequencer, with an accompanying converter box to produce analogue voltages.	Cassette storage of the maximum 5300 note events.
1978	Roland	Roland launched the CR-78 Compu-Rhythm – one of the first commercial drum machines to provide user programmability.	Housed in a large box that was almost a cube, the CR-78 has a unique appearance – not too dissimilar to the very earliest rhythm units!
1979	Linn	LM-1 – sampled sounds as a contrast to the analogue drum machines of the time.	Although only about 500 were made, this was a hugely influential machine at that time.
1979	Roland	TR-808 Rhythm Composer – an analogue drum machine whose limitations (sounds, tempo stability) were its greatest assets. Widely misused live in the hip-hop, techno and house music genres.	Saw major success only after it had ceased production in 1981. The TR909 from 1984 is a Latin percussion follow-on.
1979	Technics	SL-1200 Mk2 hi-fi turntable. Became the definitive 'industry standard' DJ deck (current new model is the Mk.6).	A very informed design. The motor, casing and grounding were improved to give the Mark 2 version.
1980	Electronic Dream Plant	Spider Sequencer for the Wasp Synthesizer. One of the first low-cost digital sequencers.	252 note memory and used the Wasp DIN (Deutsche Industrie Norm) plug interface.
1980	Grand Wizard Theodore	Pioneer of scratching and needle drop techniques for vinyl disks.	Grand Wizard Theodore was a DJ and one of the first hip-hop producers from New York.
1980	Oberheim	DMX drum machine.	Pre-MIDI (musical instrument digital interface, although could be retro-fitted) sampled drum machine, using drum sounds in EPROMs (electrically programmable read-only memory).
1980	Sony	The 3.5-inch floppy disk introduced for portable data storage.	The 3.5-inch Sony floppy faced competition from sizes of 2, 2.5, 2.8, 3, 3.25 and 4 inches alternatives.

(*Continued*)

Date	Name	Event	Notes
1982	Linn	LinnDrum – the first commercially successful drum machine to feature digitally sampled drum sounds.	An upgraded LM-1 (better sampling rate and some new samples). The 'sound' of the early 1980s was almost all LinnDrum.
1982	Roland	TB303 'Bass Line' – a monophonic sequencer and simple single-VCO bass synthesizer. Intended originally as an accompaniment device for guitarists.	Found increased popularity just after production ceased in 1995. Manual adjustment of the filter cut-off and resonance knobs became the basis of 'Acid House' genre.
1983	Sequential	DrumTraks. One of the first MIDI equipped drum machines.	Analogue drums with (for the time) very sophisticated per beat programming of level and tuning.
1984	Roland	TR909 drum machine.	More accenting detail than the TR808 and shuffle to provide swing. The machine for techno and all forms of dance music.
1984	Yamaha	QX1 hardware sequencer.	Big, and it used 5¼-inch floppy disks. But it was accurate with 384 ppq timing resolution.
1986	Roland	TR-505 drum machine.	Budget 12-bit sample equipped drum machine with LCD (liquid crystal display) 'blob' view.
1986	Yamaha	RY30 drum machine. One of the last conventional 'studio' drum machines from the Japanese manufacturer.	Incorporated S&S generated drum sounds, plus a miniature modulation wheel-style real-time controller for volume, pitch, pan,…
1986	Yamaha	RX5 drum machine.	Top of the range at the time. Lots of pads, programmable drum pitch and drums sounds on plug-in cartridges.
1987	Korg	DDD-1 drum machine.	Sampled drum machine with ROM (read-only memory) Card port for additional drum sounds. Good MIDI implementation.
1988	Roland	D-20 synthesizer.	Included a sequencer and floppy disk storage.
1989	Roland	W-30 Music Workstation.	A sample- or S&S-based keyboard workstation with floppy disk storage and SCSI (small computer system interface) port for CD-ROM access.
1991	Roland	CR-80 drum machine with special randomizer to simulate human playing.	CD quality drum samples in Roland's last stand-alone studio drum machine.
1996	Novation	DrumStation drum module.	1U rack containing modeled drums from the 808 and 909 stable.
1996	Roland	MC303 Groovebox.	A combination of drum machine, sequencer, synthesizer and lots of preset and user-definable phrases that could be strung together easily into songs.

(*Continued*)

Date	Name	Event	Notes
1997	Jomox	X-Base 09 drum machine.	German revisiting of the classic 808 or 909 style of drum machine. Analogue sounds with a fully up-to-date feature list.
1998	Roland	MC505 Groovebox.	The second generation of the phrase sequencer box. Bigger and better.
2000	Traktor	Traktor, a software DJ solution was developed.	Later licensed to Native Instruments.
2001	Alesis	AirFX, 3D controlled effects unit.	Uses infra-red sensors to detect hand position and movement.
2001	Korg	Karma, a combination of a synthesizer with a powerful set of algorithmic time and timbre processing.	Karma 2 added extra facilities and appeared in the OASYS (Open Architecture Synthesis System), Triton and M3 instruments, with a stand-alone software version planned for 2008.
2002	Korg	Kaoss Pad KP2, 2D controller and effects unit.	Real-time control over effects.
2003	Native Instruments	Traktor DJ Studio 2.5 DJ software is launched.	Adds time-stretching, OSC (Open Sound Control) support and skins.
2004	Native Instruments	Guitar Rig, guitar audio path modeling software.	Models effects, amplifiers, speaker cabinets and even microphones.
2006	Native Instruments	Audio Kontrol, an audio interface with MIDI input/output, plus extra controller soft-knob and three soft-key buttons.	The controller knob and buttons allow detailed mapping to software keyboard shortcuts, as well as being conventional MIDI ‘learn’able controllers.

This page intentionally left blank

CHAPTER 3

Making Sounds with Analogue Electronics

CONTENTS

3.1 Before the synthesizer

The use of electronics for audio started with the invention of the telephone in the last part of the nineteenth century. Before this, microphones were very insensitive and produced lots of distortion, and loudspeakers were very quiet! Since then electronics has developed enormously and now offers sensitive microphones with low distortion, as well as loudspeakers that are loud, plus many other inventions.

3.1.1 Microphones and loudspeakers

Microphones and loudspeakers turn sound into electrical signals and vice versa. It is now such an everyday experience that it is difficult to appreciate how significant it was to the world of just over 100 years ago that had only natural sounds and gramophone recordings. Since then microphones and loudspeakers have been refined, and Alan Blumlein's invention of stereo in the 1930s enabled the positioning of sounds across a sound stage. By the 1960s, affordable hi-fi meant that anyone could experiment with audio. The 1970s saw commercial experimentation with what was then called quadrophonic sound, but would now be called 4.0 surround sound: four speakers instead of the two used in stereo. Quad's complexity, plus problems with standards for LP discs, meant that it was not a commercial success. In the twenty-first century, a number of researchers are using multiple microphones and surround sound loudspeakers to move complete sound-fields from one location to another.

3.1.2 Oscillators

Oscillators are pieces of electronics laboratory equipment that were used for musical purposes long before synthesizers became affordable. Simple oscillators provided sine waves, whilst more sophisticated ones could provide other waveshapes. Intended for use in radio or audio testing, they were usually not

This chapter describes analogue synthesis: from voltage control to musical instrument digital interface (MIDI); from monophonic to polyphonic; from modular to performance oriented; from subtractive synthesis to formant synthesis and beyond.

temperature stable and had continuously variable frequency dials that made their use for any pitched music difficult. Despite these problems, early experimental music groups such as The Silver Apples used multiple oscillators in performance in the late 1960s.

Although better known now for printers and computers, Hewlett HP, the US technology company had its roots in audio oscillators. The first product from Bill Hewlett and Dave Packard (Hewlett-Packard (HP)) was the Model 200A oscillator, the origins of which were in Bill's thesis at Stanford University in the late 1930s.

3.1.3 Mixers

Mixers take several audio sources and combine them. Often, mixers are used to combine a few selected audio signals from a larger set and so are also used as selectors or switches. Mixers effectively move the level or volume controls from the outputs of all the connected audio devices and put them into one device. This greatly eases the selection and balancing of levels from the audio devices.

3.1.4 Amplifiers

Amplifiers take an audio signal and amplify it. Microphone amplifiers are used for low-output microphones or for extra gain with quiet sound sources. Power amplifiers are used to drive loudspeakers in public address (PA) applications. Guitar amplifiers turn the quiet sounds produced by the strings and amplify the outputs from the electromagnetic pickups on the guitar to produce audible sound. By connecting a microphone into an amplifier that is driving a loudspeaker, it is possible to create feedback by adjusting the gain of the amplifier and the positioning of the microphone and loudspeaker. This can be used to create some interesting sounds, especially if the gain is reduced slightly so that it is just about to break into oscillation. Electric guitars can be used instead of a microphone, and the same effects can be produced because the strings and body of the guitar can pick up enough of the amplified audio to create a feedback loop.

3.1.5 Filters

Filters allow some frequencies to pass through, but reject others. They range from subtle tone controls to making large changes to the sound – one common use is to simulate the restricted bandwidth of telephones. Filters are used as audio laboratory test equipment and in recording studios.

3.1.6 Radio technology spin-offs

Oscillators, mixers, amplifiers, filters, modulation and many other devices and terms that are used in audio electronics are derived in part from radio electronics. Radio uses a combination of audio frequency electronics with much higher-frequency radio electronics. Sounds produced by radio receivers as radio stations are tuned in, or deliberately mistuned, are often used as sound effects or metaphors for communications. Radio modulation circuits, adapted for audio frequencies, are used to produce complex transformations on audio signals. In particular, ring modulation is frequently used to create alien and robot voices by processing speech.

3.1.7 Disks, wire and tape recorders

Pre-recorded sounds on disk can be used as sound sources, and a disk-cutting lathe can be used to create special effects such as looped tracks, or multiple sets of spiral grooves instead of just one. Loops can also be simulated manually by a human being manipulating the disk or turntable.

Tape recorders (or their older counterpart, wire recorders) can not only be used as sound sources but also be used as simple echo units by using one as a recorder and a second as a playback unit, with the tape passing from one to the other. By adjusting the distance between the two tape recorders, the echo time can be controlled. By feeding back the echo signal to the recorder, further echoes of the echoes can be produced, but this technique is prone to feeding back or amplification of the noise introduced by the tape recording and playback process.

Adjusting the playback of any mechanical audio playback device will change the pitch and the tempo. This can be used for various special effects.

3.1.8 Effects (reverb, echo, flange,...)

Reverb and echo effects can be produced by using a loudspeaker and microphone in a room, particularly if the room is large and has non-parallel walls so that the sound bounces around rather than just back and forth between two parallel walls. Flanging effects can be produced by mixing together the outputs of two tape-delayed audio signals and then adjusting the playback speed of one of the tape recorders, often by touching the flange of the tape reel.

3.1.9 Performing

The environment for creating sounds using analogue audio equipment before synthesizers offers a wealth of possibilities, and this should not be overlooked even in a world of digital electronics and computers.

One notable example of what can be done with equipment as described earlier is the original theme music for the BBC television programme called ‘Doctor Who’. This used audio oscillators adjusted by hand to produce the frequency swoops. The noise of the Tardis dematerializing is derived from scraping a piano string.

3.2 Analogue and digital

The word ‘analogue’ means that a range of values are presented in a continuous rather than a discrete way. ‘Continuous’ implies making measurements all the time, and also infinite resolution – although inherent physical limitations such as the grain size on photographic film or the noise level in an electronic circuit will prevent any real-world system from being truly continuous. 'Discrete' means that you use individual finite sample values taken at regular intervals rather than measure all the time, with the assumption that the samples are a good representation of the original signal. Digital synthesis uses these discrete values.

The word 'analogue' can also be spelt without the '-ue' ending. In this book, the longer version will be used.

An analogue synthesizer is thus usually defined as one that uses voltages and currents to directly represent both audio signals and any control signals that are used to manipulate those audio signals. In fact, 'analogue' can also refer to any technology in which sound is created and manipulated in any way where the representation is continuous rather than discrete. Analogue computers were used before low-cost digital circuitry became widely available, and they used voltages and currents to represent numbers. They were used to solve complex problems in navigation, dynamics and mathematics.

Analogue electronics happens to be a convenient way of producing sound signals – but there are many other ways: mechanical, hydraulic, electrostatic, chemical, etc. For example, vinyl discs use analogue technology where the mechanical movement of the stylus is converted into sound. Tape recorders reproduce sound from analogue signals stored on magnetic tape.

In synthesizers, the use of the word 'analogue' often implies voltage-controlled oscillators (VCOs) and filters (VCFs). These have a set of audio characteristics: VCOs can have tuning stability or modulation linearity problems, for example; and analogue filters can break into self-oscillation or may distort the signal passing through them. These features of the analogue electronics that are used in the design can contribute to the overall 'tone quality' of the instruments.

Analogue synthesizers are commonly regarded as being very useful for producing bass, brass and the synthesizer 'cliché' sounds, but not a very good choice for simulating 'real' sounds. The typical clichéd sound is usually a 'synthy' sound consisting of slightly detuned oscillators beating against each other, with a resonant filter swept by a decaying envelope.

Digital synthesizers can deliberately introduce randomness, of course!

In contrast, digital synthesizers use discrete numerical representations of the audio and control signals. They are thus capable of reproducing prerecorded samples of real instruments with a very high fidelity. They also tend to be very precise and predictable, with none of the inherent uncertainty of analogue instruments. Some of the many digital synthesis techniques are described in Chapter 5. The difference between analogue and digital representations can be likened to an experiment to measure the traffic flow through a road junction. The actual passage of cars can be observed and the number of cars passing a specific point in a given time interval are noted down. The movement of the cars is analogue in nature since it is continuous, whereas the numbers are digital since they only provide numbers at specific times (Figure 3.2.1).

This link between a physical experiment and the numbers, which can be used to describe it, is also significant because the first analogue synthesizers, and in fact the first computers, were analogue not digital. An analogue computer is a device that is used to solve mathematical problems by providing an electrical circuit which behaves in the same way as a real system, and then

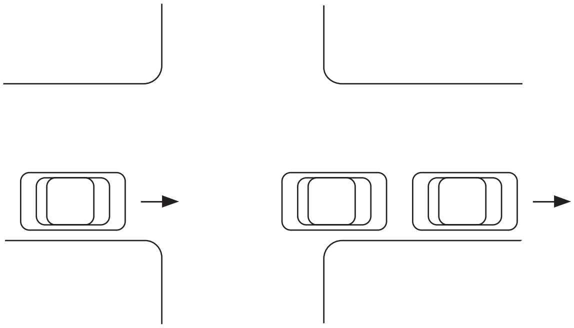


FIGURE 3.2.1 The movement of the cars is continuous or analogue, whereas the number of cars is discrete or digital.

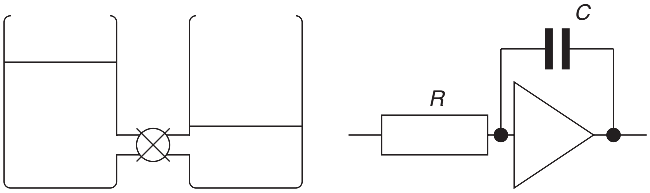


FIGURE 3.2.2 Two connected buckets can model an integrator circuit.

observing that happens when some of the parameters are changed. A simple example is what happens when two containers filled with water are connected together. This can be *modelled* by using an integrator circuit: a capacitor in a feedback loop (Figure 3.2.2). A step voltage applied to the integrator input simulates pouring water into one container – the voltage at the output of the integrator will rise steadily until the voltage is the same as the applied voltage, and then stops. If the integrator time constant is made larger, which is equivalent to reducing the flow of water between the containers (or making the second container larger), then the integrator will take longer to reach a steady state after a step voltage has been applied.

More sophisticated situations require more complex models, but the basic idea of using linear electronic circuits to simulate the behavior of real-world mechanical systems can be very successful. For more information on modelling techniques, see Section 5.3.

3.2.1 Voltage control

One of the major innovations in the development of the synthesizer was voltage control. Instead of providing mechanical control over many parameters that are used to set the operation of a synthesizer, voltages are used. Since the component parts of the synthesizer produce audio signals which are also voltages, the same signals which are used for audio can also be used for control purposes.

'Mechanical control' here means human-operated switches and knobs.

One example is an oscillator used for tremolo or vibrato modulation when used at a frequency of a few tens of hertz, but the same oscillator becomes a sound source itself if the frequency is a few hundred hertz.

Controlling a synthesizer with voltages requires some way of manipulating the voltages themselves, and for this voltage-controlled amplifiers (VCAs) are used. These use a control voltage (also known as CV) to alter the gain of the amplifier and can be used to control the gain of audio signals or CVs. Using VCAs means that a synthesizer can provide a single common gain control element. Although not all analogue synthesizers contain the same elements, many of the parts are common, and the method of control is the same throughout. Voltage control requires two main parts: sources and destinations.

Voltage control sources include the following:

- *Low-frequency oscillators (LFOs)*: These are required for vibrato, tremolo and other cyclic effects.
- *Envelope generators (EGs)*: These produce multi-segment CVs, where the time and slope of each segment can be controlled independently.
- *Pitch control*: Typically provided by a pitch wheel or lever, which provides a CV where the amount of pitch-bend is proportional to the voltage.
- *Keyboard control*: The output from a music keyboard provides a CV where the pitch is proportional to the voltage.
- *VCFs*: These can self-oscillate and so provide control signals.
- *VCOs*: These can be used as part of frequency modulation (FM) or ring modulation sounds.

Voltage-controlled destinations include:

- LFOs, where the voltage is used to control the frequency or the waveshape.
- EGs, where the voltages can be used to control the time or slopes of each of the segments.
- VCFs, where the voltage is used to control the cut-off frequency of the filter and perhaps the Q or resonance of the filter.
- VCOs, where the voltage is used to control the frequency of the oscillator, or sometimes the shape or pulse width of the output waveform.
- Voltage-controlled pan, where the voltage is used to control the stereo positioning of the sound.
- VCAs, where the voltage is used to control the gain of the amplifier.

Each of these modules will be explained in more depth in this chapter.

3.2.2 Tape and models

Not all analogue synthesizers have to be voltage controlled. The use of tape manipulation and real physical instruments to synthesize sounds might be regarded as the ultimate in 'analogue' synthesis, since it is actually possible to interact with the actual sounds directly and continuously. Despite this, the word 'analogue' usually implies the use of electronic synthesizers.

The 'source and modifier' model is often applied to analogue synthesizers, where the VCOs are the source of the raw audio, and the VCF, VCA and ADSR (attack decay sustain release) envelopes form the modifiers. But the same model can be applied to sample and synthesis (S&S) synthesizers or even to physical modelling. Even real-world musical instruments tend to have a source (for a violin, you vibrate the string using the bow) and modifier structure (for a violin it is the resonance of the body that gives the final 'tone' of the sound).

The controls of the sound source and the modifier can be split into two parts: performance controls which are altered during the playing of the instrument and fixed parameter controls which tend to remain unchanged whilst the instrument is being played (Figure 3.2.3).

Because it came first, many of the terminology, models and metaphors of analogue synthesis are reused in the more recent digital methods. Although this serves to improve the familiarity for anyone who has used an analogue synthesizer, it does not help a more conventional musician who has never used anything other than a real instrument.

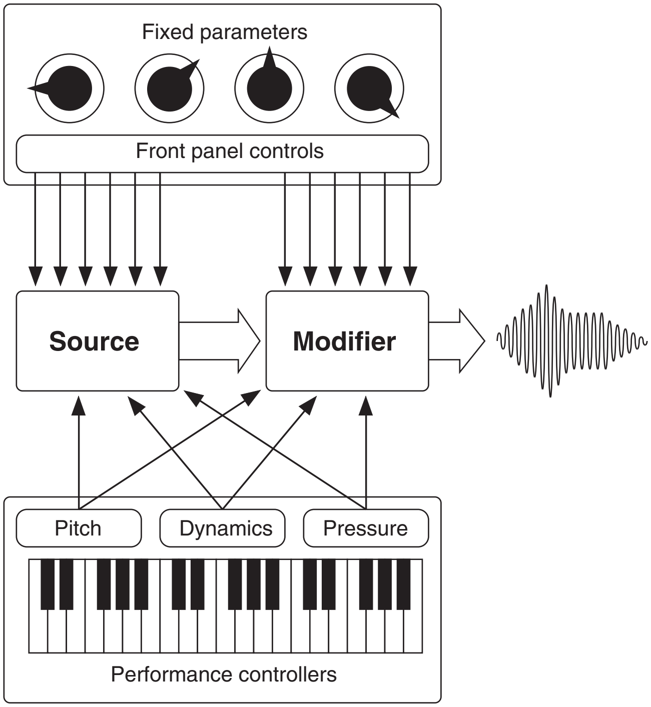


FIGURE 3.2.3
Performance controls are altered during the playing of the instrument, whilst fixed parameter controls normally remain unchanged.

3.3 Subtractive synthesis

Subtractive synthesis is often mistakenly regarded as the only method of analogue sound synthesis. Although there are other methods of synthesis, the majority of commercial analogue synthesizers use subtractive synthesis. Because it is often presented with a user interface consisting of a large number of knobs and switches, it can be intimidating to the beginner. Because there is often a one-to-one relationship between the available controls and the knobs and switches, it is well suited to educational purposes. It can also be used to illustrate a number of important principles and models that are used in acoustics and sound theory.

3.3.1 Theory: source and modifier

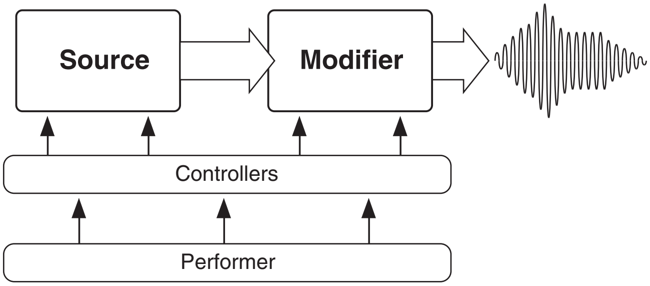
Subtractive synthesis is based around the idea that real instruments can be broken down into three major parts: a source of sound, a modifier (which processes the output of the source) and some controllers (which act as the interface between the performer and the instrument). This is most obviously apparent in many wind instruments, where the individual parts can be examined in isolation (Figure 3.3.1).

For example, a clarinet, where a vibrating reed is coupled to a tube, can be taken apart and the two parts can be investigated independently. On its own, the reed produces a harsh, strident tone, whilst the body of the instrument is merely a tube that can be shown to have a series of acoustic resonances related to its length, the diameter of the longitudinal hole and other physical characteristic; in other words, it behaves like a series of resonant filters. Put together, the reed produces a sound which is then modified by the resonances of the body of the instrument to produce the final characteristic sound of the clarinet.

Although this model is a powerful metaphor for helping to understand how some musical instruments work, it is by no means a complete or unique answer. Attempting to apply the same concept to an instrument such as a guitar is more difficult, since the source of the sound appears to be the plucked string, and the body of the guitar must therefore be the modifier of the sound produced by the string. Unfortunately, in a guitar, the source and the modifier are much more

FIGURE 3.3.1

The performer uses the instrument controllers to alter the source and modifier parameters.



closely coupled, and it is much harder to split them into separate parts. For example, the string cannot be played in isolation in quite the same way as the reed of a clarinet can, and all of the resonances of the guitar body cannot be determined without the strings being present and under tension.

Despite this, the idea of modifying the output of a sound source is easy to grasp and it can be used to produce a wide range of synthetic and imitative timbres. In fact, the underlying idea of source and modifier is a common theme in most types of sound synthesis.

3.3.2 Subtractive synthesis

Subtractive synthesis uses a subset of this generalized idea of source and modifier, where the source produces a sound that contains all the required harmonic content for the final sound, whilst the modifier is used to filter out any unwanted harmonics and shape the sound's volume envelope. The filter thus 'subtracts' the harmonics that are not required; hence the name of the synthesis method (Figure 3.3.2).

3.3.3 Sources

The sound sources used in analogue subtractive synthesizers tend to be based on mathematics. There are two basic types: waveforms and random. The waveforms are typically named after simple waveshapes: sawtooth, square, pulse, sine and triangle are the most common. The shapes are the ones which are easy to describe mathematically and also to produce electronically. Random waveshapes produce noise, which contains a constantly changing mixture of all frequencies.

The waveshapes in analogue synthesizers are only approximations to the mathematical shapes and the differences give part of the appeal of analogue sounds.

Oscillators are related to one of the component parts of analogue synthesizers: function generators. A function generator produces an output waveform, and this can be of arbitrary shape and can be continuous or triggered. An oscillator that is intended to be used in a basic analogue subtractive synthesizer normally produces just a few continuous waveshapes, and the frequency needs to be controlled by a voltage.

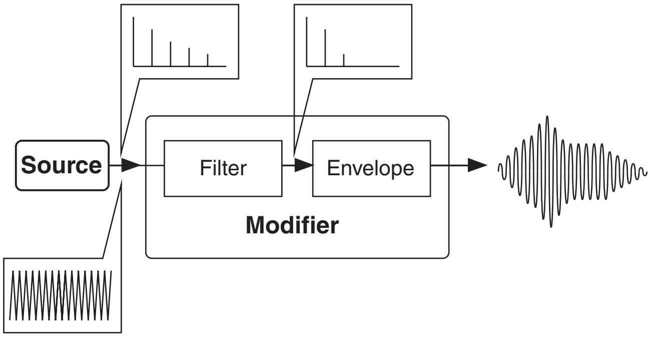


FIGURE 3.3.2
The source produces a constant raw waveform. The filter changes the harmonic structure, whilst the envelope shapes the sound.

It should also be noted that, in general, sources produce continuous outputs. You need to use a modifier in order to alter the timbre or apply an envelope to the sound.

VCOs

The VCOs provide voltage control of the frequency or pitch of their output. Some VCOs also provide voltage control inputs for modulation (usually FM) and for varying the shape of the output waveforms (usually the pulse width of the rectangular waveshape, although some VCOs allow the shape of other waveforms to be altered as well). Many VCOs have an additional input for another VCO audio signal, to which the VCO can be synchronized. Hard synchronization forces the VCO to reset its output to keep in sync with the incoming signal, which means that the VCO can only operate at the same or multiple frequencies of the input frequency. This produces a characteristic harsh sound. Other 'softer' synchronization schemes can be used to produce timbral changes in the output rather than locking of the VCO frequency.

A typical VCO has controls for the coarse (semitones) and fine (cents) tuning of its pitch, some sort of waveform selector (usually one of sine, triangle, square, sawtooth and pulse), a pulse width control for the shape of the pulse waveform and an output level control (Figure 3.3.3). Sometimes multiple simultaneous output waveforms are available, and some VCOs also provide 'sub-octave' outputs that are one or two octaves lower in pitch. A CV for the pulse width allows the shape of the pulse waveform (and sometimes other waveforms as well) to be altered. This is called pulse width modulation (PWM) or shape modulation.

One example: the Minimoog waveforms are arranged in the order of increasing harmonic content.

Harmonic content of waveforms

The ordering of waveforms on some early analogue synthesizers was not random. The waveforms are deliberately arranged so that the harmonic content increases as the rotary control is twisted.

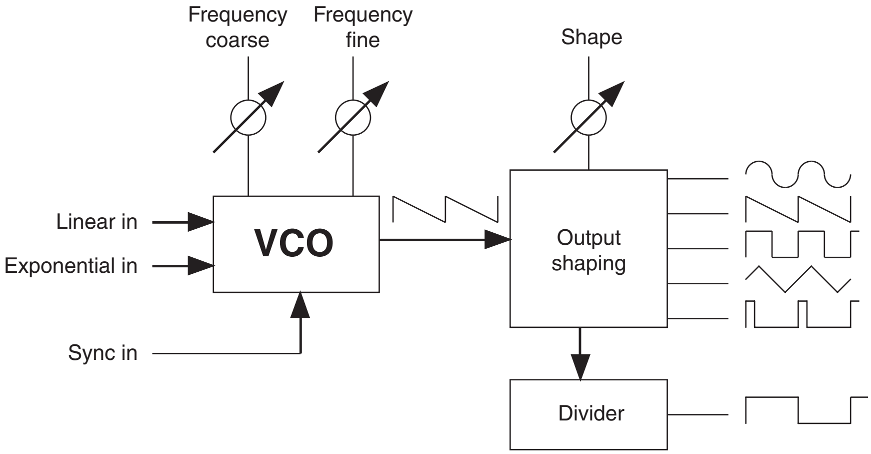


FIGURE 3.3.3
A block diagram of a typical VCO.

Arguably the simplest waveshape is the sine wave (Figure 3.3.4). This is a smooth, rounded waveform based on the mathematical sine function. A sine wave contains just one 'harmonic', the first or fundamental. This makes it somewhat unsuitable for subtractive synthesis since it has no harmonics to be filtered.

A triangle waveshape has two linear slopes (Figure 3.3.5). It has small amounts of odd-numbered harmonics, which give it enough harmonic content for a filter to work on.

A square wave contains only odd harmonics (Figure 3.3.6). It has a distinctive 'hollow' sound and a very synthetic feel.

A sawtooth wave contains both odd and even harmonics (Figure 3.3.7). It sounds bright, although many pulse waves can actually have more harmonic content. 'Super-sawtooth' waveshapes replace the linear slope with exponential slopes, as well as gapped sawtooths: these can contain greater levels of the upper harmonics than the basic sawtooth.

Depending on the ratio between the two parts (known as the mark–space ratio, shape, duty cycle or symmetry), pulse waveforms (Figure 3.3.8) can contain both odd and even harmonics, although not all of the harmonics are always present. The overall harmonic content of pulse waves increases as the pulse width narrows, although if a pulse gets too narrow, it can completely

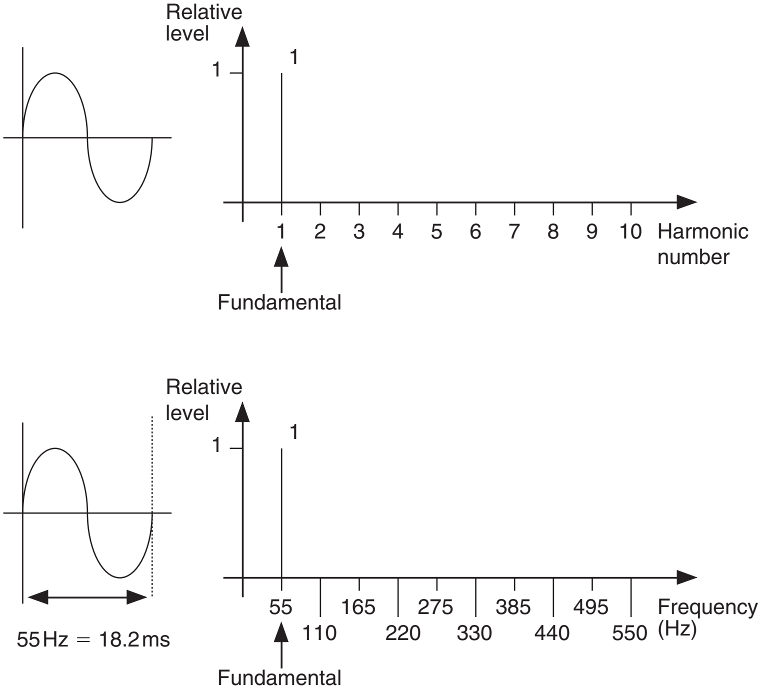


FIGURE 3.3.4
A sine waveform and harmonic spectrum and the same diagrams with actual frequencies shown.

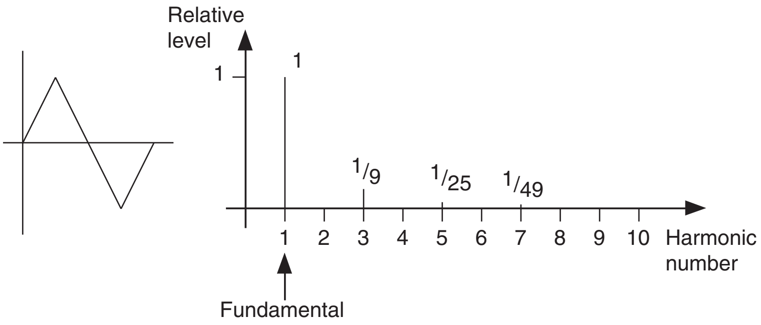
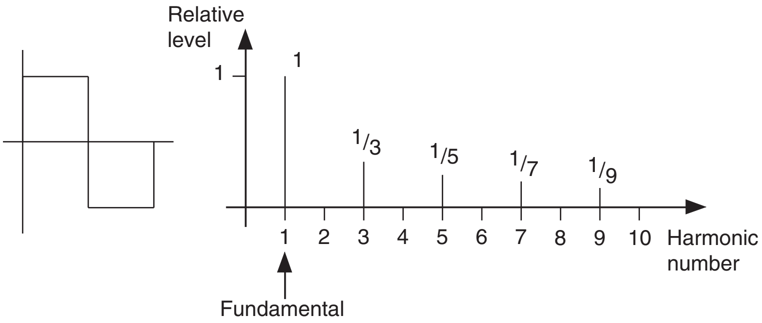


FIGURE 3.3.5
A triangle waveform and spectrum.



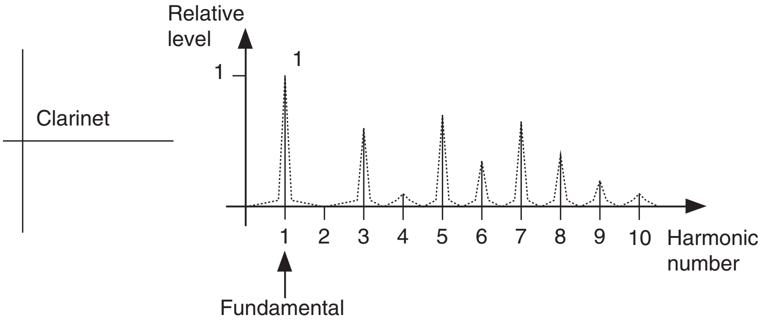


FIGURE 3.3.6
A square waveform and spectrum, with a typical clarinet spectrum for comparison.

disappear (the depth of PWM needs to be carefully adjusted to prevent this). A special case of a pulse waveshape is the 50:50 equal ratio square wave, where the even harmonics are not present. Pulse width modulated pulse waveforms are known as PWM waveforms and their harmonic content changes as the width of the pulse varies. PWM waveforms are normally controlled with LFO or an envelope, so that the pulse width changes with time. The audible effect when a PWM waveform is cyclically changed by an LFO is similar to two oscillators beating together.

It is possible to adjust the pulse width to give a square by ear: listening to the fundamental, the pulse width is adjusted until the note one octave up fades

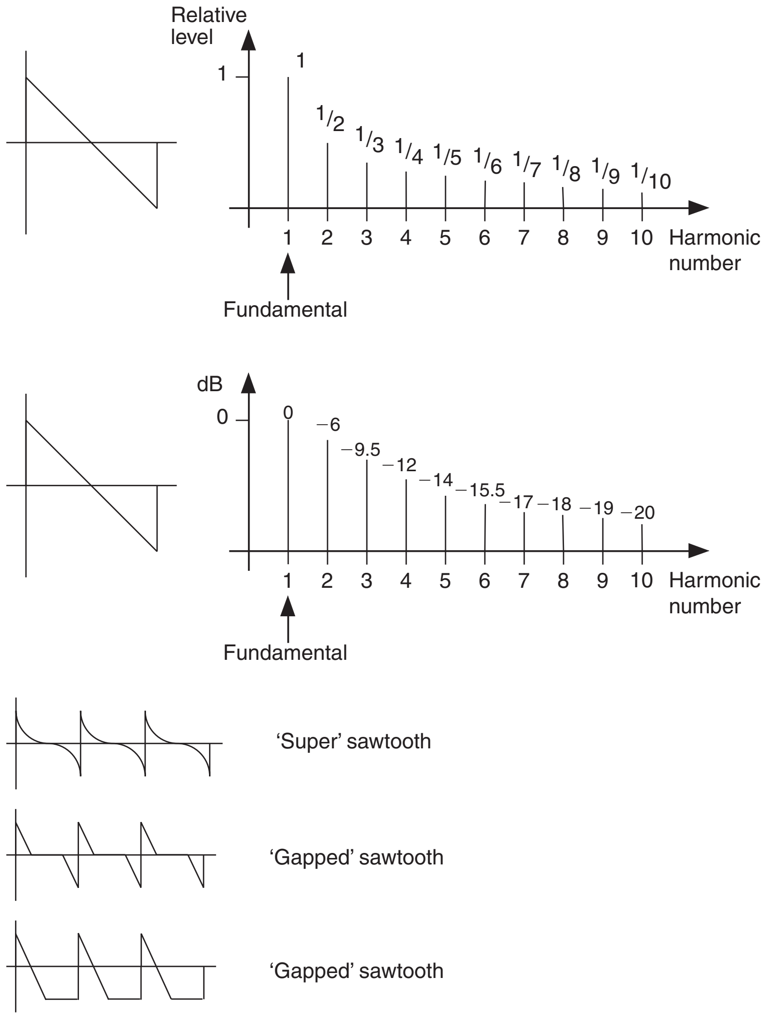


FIGURE 3.3.7

A sawtooth waveform and spectrum, with the spectrum also shown on a vertical decibel scale.

away. This note is the second harmonic and is thus not present in a square waveform. See also Figure 3.3.8.

All of the waveshapes and harmonic contents shown previously are idealized. In the real world the edges are not as sharp, the shapes are not so linear and the spectra are not as mathematically precise. Figure 3.3.9 shows a more realistic spectrum with dotted lines. This is a result of the filtering process used in producing the spectrum display and does not mean that there are extra frequencies present.

Although the waveshapes are based on mathematical functions, this does not always mean that they are all produced directly from mathematical formulas expressed in analogue electronics. For example, the 'sine' wave output on

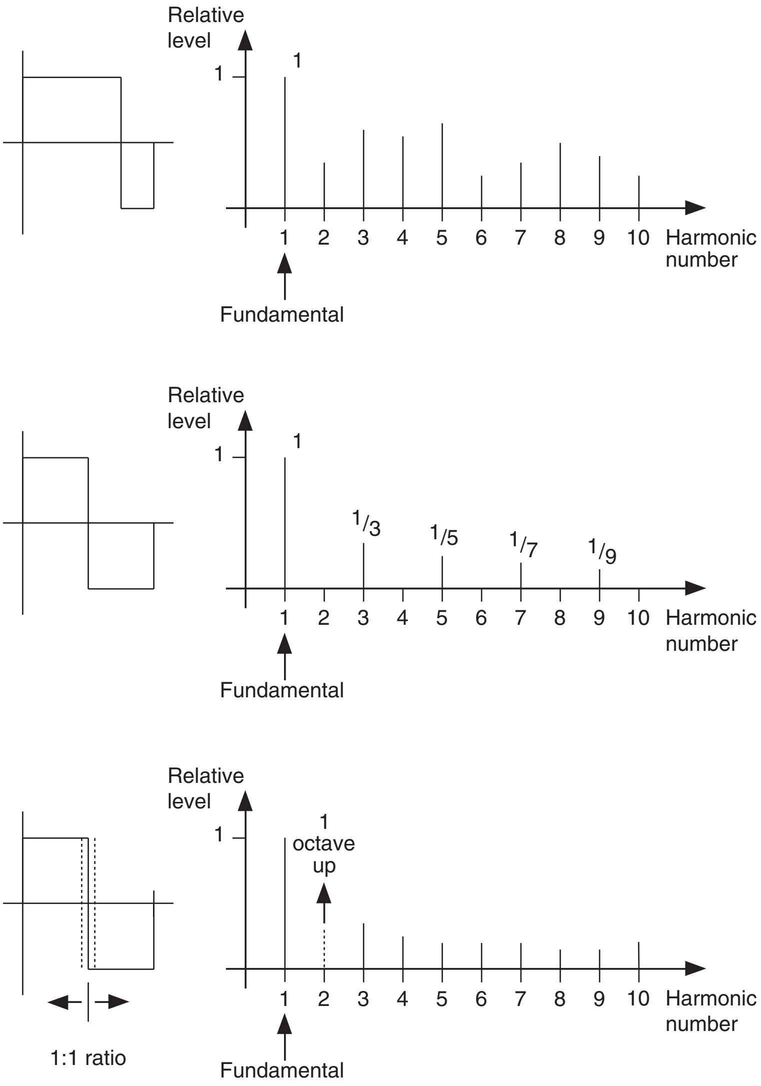


FIGURE 3.3.8
A pulse wave and spectrum. The relative levels of the harmonics depend on the width of the pulse.

many VCOs is produced by shaping a triangle wave through a non-linear amplifier which rounds off the top of the triangle so that it looks like a true sine wave (Figure 3.3.6). The resulting waveform resembles a sine wave, although it will have some additional harmonics – but for the purposes of subtractive synthesis, it is perfectly adequate. Section 3.4 on additive synthesis shows what real-world waveforms look like when they are constructed from simpler waveforms, rather than the perfect cases shown earlier.

3.3.4 Modifiers

There are two major modifiers for audio signals in analogue synthesizers: filters and amplifiers. Filtering is used to change the harmonic content or timbre

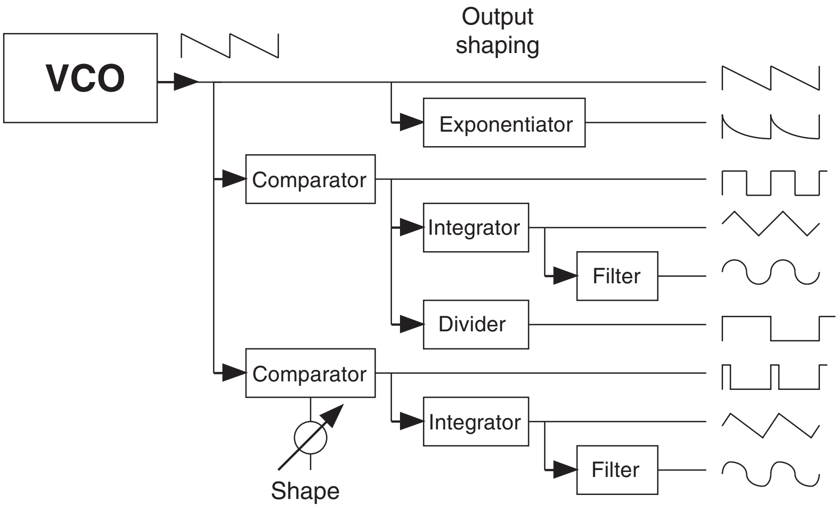


FIGURE 3.3.9 Analogue waveshaping allows the conversion of one waveform shape into others. In this example the sawtooth is the source waveform, although others are possible.

of the sound, whilst amplification is used to change the volume or 'shape' of the sound. Both types of modifiers are typically controlled by EGs, which produce complex CVs that change with time.

Effects such as reverb and chorus are not normally included as 'modifiers' in analogue synthesizers, although there are some notable exceptions: For instance, the EMS (Electronic Music Studios) VCS-3 has a built-in spring-line reverb unit.

3.3.5 Filters

A filter is an amplifier whose gain changes with frequency. It is usually the convention to have filters whose maximum gain is one, and so it is more correct to say that for a filter, the attenuation changes with frequency. A VCF is one where one or more parameters can be altered using a CV. Filters are powerful modifiers of timbre, because they can change the relative proportions of harmonics in a sound.

Filters come in many different forms. One classification method is based on the shape of the attenuation curve. If a sine wave test signal is passed through a filter, then the output represents the attenuation of the filter at that frequency; this is called the frequency response of the filter. An alternative method injects a noise signal into the filter and then monitors the output spectrum, but the sine wave method is easier to carry out. The major types of frequency response curve are

- low-pass
- band-pass
- high-pass
- notch.

In general, analogue synthesizer filters have two or four poles, whilst digital filters can have up to eight or more.

Low-pass

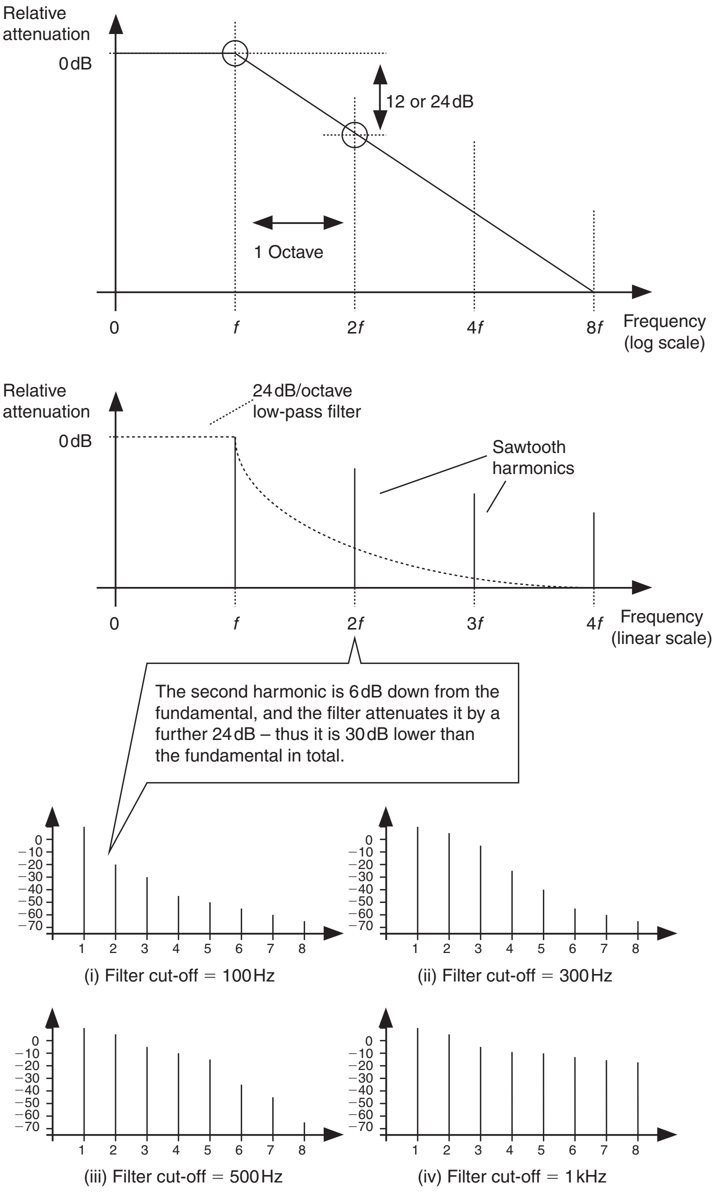
A low-pass filter has more attenuation as the frequency increases. The point at which the attenuation is 3 dB is called the cut-off frequency, since this is the frequency at which the attenuation first becomes apparent. It is also the point at which half of the power in the audio signal has been lost and so it is sometimes called the half-power point. Below the cut-off frequency, a low-pass filter has no effect on the audio signal and it is said to have a flat response (the attenuation does not change with frequency). Above the cut-off frequency, the attenuation increases at a rate which is called a slope. The slope of the attenuation varies with the design of the filter. Simple filters with one resistor and capacitor (RC) will have slopes of 6 dB/octave, which means that for each doubling of frequency, the attenuation increases by 6 dB. Each pair of RC elements is called a pole and the slope increases as the number of poles increases. A two-pole filter will have an attenuation of 12 dB/octave, whilst a four-pole filter will have 24 dB/octave. Audibly, a four-pole filter has a more 'synthetic' tone and makes much larger changes to the timbre of the sound as the cut-off frequency is changed. A two-pole filter is usually associated with a more 'natural' sound and more subtle changes to the timbre (Figure 3.3.10).

Low-pass VCFs usually have the cut-off frequency as the main controlled parameter. A sweep of cut-off frequency from high to low frequencies makes any audio signal progressively 'darker', with the lower frequencies emphasized and less high frequencies present. A filter sweeping from high frequency to low frequency of cut-off is often referred to as changing from 'open' to 'closed'. When the cut-off frequency is set to maximum, and the filter is 'open', then all frequencies can pass through the filter.

As the cut-off frequency of a low-pass filter is raised from zero, the first frequency that is heard is usually the fundamental. As the frequency rises, each of the successive harmonics (if any) of the sound will be heard. The audible effect of this is an initial sine wave (the fundamental), followed by a gradual increase in the 'brightness' of the sound as any additional frequencies are allowed through the filter. If the cut-off frequency of a low-pass filter is set to allow just the fundamental to pass through the filter, then the resulting sine wave will be identical for any input signal waveform. It is only when the cut-off frequency is increased and additional harmonics are heard, the differences between the different waveforms will become apparent. For example, a sawtooth will have a second harmonic, whilst a square wave will not.

High-pass

A high-pass filter has the opposite filtering action to a low-pass filter: it attenuates all frequencies that are below the cut-off frequency. As with the low-pass VCF, the primary parameter that is voltage controlled is the cut-off frequency. High-pass filters remove harmonics from a signal waveform, but as the frequency is raised from zero, it is the fundamental which is removed first. As additional harmonics are removed, the timbre becomes 'thinner' and brighter,



The second harmonic is 6 dB down from the fundamental, and the filter attenuates it by a further 24 dB – thus it is 30 dB lower than the fundamental in total.

FIGURE 3.3.10 Filter responses are normally shown on a log frequency scale since a dB/octave cut-off slope then appears as a straight line. But harmonics are based on linear frequency scales and on these graphs the filter appears as a curve. Low-pass filtering a sawtooth waveform with the cut-off frequency set to four different values:

(i) At 100 Hz, the filter cut-off frequency is the same as the fundamental frequency of the sawtooth waveform. The second harmonic is 30 dB below the fundamental and so the ear will hear an impure sine wave at 100 Hz.

(ii) At 300 Hz, the first three harmonics are in the pass-band of the filter and the output will sound considerably brighter.

(iii) At 500 Hz, the first five harmonics are in the filter pass-band, and so the output will sound like a slightly dull sawtooth waveform.

(iv) At 1 kHz, the first ten harmonics are all in the pass-band of the filter and the output will sound like a sawtooth waveform.

with less low-frequency content and more high-frequency content, and the perceived pitch of the sound may change because the fundamental is missing. Some subtractive synthesizers have a high-pass (not voltage-controlled) filter connected either before or after the low-pass VCF in the signal path. This allows limited additional control over the low frequencies that are passed by the low-pass filter. It is usually used to remove or change the level of the fundamental, which is useful for imitating the timbre of instruments where the fundamental is not the largest frequency component.

Band-pass

A band-pass filter only allows a set range of frequencies to pass through it unchanged – all other frequencies are attenuated. The range of frequencies that are passed is called the bandwidth, or more usually, the pass-band, of the filter. Band-pass VCFs usually have control over the cut-off frequency and the bandwidth.

Band-pass (and notch) filters are the equivalent of the resonances that happen in the real world. A wine-glass can be stimulated to oscillate at its resonant frequency by running a wet finger around the rim.

A band-pass filter can be thought of as a combination of a high-pass and a low-pass filters, connected in series, one after the other in the signal path. By using the same CV to the cut-off frequency inputs of two VCFs (one high-pass and the other low-pass), the cut-off frequencies will 'track' each other and the effective bandwidth of the band-pass filter will stay constant as the cut-off frequencies are changed. The width of the band-pass filter's pass-band can be controlled by adding an extra CV offset to one of the filters. If the cut-off frequency of the low-pass filter is set below that of the high-pass filter, then the pass-band does not exist, and no frequencies will pass through the filter (Figure 3.3.11).

Band-pass filters are often described in terms of the shape of their pass-band response. Narrow pass-bands are referred to as 'narrow' or 'sharp', and they produce marked changes in the frequency content of an audio signal. Wider pass-bands have less effect on the timbre, since they merely emphasize a range of frequencies. The middle frequency of the pass-band is called the center frequency.

Very narrow band-pass filters can be used to examine a waveform and determine its frequency content. By sweeping through the frequency range, each harmonic frequency will be heard as a sine wave when the center frequency of the band-pass filter is the same as the frequency of the harmonic (Figure 3.3.12).

Notch

A notch filter is the opposite of a band-pass filter. Instead of passing a band of frequencies, it attenuates just those frequencies and allows all others to pass through unaffected. Notch filters are used to remove or attenuate specific ranges of frequencies and narrow 'notches' can be used to remove single harmonic frequencies from a sound. Notch VCFs usually provide control over both the cut-off and the bandwidth (or 'stop-band') of the filter (Figure 3.3.13).

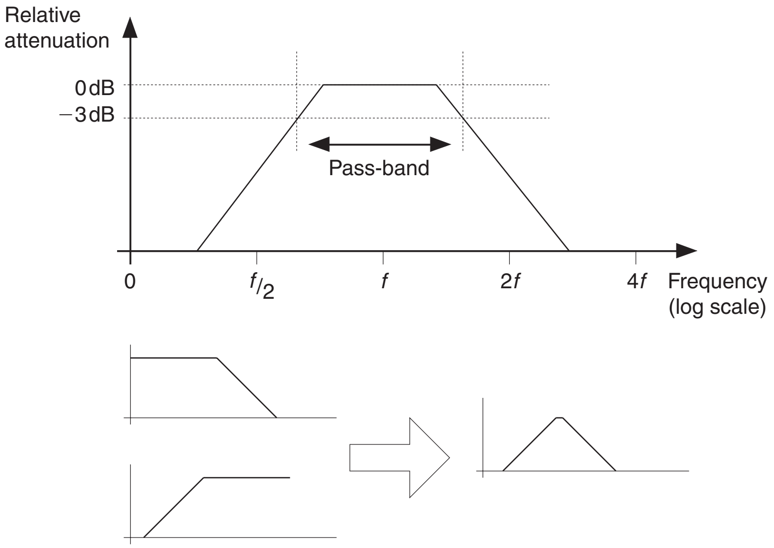


FIGURE 3.3.11 A band-pass filter only passes frequencies in a specific range. This is normally the two points at which the filter attenuates by 3dB. It can be thought of as a low-pass and a high-pass filter connected in series (one after the other). In the example shown, the lower cut-off frequency is about 0.6*f* (for the high-pass filter), whilst the upper cut-off frequency is about 1.6*f* (for the low-pass filter). The bandwidth of the filter is the difference between these two cut-off frequencies. Small differences are referred to as 'narrow', whilst large differences are known as 'wide'.

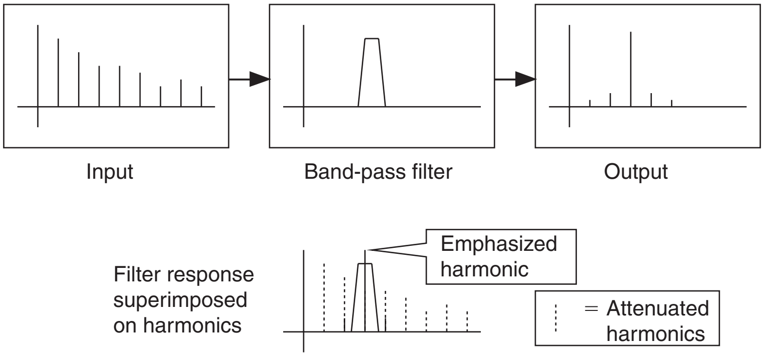


FIGURE 3.3.12 If a narrow band-pass filter is used to process a sound that has a rich harmonic content, then the harmonics which are in the pass-band of the filter will be emphasized, whilst the remainder will be attenuated. This produces a characteristic resonant sound. If the band-pass filter is moved up and down the frequency axis, then a characteristic 'wah-wah' sound will be heard – this is sometimes used on electric guitar sounds.

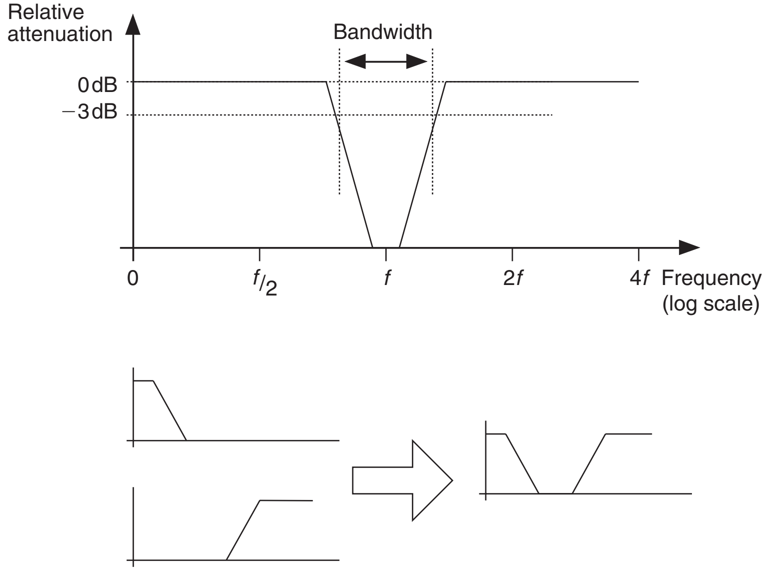


FIGURE 3.3.13 A notch filter is the opposite of a band-pass filter, which it attenuates a band of frequencies. It can also be formed from a series combination of a low- and a high-pass filters, provided that the low-pass cut-off frequency is lower than the high-pass cut-off frequency. If not, then no notch will be present.

Scaling

If the keyboard pitch voltage is connected to the cut-off frequency CV input of a VCF, then the cut-off frequency can be made to track the pitch being played on the keyboard. This means that any note played on the keyboard is subjected to the same relative filtering, since the cut-off frequency will follow the pitch being played. This is called pitch tracking or keyboard scaling (Figure 3.3.14).

Resonance

Low-pass and high-pass filters can have different response curves depending on a parameter called resonance or *Q* (short for 'quality', but rarely referred to as such). Resonance is a peaking or accentuation of the frequency response of the filter at a specific frequency. For band-pass filters, the *Q* figure is given by the formula:

$$Q = \text{Center frequency/Bandwidth (or pass-band)}$$

This formula is often also used for the resonance in the low-pass and high-pass filters used in synthesizers. For these low-pass and high-pass filters, the resonance is usually at the cut-off frequency and it forms a 'peak' in the frequency response (Figure 3.3.15).

In many VCFs, internal feedback is used to produce resonance. By taking some of the output signal and adding it back into the input of the filter, the

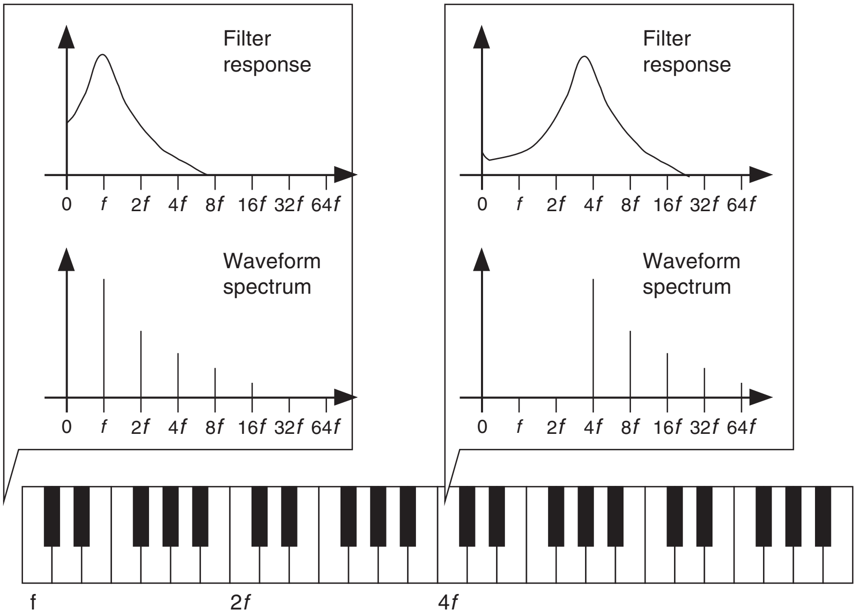


FIGURE 3.3.14 Filter scaling, tracking or following is the term used to describe changing the filter cut-off so that it follows changes in the pitch of a sound. This allows the spectrum of the sound produced to stay the same. In the example shown, the filter peak tracks the changes in the pitch of the sound when two notes two octaves apart are played – the peak coincides with the fundamental frequency in each case. With no filter scaling then the note with a fundamental of 4*f* two octaves up would be strongly attenuated if the filter cut-off frequency did not change from the peak at a frequency of *f*.

response of the filter can be emphasized at the cut-off frequency. This also means that the resonance of the filter can be made voltage controllable by varying the amount of feedback with a VCA. See Section 3.3.5 for more on VCAs and see Section 3.6 for more information on the implementation of filters.

Most subtractive synthesizers implement only low-pass and band-pass filtering, where the band-pass is often produced by increasing the *Q* of the low-pass filter so that it is a 'peaky' low-pass rather than a true band-pass filter. This phenomenon of a peak of gain in an otherwise low-pass (or high-pass) response is called 'corner peaking'. Some models of analogue synthesizer also have an additional simple high-pass filter, whilst notch filters or band-rejects are very uncommon.

There are two types of filters: constant-*Q* and constant bandwidth. Constant-*Q* filters do not change their *Q* as the frequency of the filter is changed. This means that they are good for applications where the filter is used to produce a sense of pitch from an unpitched source such as noise. Since the *Q* is constant, the bandwidth varies with the filter frequency and so sounds 'musical'. Constant-bandwidth filters have the same bandwidth regardless of the filter frequency. This means that a relatively narrow bandwidth of 100 Hz

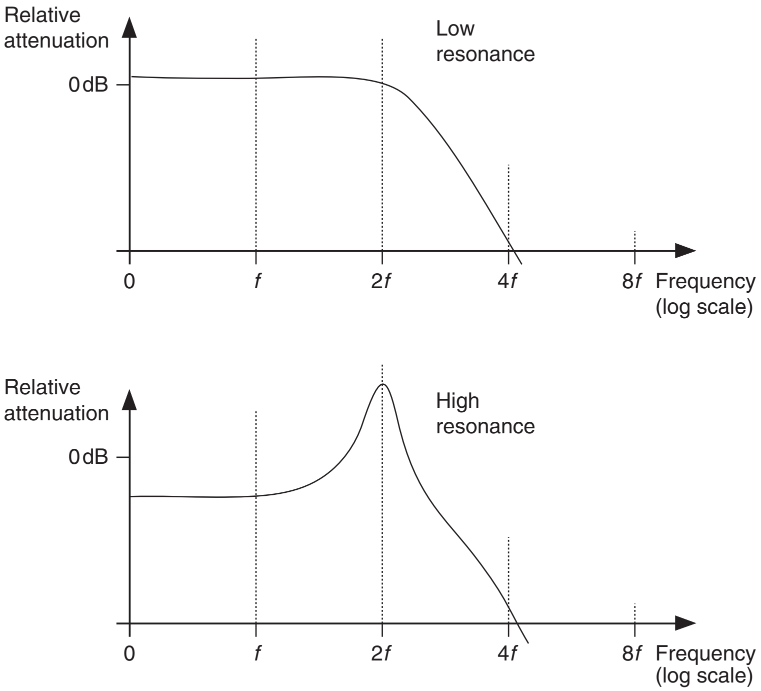


FIGURE 3.3.15 Resonance changes the shape of a low-pass filter response most markedly at the cut-off frequency. The result is a smooth and continuous transition from a low-pass to something like a narrow band-pass filter.

for a filter frequency of 4 kHz, is very wide for a 400-Hz frequency: the *Q* of a constant-bandwidth filter changes with the filter frequency. Most analogue synthesizer filters are constant-*Q*.

The effect of changing the cut-off frequency of a highly resonant low-pass filter in 'real time', with a source sound rich in harmonics, is quite distinctive and can be approximated by singing 'eee-yah-oh-ooh' as a continuous sweep of vowel sounds.

Filter oscillation

If the resonance of a peaky low-pass or a band-pass VCF is increased to the point at which the filter plus its feedback has a cumulative gain of more than one at the cut-off frequency, then it will break into self-oscillation. In fact, this is one method of producing an oscillator – you put a circuit with a narrow band-pass frequency response into the feedback loop of an amplifier or operational amplifier (op-amp) (Figure 3.3.16). The oscillation produces a sine wave, sometimes much purer than the 'sine' waves produced by the VCOs!

3.3.6 Envelopes

An envelope is the overall 'shape' of the volume of a sound, plotted against time (Figure 3.3.17). In an analogue synthesizer, the volume of the sound output at any time is controlled by a voltage-controlled amplifier (see VCA) and

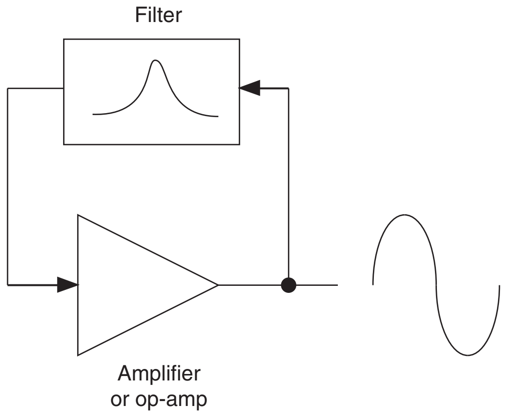


FIGURE 3.3.16 If a filter with a strong resonant peak in its response is connected around an amplifier, then the circuit will tend to oscillate at the frequency with the highest gain – at the peak of the filter response. This can be easily demonstrated (perhaps too easily) with a microphone and a PA system.

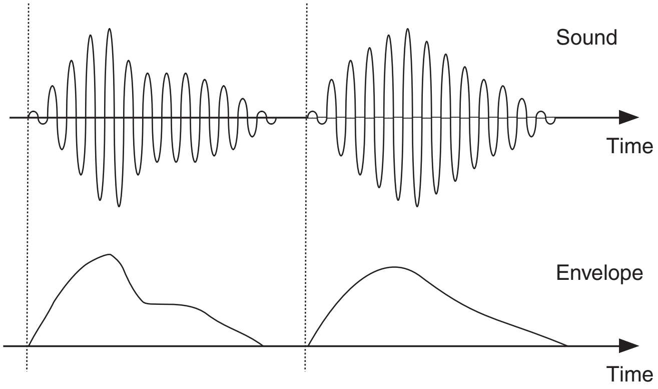


FIGURE 3.3.17 The 'envelope' of a sound is the overall shape – the change in volume with time. The shape of an envelope often forms a distinctive part of a sound.

the voltage that is used is called an envelope. Envelopes are produced by 'EGs' and have many variants. EGs are categorized by the number of controls which they provide over the shape of the envelope. The simplest provide control only over the start and end of a sound, whilst the most complex may have a very large number of parameters.

Envelopes are split into segments or parts (Figure 3.3.18). The time from silence to the initial loudest point is called the attack time, whilst the time for the envelope to decrease or decay to a steady value is called the decay time. For instruments that can produce a continuous sound, such as an organ, the decay time is defined as the time for the sound to decay to the steady-state 'sustain'

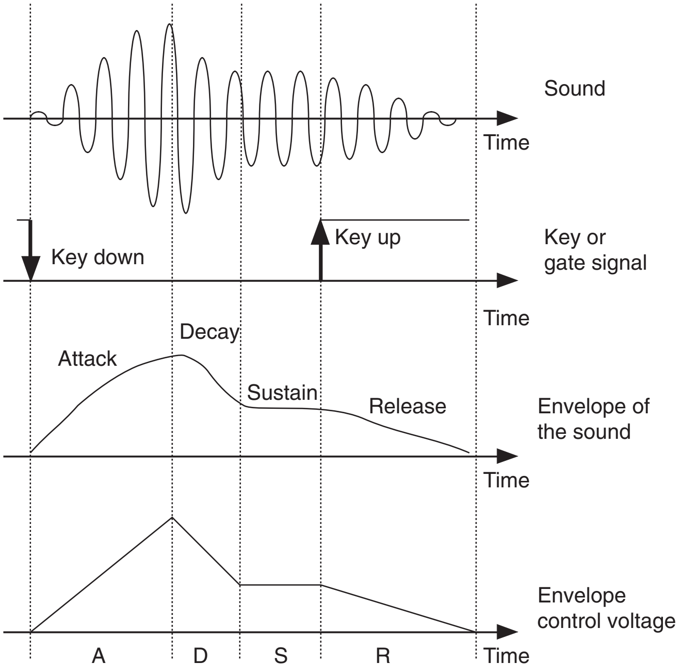


FIGURE 3.3.18 Envelopes are divided into segments depending on their position. The start of the sound is called the 'attack segment'. After the loudest part of the sound, the fall to a steady 'sustain' segment is called the 'decay' segment. When the sound ends, the fall from the sustain segment is called the 'release' segment.

level, whilst the time that it takes for the sound to decay to silence when it ends is called the release time.

Bowed stringed instruments can have long attack, decay and release times, whilst plucked stringed instruments have shorter attack times and no sustain time. Pianos and percussion instruments can have very fast attack times and complex decay/sustain segments. There is an almost standardized set of names for the segments of envelopes in analogue synthesizers, which contrasts with the more diverse naming schemes used in digital synthesizers.

Envelopes are usually referred to in terms of the CV that they produce, and it is normally assumed that they are started by a key being pressed on a keyboard. Envelopes can be considered to be sophisticated time-based function generators with manual key triggering.

The following are some of the common types of EGs.

Attack release

Attack release (AR) envelopes only provide control over the start and end of a sound (Figure 3.3.19). The two-segment envelope CV, which is produced, rises up to the maximum level and then falls back to the quiescent level,

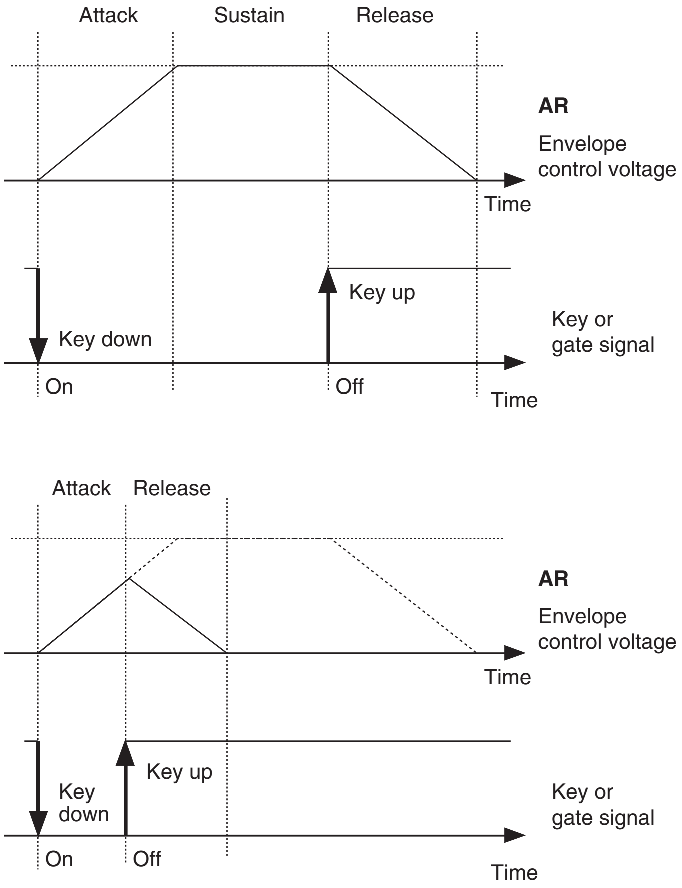


FIGURE 3.3.19 In an AR envelope the pressing down of a key (or a similar gating device on a synthesizer that does not use keys) starts the attack segment. When the peak level has been reached, then the envelope stays at this level until the key is released (of the gating signal is removed) and the envelope falls in the release segment. If the key is released whilst the envelope is in the attack segment, then the envelope normally moves to the release segment, and need not reach the peak level (see also Figure 3.3.27). Some synthesizers provide a control which forces the whole of the attack segment to be completed.

which is usually 0 volts. AR envelopes are often found on 1970s vintage string machines: simple polyphonic keyboards that used organ 'master oscillator and divider' technology with simple filtering and chorus effects processing to give an emulation of an orchestral string sound (see Section 3.4 for more information).

Attack decay

If the envelope moves into the decay segment as soon as the attack segment has reached its maximum level, then the decay time sets how long it takes for the envelope to drop to zero. This means that only percussive (non-sustaining) envelopes can be produced (unless the decay time is set to be very long, as in some attack decay release (ADR) envelopes). These two-segment attack decay (AD) envelopes (Figure 3.3.20) are often found connected to the frequency control input of VCOs, where the envelope then produces a rapid change in pitch at the start of the note, known as a 'chirp'. This can be effective for vocal and brass sounds. Inverting the envelope can produce changes downwards in pitch instead of upwards.

Attack decay release

The ADR envelope uses long decay times to simulate a high sustain level, in which case the resulting envelope is very much like an AR envelope, or else a percussive AD envelope by using shorter decay times (Figure 3.3.21).

Attack decay sustain

If a sustain level is added to an AD envelope, then the attack decay sustain (ADS) EG is the result (Figure 3.3.22). The attack segment reaches a maximum

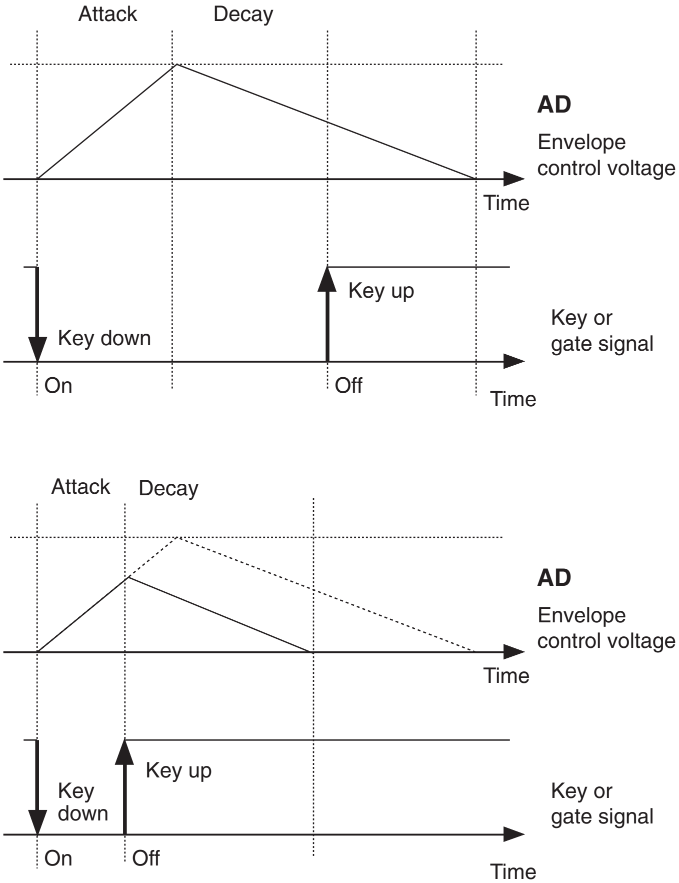


FIGURE 3.3.20
An AD envelope is similar to an AR envelope, except that there is no sustain segment. When the peak level is reached, the envelope decays, even if the key is held down.

value and the decay time then sets how long it takes for the envelope to reach the sustain level. Some ADS EGs have switches that make the release time the same as the decay time or else have a very short release time. The type of envelope that is produced depends on the sustain level. If the sustain level is set to the maximum level (the same as the attack reaches), then two-segment AR-type envelopes are produced. If the sustain level is set to zero, then only two-segment AD envelopes are produced. With the sustain level set mid-way, then four-segment ADSR-type envelopes can be produced. If these have an initial attack and decay portion, then the sustain portion whilst the key is held down and then a release portion when the key is released.

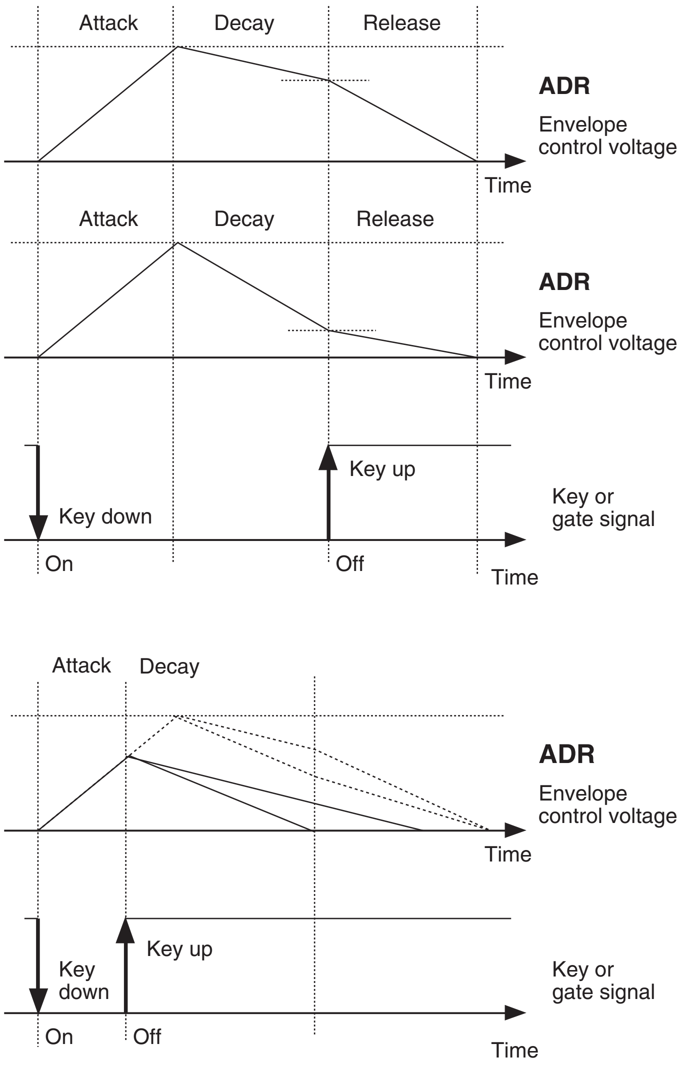


FIGURE 3.3.21
The ADR envelope provides control over separate decay and release segments. This allows more complex envelope shapes to be produced than is possible with AR or AD EGs. If the key or gate is released during the attack segment, then the envelope moves to the release segment and ignores the decay segment.

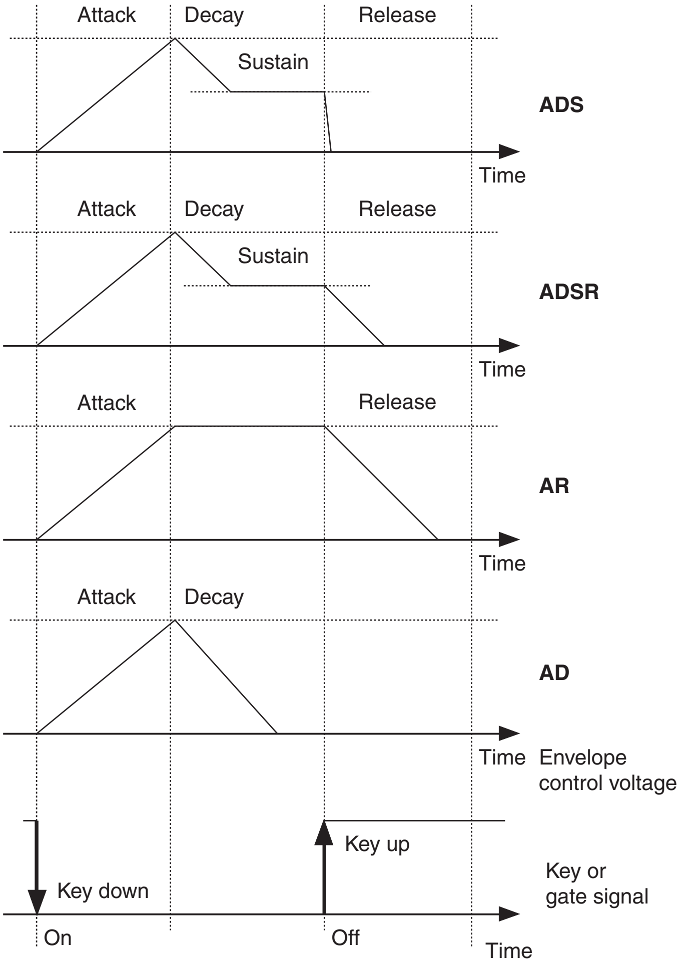


FIGURE 3.3.22 An ADS envelope adds a sustain segment at the end of the decay segment. The 'release' time is normally set to the same as the decay time, although some synthesizers provide a switch which forces a fast release time regardless of the setting of the decay time. An ADS EG can be used to produce a wide variety of envelopes, including the ones which have many of the characteristics of ADSR (see later), AR and AD envelopes.

Attack decay sustain release

The most widely adopted EG is probably the ADSR (Figure 3.3.23). With just four controls, it is capable of producing a wide variety of envelope shapes; with only the attack decay 1 break decay 2 release (ADBDR) dual-decay variant offering superior flexibility at the cost of one extra control. The ADSR EG's main weakness is that the sustain segment is static, it is a fixed level. For this reason, ADSR-type envelopes are not particularly well suited in producing percussive piano-type envelopes, where the 'sustain' portion of the sound gradually decays to zero. See ADBDR envelope later for a better alternative.

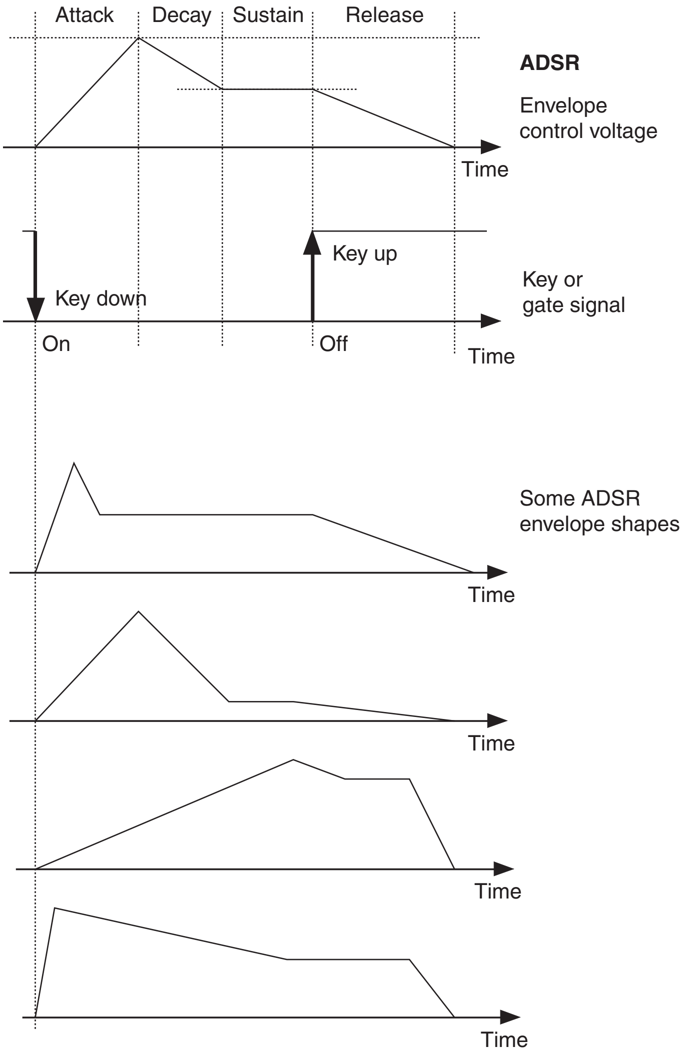


FIGURE 3.3.23 The ADSR envelope adds a separate control for the release time. This provides enough flexibility to produce a large number of envelopes with a small number of controls and the ADSR envelope is widely used in synthesizers.

Attack hold decay sustain release

Some envelopes force the envelope to stay at the maximum or peak level for a fixed time when the attack segment has finished and before the decay segment can start (Figure 3.3.24). These are called attack hold decay sustain release (AHDSR) envelopes. This is useful when a percussive envelope is set with very rapid attack and decay times, and the minimum length of the envelope needs to be controlled. For some sounds, an AD envelope with fast times (less than 10 ms) can be too short to be audible.

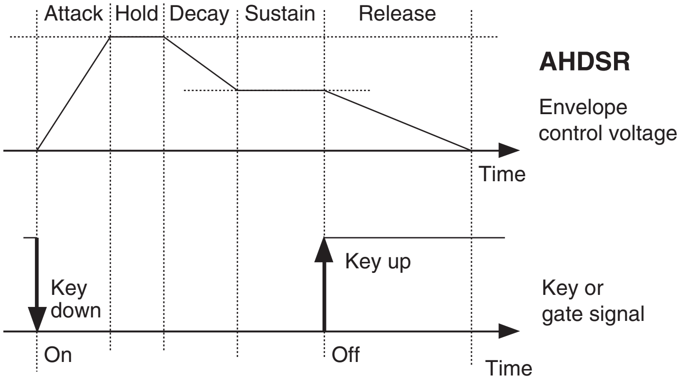


FIGURE 3.3.24 An AHDSR envelope adds a 'hold' segment at the end of the attack segment, rather like the sustain segment, but the length is set by a time rather than when the key or gate is released. As with other envelope shapes, if the key is released before the sustain segment, then the envelope moves to the release segment.

A variation on the hold segment being after the 'attack' segment of the envelope is the attack decay hold release (ADHR) envelope, where the 'sustain' segment is only held up to a specific time, after which it begins to decay. This is arguably better suited to percussive and piano sounds than the ADSR.

Attack decay 1 break decay 2 release

By splitting the decay segment into two portions, with a 'break-point' level controlling when one decay portion finishes and the other starts, a wide range of envelope shapes can be produced (Figure 3.3.25). By setting the second decay to a very long time, it can be used in much the same way as a sustain segment, although it has the advantage that it can still decay away slowly. This is arguably a better emulation of real-world envelopes for instruments such as pianos, where the sustain segment is actually a long decay time. In some implementations of ADBDR envelopes, this second decay is called the 'slope' segment to distinguish it from the decay segment.

Advanced EGs

There are many sophisticated enhancements of the basic analogue ADSR EG (Figure 3.3.26). Most of these are ADSRs with the addition of initial time delay, break-points in the attack or decay segments and times for the peak and sustain levels. Although the extra controls provide more possibilities for envelope shapes, they also greatly increase the complexity of the user interface. Delayed envelopes (denoted by an initial 'D' in the abbreviation: DADSR for delayed ADSR) are used when the start of the envelope needs to be delayed in time without the need for using a long attack time, or where the attack needs to be rapid after the delay time.

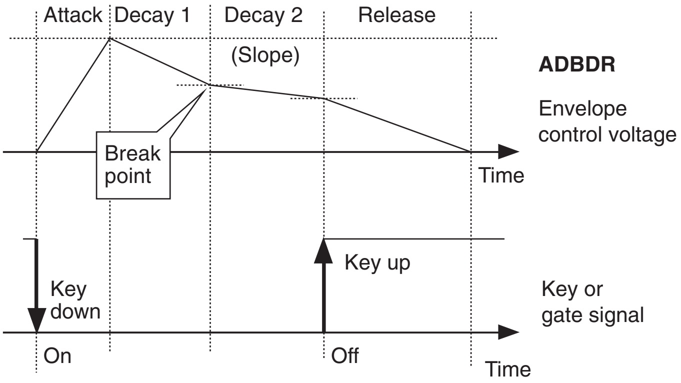


FIGURE 3.3.25 The ADBDR envelope has two decay segments and the transition from one decay is set by a variable level control, rather like a sustain level control. By setting the decay time to a long value, they can be used as pseudo-sustain segments, and so an ADBDR envelope can produce similar envelopes to an ADSR type.

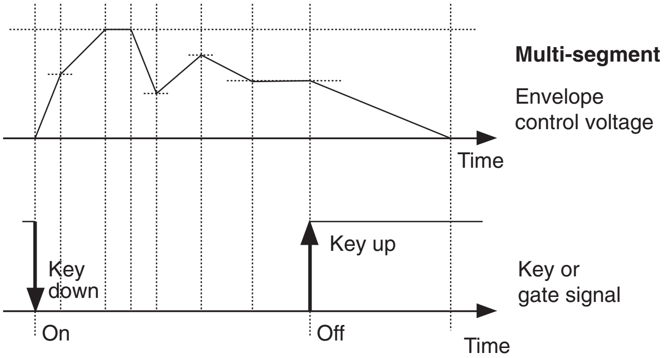


FIGURE 3.3.26 Multi-segment envelopes can have several attack, decay and release segments, as well as hold and sustain segments. Break-points can also be used to split a segment into smaller segments.

Some of these EGs provide a break-point in the attack segment, so that two different attack times can be controlled. This is especially useful for long attack times, where the start of the audio signal is too quiet to be heard, and the initial portion of the attack segment is heard as a delay. By having a rapid rise to a level where the audio signal is audible, followed by a slower second attack portion, this unwanted apparent delay can be avoided. This extra break-point is also useful for simulating more complicated attack curves.

Break-points are not always explicitly named as such. The interaction between the gate signal and the envelope often has implied break-points at the transitions between attack, decay, sustain and release. These are frequently not documented in the manufacturer's product information. The usual method of operation is shown in Figure 3.3.27. If the key is only held down for a short time,

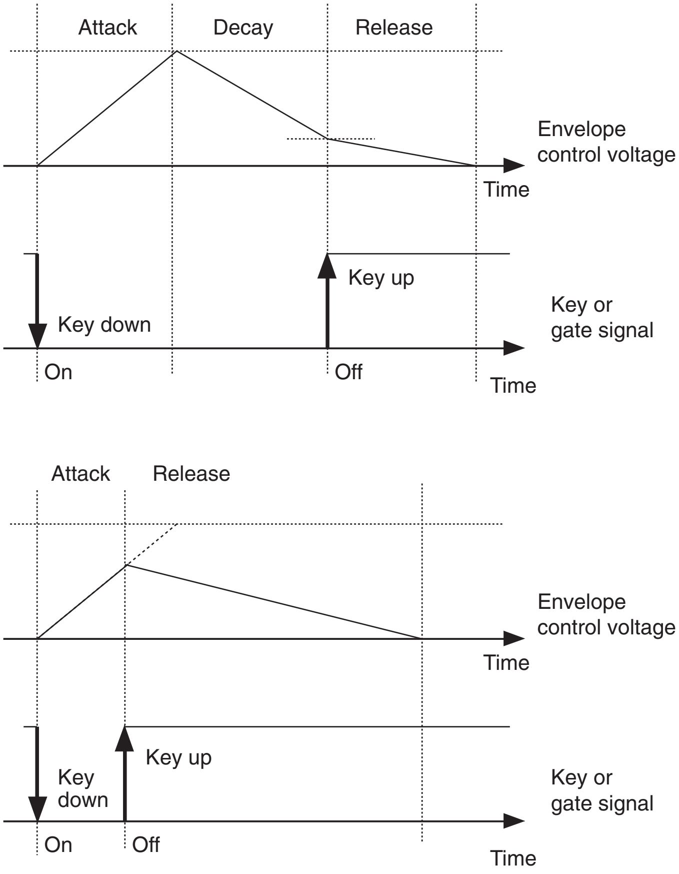


FIGURE 3.3.27 The transition from the attack segment to the release segment when the key or gate is released can be thought of as adding in a break-point to the attack segment.

and the envelope is still in the attack segment when the key is released, then the envelope will go into the release segment. In this case the envelope may not reach the maximum level, although some EGs always rise to the maximum level. If there is a hold time associated with the maximum level, then this is usually not affected by the key being released. If the envelope has reached the decay segment, then when the key is released, the envelope will go into the release segment.

If the initial, final, peak and sustain levels are all controllable, then the envelope flexibility can become approximately equivalent to the multi-segment envelopes often found in digital synthesizers, although the terminology is normally very different. See Chapter 5 for more details on digital envelopes.

Some analogue synthesizers only have one EG, which is then used to control both the VCF and VCA. If two envelopes are available, then patching one to the filter and the other to the amplifier provides independent control over the volume and timbre. A third envelope could be used to control the pitch of the VCOs or perhaps the stereo position of the sound using two VCAs arranged as a pan control.

Linear or exponential?

Many real-world quantities change in a non-linear way. This can be due to the process involved or the way that the change is perceived. For example, the theoretical population growth curve of many animal species shows an exponential or power-law growth because the initial two animals produce two new individuals, who then eventually join the breeding population, and then these four individuals produce four new offsprings. The doubling of the population in each successive generation produces a rapidly increasing population curve. Conversely, because human ears perceive sound in a non-linear way, each doubling of the apparent volume level requires about 10 times the energy in the sound. Again, the relationship connecting the two variables is a non-linear one.

Many natural sound envelopes have non-linear curves. Changes are usually rapid at first and gradually slow down (Figure 3.3.28). This is particularly apparent with the attack segment of envelopes, where a linear rise in volume sounds too slow at first, whereas an exponential rise in volume sounds 'correct' – in fact, it sounds 'linear' to the human ear! Some EGs enable a switched selection between linear and exponential curves. EGs with breakpoints in the attack, decay and release segments can produce similar effects to exponential curves, albeit with a crude approximation.

Triggering

The initiation of an EG is often assumed to be caused by a key being pressed on a music keyboard. Although this is the way that many synthesizers are set up, it is not the only way that envelopes can be started – an LFO or a VCO could provide a trigger which will start the EG. In this case, the envelope is not tied to the keyboard and can be used when a complex repeated CV is required (Figure 3.3.29).

When the keyboard is used to start an envelope, two separate signals are produced. The 'gate' signal indicates when the key is up or down, whilst the

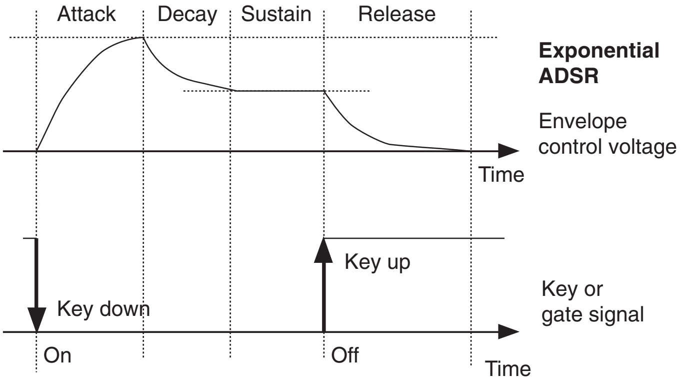


FIGURE 3.3.28 An exponential envelope does not use linear slopes and often provides more realistic sounding envelopes.

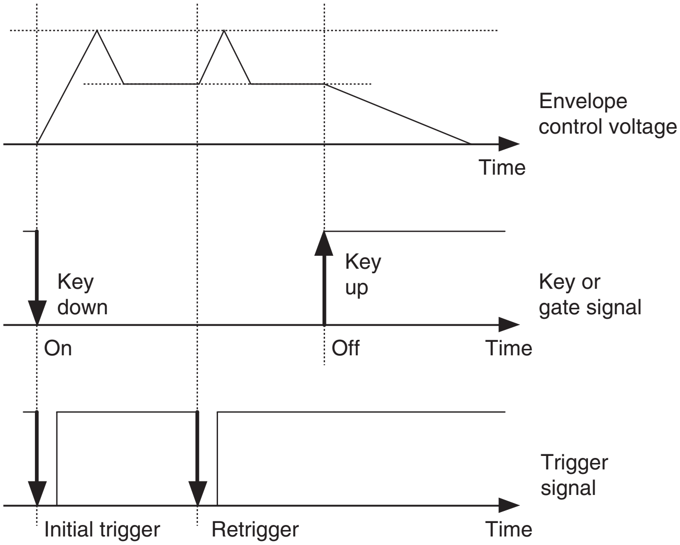


FIGURE 3.3.29 The retriggering of an EG can sometimes be used to add in a break-point and start a new attack, normally from the level which had been reached by the envelope. The overall length of the envelope is controlled by the key being pressed down, or a similar gate control in synthesizer which are not controlled by a keyboard. The retriggering of the envelope is controlled by a trigger signal which is generated by the start of each new note. This is normally found on monophonic synthesizers, where the gate is produced globally from any keys which are being held down, whilst the triggers are produced individually by each key.

start of the key depression is shown by a 'trigger' pulse (Figure 3.3.30). The response of an EG to these two signals depends on how the EG is configured.

'Single trigger' EGs start when they receive a gate and a trigger and progress through the envelope, entering the release segment when the gate signal ends to indicate that the key is no longer being held down. 'Multi trigger' EGs start when they receive a gate signal and a trigger pulse, but additional trigger pulses will restart part of the attack segment and the decay segment. These extra trigger pulses are normally produced by monophonic synthesizers (one note at once) only when a key is held down and another key is pressed.

'LFO trigger' or 'external trigger' EGs normally ignore the trigger pulse and treat the input signal as a gate. The width of the LFO waveform or the length of the external signal sets the length of the gate signal.

Whereas sources of audio signals or CVs can be routed to almost any destination in a synthesizer, the routing of trigger and gate signals is often much more restricted – usually they are hard-wired from the keyboard in performance instruments.

Voltage-controlled parameters

Some EGs provide voltage control of the segment times and levels. This enables the shape of the envelope to be changed with one or more CVs. One use of this facility is for 'scaling', where the length of all the times in the envelope are

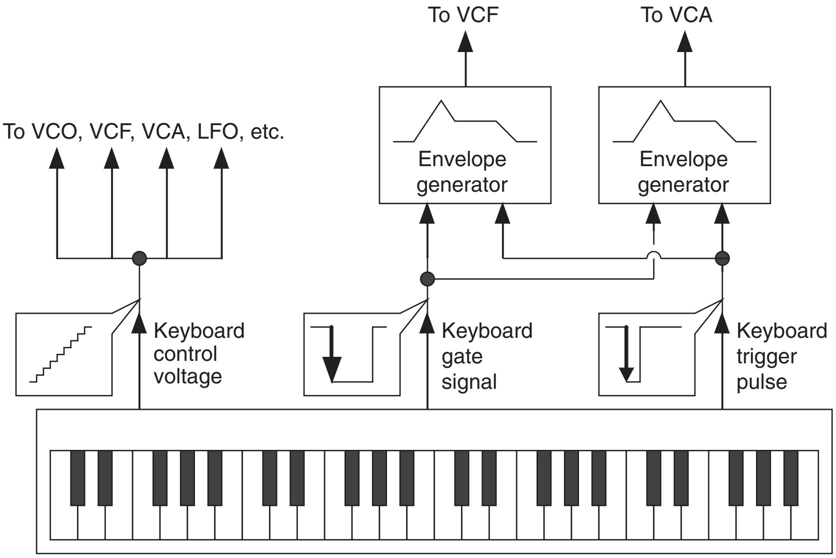


FIGURE 3.3.30 The gate and trigger routing from a keyboard to the EG is normally fixed, whilst the keyboard CV can be routed to a number of destinations.

Table 3.3.1 Summary of Envelope Segments

Symbol	Segment	Description	Type
D	Delay	The time from the start of the envelope to the start of the attack segment.	Time
I	Initial	The first level of the envelope. The quiescent level.	Level
A	Attack	The time taken for the envelope to rise from the initial level to the maximum (peak) level.	Time
H	Hold	The time that the envelope stays at the maximum (peak) level.	Time
P	Peak	The level to which the envelope rises at the end of the attack time.	Level
D	Decay	The time for the envelope to fall from the maximum (peak) level to the sustain or final level.	Time
B	Break-point	The level at which one decay segment changes to another.	Level
D	Decay	The time for the second decay segment to fall from the break point level to the sustain or final level.	Time
S	Sustain	The level at which the envelope stays whilst the key is held down (Gate signal On).	Level
S	Sustain	The time for which the sustain segment lasts (often the minimum time).	Time
R	Release	The time for the envelope to fall from the sustain level to the final level.	Time
F	Final	The final level of the envelope (usually the same as the initial).	Level

changed to imitate variations in envelope shape with pitch, in which case the CV would be derived from the keyboard pitch CV. This type of facility is much more commonly found in digital synthesizers.

3.3.7 Amplifiers

Most analogue synthesizers have a VCA as the final stage of the modifier section. The CV is used to change the gain of an amplifier.

The VCA controls the volume of the audio signal and is sometimes connected directly to the output of an EG. An offset voltage can also be used to provide a volume control; so even the output volume of a synthesizer can be voltage controlled.

The following are the two types of input to VCAs:

1. Linear inputs are used for tremolo and AM (amplitude modulation). They are also used with exponential curve envelopes.
2. Exponential inputs are used for volume changes and linear curve envelopes.

The combination of linear and exponential envelopes with linear and exponential VCAs provides much scope for confusion. Using an exponential curve envelope with an exponential VCA produces a result that has sudden or abrupt changes rather than steady transitions.

Tremolo is a cyclic variation in the volume of a sound. It is produced by using an LFO CV to alter the gain of a VCA. Tremolo normally uses a sine or triangle waveform at frequencies between 5 and 20 Hz. Higher frequencies from an LFO or a VCO produce AM, where the output of the VCA is a combination of the audio signal and the LFO or VCO frequency. See Section 3.3.1 for more details on AM.

Apart from their normal use as volume-controlling devices, VCAs can also be used to provide 'filtering' effects. By connecting the keyboard pitch voltage to the CV input of a VCA, the gain of the VCA is then dependent on the pitch CV from the keyboard. Since the keyboard pitch voltage normally rises as the keyboard note position rises, the VCA will act much as in a high-pass filter, since low notes will be at a lower volume than higher notes. By inverting the keyboard pitch voltage, a low-pass 'filter' effect can be produced. This coupling of the VCA to the keyboard pitch voltage is called 'scaling', since the output of the VCA is scaled according to the pitch (Figure 3.3.31).

3.3.8 Other modifiers

LFOs

LFOs are used to produce low-frequency CVs. They are in two forms: VCOs and special-purpose oscillators. VCO-based LFOs can have their frequency controlled with an external CV, whilst special-purpose oscillators cannot.

Unlike audio frequency VCOs, LFOs need to produce waveforms where the shape is normally more important than the harmonic content. So, in addition to the sine, square, pulse and sawtooth waveforms, additional shapes such as an inverted sawtooth are also provided. These might be used when the LFO is connected to a source such as a VCO and is controlling the pitch of the VCO. The basic sawtooth, or ramp-up waveform, would then produce a pitch that rose slowly and dropped quickly. The inverted shape, although still called a sawtooth, would now be a ramp-down waveform and would give a pitch that rose quickly and dropped slowly.

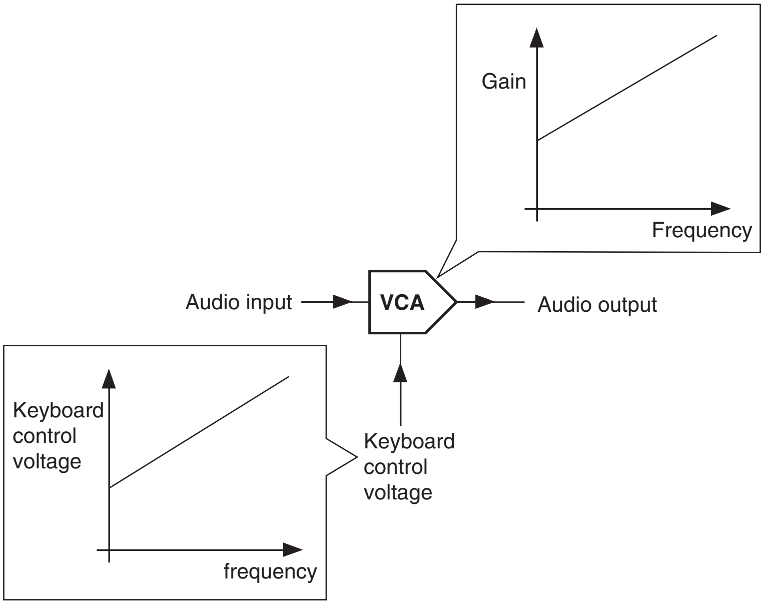


FIGURE 3.3.31 A VCA can be used to produce control of volume which follows the keyboard by routing the keyboard CV to the VCA gain control. This is similar to the tracking of a filter, and produces a coarse high-pass filtering effect, where higher notes are attenuated less than lower notes.

Two specialized LFO waveform outputs are often found on LFOs: (i) sample and hold and (ii) arbitrary.

Sample and hold is the name given to a random or repetitive sequence of CVs, that are produced by using the LFO to repeatedly take the value of another voltage source, and then keeping that value until the next time that it measures the value again (Figure 3.3.32). This process is called 'sampling' the value, and that value is then 'held' until the next sample is taken. The technique is thus called sample and hold.

If the voltage source that is sampled is noise, then the sample values will be random in level. This produces a series of values which do not repeat and are not regular or predictable. The regular timing from the periodic sampling is the only known quantity. If another LFO or VCO is sampled, then one of two results is possible. If the second LFO or VCO is not synchronized to the sampling LFO, then the output of the sample and hold will be a series of values which are partly random and partly repetitive – the exact pattern depends on the relative frequencies and the LFO/VCO waveform. If the sampling LFO and the second LFO/VCO are synchronized so that they are locked together with the LFO/VCO being a multiple or fraction of the sampling LFO, then the output pattern will repeat.

Sample and hold is often used to control the cut-off frequency of a resonant low-pass filter. This is an effective way of providing 'interest' and 'movement' in a sound when it is in the sustain segment of an ADSR envelope.

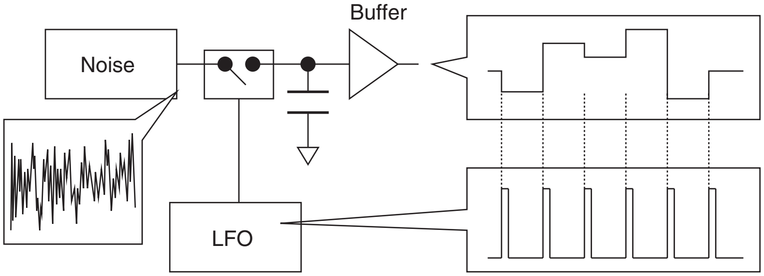


FIGURE 3.3.32 Sample and hold circuits take regular 'samples' of a noise (or other waveform) and then maintain that level until the next sample is taken. The rate of the samples is normally controlled by an LFO. The output consists of a series of steady voltages with rapid transitions, but whose level is not predictable. If the noise source is replaced with a repetitive waveform, then the output levels depend on the timing relationships between the sample LFO and the waveform being sampled.

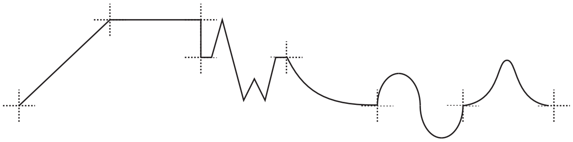


FIGURE 3.3.33 Arbitrary waveshape generators extend the concept of the multi-segment EG by providing additional shapes for the transition from one break-point to the next.

Unfortunately, the rhythmic random changing timbres, that this type of filter modulation produces, have become an overused cliché. But by reducing the amount of variation of cut-off frequency and using a slow LFO, or preferably a slow LFO triggered by key gates, it can be used as a way of making the timbre of successive notes slightly different.

Arbitrary waveforms are the ones, which are constructed from a series of simpler waveform segments (Figure 3.3.33). There are many variations possible:

- two or more levels (rather like a simple sequencer)
- two or more straight-line slopes (much like an envelope)
- two or more curves (exponential, linear, sine, power law, etc.).

Arbitrary waveform generators are also called function generators. They can be used to replace EGs, control panning and effects settings and even act as simple sequencers to produce a series of pitched notes.

LFO output waveforms are frequently available simultaneously, so that a sine wave can be used at the same time as a square waveform (Figure 3.3.34). The common outputs are as follows:

- sine
- triangle

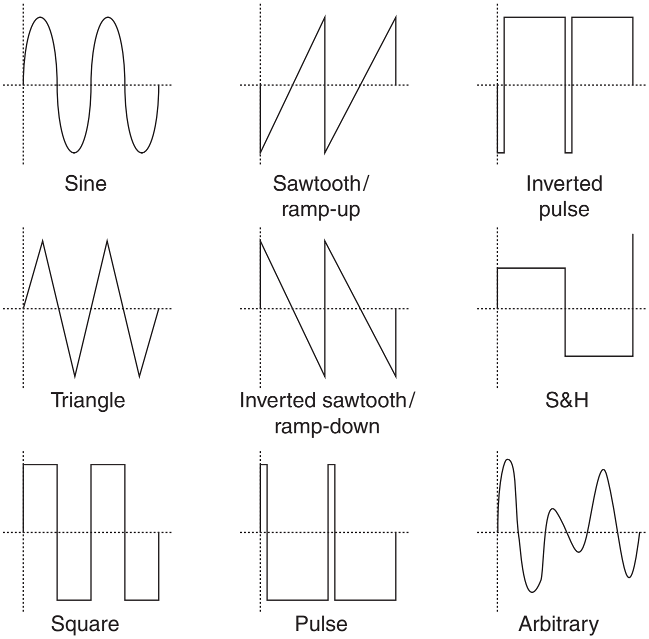


FIGURE 3.3.34 LFO outputs are normally provided in a variety of shapes to give additional control possibilities; although in practice, the sine wave is almost always used for vibrato or tremolo, and the square wave is almost exclusively used for trills. The other shapes are often presented in normal and inverted forms, and are often used for special effects sounds.

- square
- sawtooth/ramp-up
- inverted sawtooth/ramp-down
- pulse
- inverted pulse (100% pulse width)
- sample and hold
- arbitrary.

Envelope follower

An envelope follower takes an audio signal or CV, converts it to just positive values, and then low-pass filters it with a filter which has a very low cut-off frequency – a few hertz (Figure 3.3.35). This removes any high frequencies from the input, and leaves just a CV which represents the envelope of the input audio or CV. It is thus almost the opposite of a VCA: a VCA causes a CV to change the envelope of an audio signal, whilst an envelope follower takes an audio signal and produces a CV.

Some envelope followers also produce gate and trigger outputs, which are suitable for controlling EGs – the envelope follower is then a complete module for interfacing an external audio signal with an analogue synthesizer. If the envelope follower is used to process source CVs, then it can be used to 'smooth'

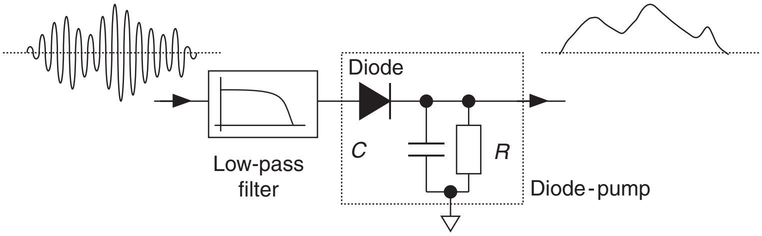


FIGURE 3.3.35 An envelope follower is used to 'extract' the envelope from an audio signal. This can be used to process external signals in a synthesizer. The audio signal is low-pass filtered and then a diode pump circuit is used to provide the final output voltage.

rapidly changing waveforms that have sharp transitions, or even produce portamento effects if the keyboard pitch CV is processed.

Externally triggered sample and hold

If a sample and hold circuit has an external sample clock input, then it can be used to sample voltage sources at non-periodic intervals. One suitable sample clock source is the keyboard gate or trigger signals. Using the keyboard to control the sample and hold, an output is produced, which changes only when a new key is pressed on the keyboard. By using an envelope follower to produce gate or trigger signals from an external audio input, the sample and hold can be driven from an external audio signal. In this way, any audio signal can be used as a source of CVs.

Waveshaper

Although rarely implemented on analogue synthesizers, the waveshaper is a nonlinear amplifier, which allows control over the relationship between the input and output signals. Any non-linearity in this relationship changes the shape of the waveform passing through the waveshaper, and this changes the harmonic content of the signal (Figure 3.3.36). Chapter 5 contains more information on the use of waveshaping in digital synthesizers.

Another interpretation of an analogue waveshaper is that it adds distortion to the signal and so it is best used for monophonic signals. A more familiar waveshaper is the 'fuzz box' used by guitarists, where the passing of polyphonic audio signals through a clipping circuit produces large amounts of distortion.

Modulation

Modulation is another type of modifier. Any parameter that can be voltage controlled is a potential means of modulation. Although VCAs are available from the front panels of many analogue synthesizers, they are also used inside to allow CVs to act as modulators – anywhere where a CV is used to change the amplitude or level of a signal or CV.

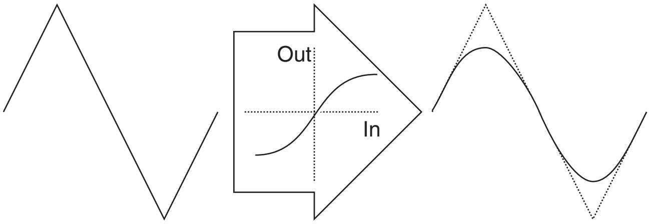


FIGURE 3.3.36 A waveshaper uses a non-linear transfer function to change the shape of a waveform. This is often used to convert a triangle waveform into an approximation of a sine wave and is adequate for shaping LFO and VCO outputs.

Some of the many possible ways that sources can be modified using modulation are as follows:

- *LFO (LFO/envelope/keyboard)*: LFO modulation changes the rate or frequency of the LFO. This can be used to produce vibrato or tremolo whose rate is not fixed.
- *VCO mod (LFO/envelope/keyboard)*: LFO modulation of a VCO produces vibrato. Envelope modulation produces pitch sweeps. Keyboard modulation changes the scaling of the VCO: it can change the keyboard so that an octave on the keyboard represents any pitch interval to the VCO.
- *Filter mod (LFO/envelope/keyboard)*: LFO modulation of a filter produces cyclic timbre changes. Envelope modulation produces dynamic timbral changes during the course of a single note. Keyboard modulation controls how the filter 'tracks' the note on the keyboard.
- *PWM (LFO/envelope/keyboard)*: PWM changes the timbre of the source waveform.
- *AM (LFO/VCO)*: AM with low frequencies produces tremolo. At higher frequencies it adds extra frequencies to the audio signal (see Section 3.4).
- *FM (LFO/VCO)*: FM uses the linear frequency CV input of the VCOs. It produces additional frequencies in the output signal (see Section 3.4 and Chapter 5).
- *Cross-modulation (VCO)*: Cross-modulation connects the outputs of two VCOs to their opposite's frequency CV input and so each frequency modulates the other. This produces complex FM-like timbres, but it can be difficult to control and keep in tune.
- *Pan (LFO/VCO/envelope/keyboard)*: LFO modulation of the stereo pan position produces 'auto-pan', where the audio signal moves cyclically from one side of the stereo image to the other. VCO modulation

can spread individual harmonics across the stereo image. Envelope modulation moves the image with the note envelope. Keyboard modulation places notes in the stereo image dependent on their position on the keyboard.

- *Other sources*: Many other sources and modifiers can be modulated. The effects section of many analogue synthesizers allows parameters like the reverberation time, flange speed and others to be controlled.

Controllers

In conventional instruments, the control of the sound production is often a mechanical linkage between the performer and the instrument. A saxophone player uses a number of levers to control the opening and closing of the holes that determine the effective length of the saxophone. Control over the timbre can be accomplished by how the lips grasp the mouthpiece and the reed, as well as the use of the tongue. Further expression comes from the lungs with control over air pressure.

The interfacing between the performer and the synthesizer sound generation circuitry is accomplished by one or more controller devices. The main note-pitch controller is usually a modified organ-type keyboard, although sometimes weighted action piano-type keyboards are used. Changes in pitch are normally produced with a rotary control called a pitch-bend wheel, and a similar control is used to add in modulation effects such as vibrato or tremolo. Control over volume and timbre can be accomplished by using a foot pedal – as used in organs for volume.

Keyboard

The familiar music keyboard with its patterned combination of black and white keys is widely used as the main discrete pitch control for note selection, as well as initiating envelopes. Although normally connected together, the pitch selection and envelope triggering functions can be separated.

Pitch-bend

Continuous control over the pitch is achieved by using a 'pitch-bend' controller. These are normally rotating wheels or levers and usually change the pitch of the entire instrument over a specified range (often a semitone or a fifth). They produce a CV whose value is proportional to the angle of the control. Pitch-bend controls normally have a spring arrangement, which always returns the control to the center 'zero' position (no pitch change) when it is released. This central position is often also mechanically detented, so that it can be felt by the operator, since it will require force to move it away from the center position.

Modulation

Modulation is controlled using rotary wheels or lever, where the CV is proportional to the angle of the control. Modulation controllers are not normally sprung so that they return to the center position. Some instruments allow pressure on the keyboard to be used as a modulation controller.

There have been some attempts to combine the functions of pitch-bend and modulation into a single 'joystick' controller, but the most popular arrangement remains the two wheels: pitch-bend and modulation.

Foot controllers

Foot controllers are pedals which provide a CV which is proportional to the angle of the pedal. Although associated with volume control, they can be used as modulation controls or even as pitch-bend controls.

Foot switches

Foot switches are foot-operated switches, which normally have only two values (some multi-valued variants are produced, but these are rare). They are used to control parameters such as sustain and portamento. See Chapters 7 and 8 for more details on controllers.

3.3.9 Using analogue synthesis

Learning how to make the best use of the available facilities provided by an analogue synthesizer requires time and effort. Although there are a number of 'standard' configurations of VCO, VCF, VCA and envelopes, the key to making the most of an analogue synthesizer is understanding how the separate parts work: both in isolation and in combination. If copies can be located, then Roland (1978, 1979) and De Furia (1986) are excellent references for further reading on this subject.

As a brief introduction to some of the techniques of using an analogue synthesizer, the remainder of this section shows how a subtractive analogue synthesizer can be an excellent learning tool for exploring some of the principles of audio and acoustics. Here are some of the demonstrations which can be carried out using a subtractive synthesizer.

Harmonic content of waveforms

The harmonic content of different waveshapes can be audibly demonstrated by using a low-pass VCF with high resonance (set just below self-oscillation) or a narrow band-pass filter. Each VCO waveform is connected to the filter input, and the filter cut-off frequency is slowly increased from zero to maximum (Figure 3.3.37). As the resonant peak passes the fundamental, the filter output will be a sine wave at that frequency. As the cut-off frequency is increased further, the fundamental sine wave will disappear, and the next harmonic will be heard as the cut-off frequency matches the frequency of the harmonic. The audible result is a series of sine waves, whose frequency matches the frequencies of the harmonics.

If noise is passed through the filter, then the output will be sine waves whose frequencies will be within the pass-band of the resonant peak, and whose levels will change randomly. The audible result is rather like whistling.

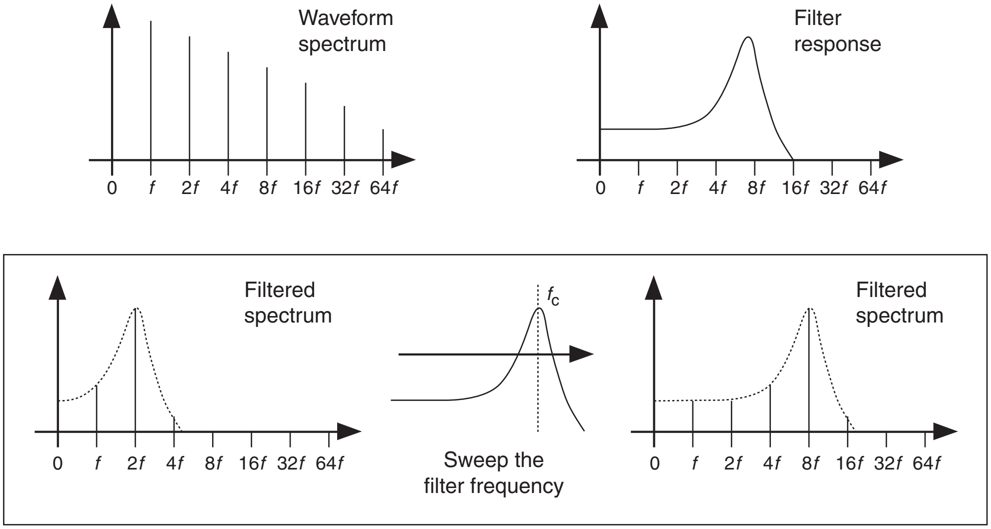


FIGURE 3.3.37 By varying the cut-off frequency of a resonant low-pass filter, the harmonic content of a waveform can be heard. As each of the harmonics which are present in the spectrum pass through the peak of the filter, they will be clearly heard. The frequency of the harmonic can be determined by noting the frequency of the filter when the harmonic is heard.

Harmonic content of pulses

The harmonic content of different pulse widths of pulse waveforms can be demonstrated by listening to the pulse waveform and changing the pulse width manually (Figure 3.3.38). At a pulse width of 50%, the sound will be noticeably hollow in timbre: this is a square wave. The square wave position can be heard because the second harmonic, which is one octave above the fundamental, will disappear. Using the resonant filter technique described in the previous example, individual harmonics can be examined – tuning the filter to the harmonic which disappears for a square wave can be used to emphasize this effect. As the pulse width is reduced, the timbre will then become brighter and brighter, and with very small pulse widths, the sound may disappear entirely. (This is a consequence of the design of the VCO circuitry, and not an acoustic effect!) Conversely, increasing the pulse width from 50% produces the same changes in the timbre and, again at very large pulse widths, may result in the loss of the sound.

Filtering

Many resonance and ringing filter effects can be demonstrated by connecting a percussive envelope to a VCF CV input and turning up the resonance. Just below self-oscillation, the filter can be made to oscillate for a short time by using the envelope to trigger the oscillation (Figure 3.3.39).

This 'ringing oscillator' is the basis of the designs for many drum machine sounds in the 1970s (see Section 3.3.5 and Figure 3.3.7).

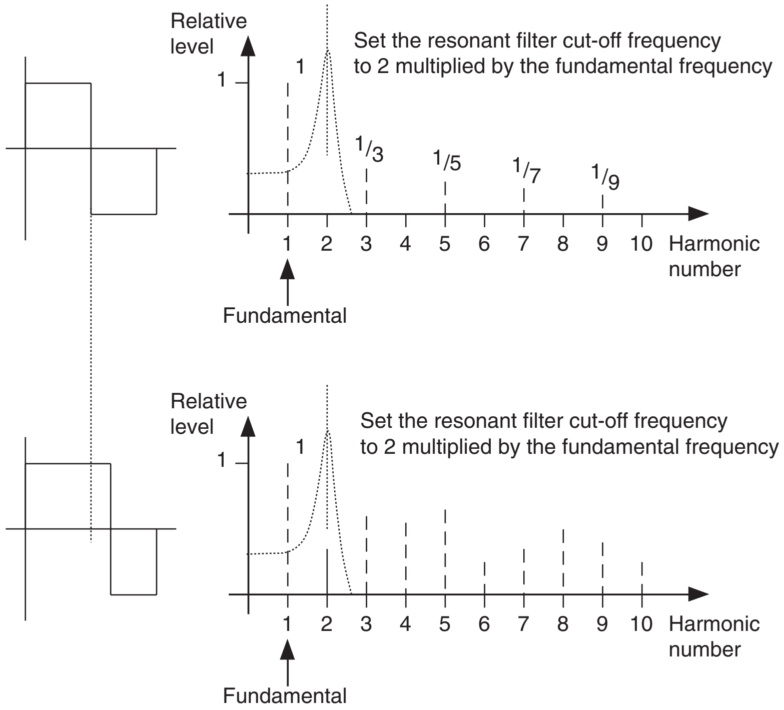


FIGURE 3.3.38 The harmonic content of a square wave and a rectangular wave is different, especially the even harmonics. The second harmonic is not present in a square wave and yet can be clearly heard in a rectangular waveform. This can be used to produce square waves from a VCO which provides control over the width of the pulse. By adjusting the pulse width control and listening for the disappearance of the second harmonic, a square wave can be produced.

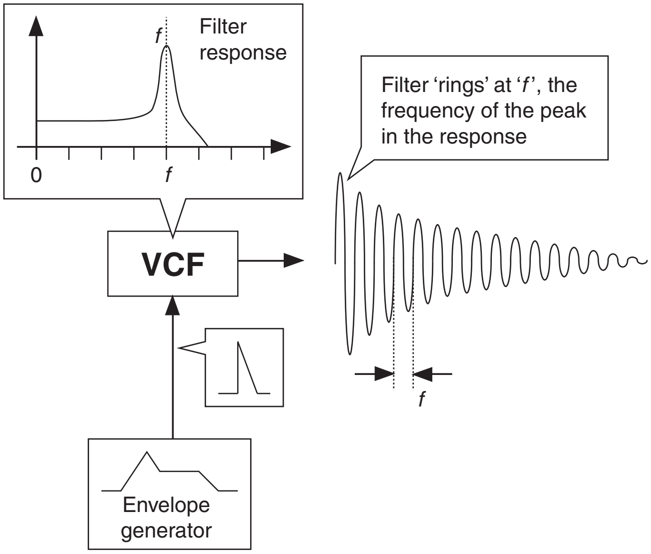

Filter response
f
0
f
Filter 'rings' at 'f', the frequency of the peak in the response
VCF
f
Envelope generator

FIGURE 3.3.39 If a strongly resonant filter is 'triggered' by a brief pulse of noise or an envelope pulse, then it can 'ring' producing a decaying oscillation at the cut-off or peak frequency.

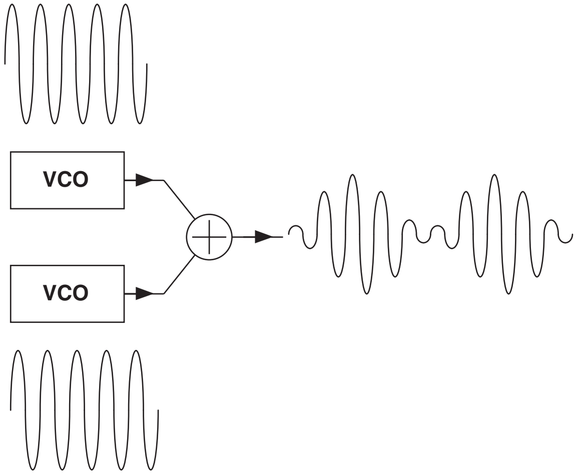


FIGURE 3.3.40 Beats can be demonstrated by mixing together the outputs of two VCOs which have slightly different frequencies. The two waveforms will cyclically add together or subtract, and so produce an output that varies in level. The audible effect is an interesting 'chorus' type of sound for frequency differences of less than 2 Hz and vibrato for 2–20 Hz.

White noise filtered by a resonant low-pass filter changes from a hiss to a rumble as the cut-off frequency is reduced, because the filter is acting as a narrow bandwidth band-pass filter. With very narrow bandwidths, the noise then begins to produce a sense of pitch; and by connecting the keyboard voltage to the VCF so that it tracks the keyboard, these 'pitched noise' sounds can then be played with the keyboard. Keen experimenters might like to compare this with an alternative approach with audibly similar results: modulating the frequency of a VCO with noise.

Beats

Beats occur when two VCOs or audio signals are detuned relative to each other. The interference between the two signals produces a cyclic variation in the overall level as they combine or cancel each other out repeatedly (Figure 3.3.40). The time between the cancellations is related to the difference in frequency between the two audio signals or VCOs. Using two VCOs with a beat frequency of 1 Hz or less produces a 'lively', 'rich' and interesting sound.

PWM uses an LFO to cyclically change the width of a pulse waveform from a single VCO. The result has many of the audible characteristics of two VCOs beating together.

Vibrato versus tremolo

- *Vibrato is FM*: The frequency of the audio signal is changed. Using an LFO to modulate the frequency of a VCO produces vibrato.
- *Tremolo is AM*: The level of the audio signal is changed. Using an LFO to modulate the level of an audio signal using a VCA produces tremolo.

Table 3.3.2 Modulation Summary

	Constant	Cyclic Change
AM	Frequency, pulse width	Amplitude
FM	Amplitude, pulse width	Frequency
PWM	Amplitude, frequency	Pulse width

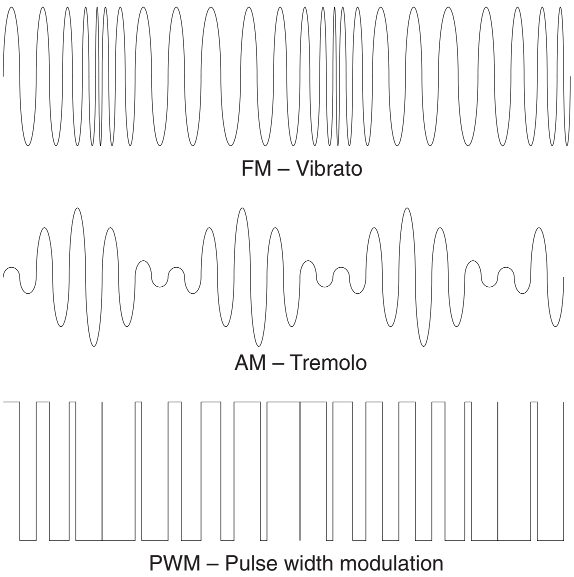


FIGURE 3.3.41 Vibrato is a cyclic variation in the frequency of a sound, whilst tremolo is a cyclic variation in the level of a sound.

Modulation summary and the cyclic variations of vibrato and tremolo are shown in Table 3.3.2 and Figure 3.3.41, respectively.

3.4 Additive synthesis

Subtractive synthesis starts out with a harmonically rich sound and 'subtracts' some of the harmonics, whereas additive synthesis does almost the exact opposite. It adds together sine waves of different frequencies to produce the final sound. Because large numbers of parameters need to be controlled simultaneously, the user interface is usually much more complex than that of a subtractive synthesizer.

3.4.1 Theory: additive synthesis

Additive synthesis is based on the work produced by Fourier, a French mathematician from the nineteenth century. In 1807, Fourier showed that the shape of any repetitive waveform could be reproduced by adding together simpler waveforms, or alternatively, that any periodic waveform could be described by specifying the frequency and amplitude of a series of sine waves. The restriction that the waveshape must repeat is imposed to keep the mathematics manageable. Without the restriction it is still possible to convert any waveform into a series of sine waves, but since the waveform is not constant, the sine waves that make it up are not constant either.

One useful analogy is to think of trying to describe writing to someone, who has never seen it, over the telephone. You might start out by describing how the words are broken up into letters and these letters are made up out of lines, dots and curves. This works perfectly well as long as the words you might try to describe stay fixed, but if they change, then you would have to keep updating your description. You could still convey the information about the shape of the letters that make up the words, but you would have to provide lots more detailed description as the letters change. The simplest example of synthesizing a waveform using Fourier synthesis is a sine wave. A sine wave is made up of just one sine wave, at the same frequency! In terms of harmonics, a sine wave contains just one frequency component, at the repetition rate of the fundamental.

More complicated waveshapes can be made by adding additional sine waves. The simplest method involves using simple integer multiples of the fundamental frequency. So, if the fundamental is denoted by f, then the additional frequencies will be $2f$, $3f$, $4f$, etc. These are the frequencies that occur in some of the basic waveshapes-sawtooth, square, etc and are known as harmonics. Because the numbering of the harmonics is based around their position above the fundamental or first harmonic, with a frequency of f, then the second harmonic has a frequency of $2f$. The second harmonic is also sometimes called the first overtone (Table 3.4.1).

Table 3.4.1 Harmonics, Frequencies and Overtones

Frequency	Harmonic	Overtone
f	fundamental	Fundamental
$2f$	2	1
$3f$	3	2
$4f$	4	3
$5f$	5	4
$6f$	6	5
$7f$	7	6
$8f$	8	7
$9f$	9	8
$10f$	10	9

3.4.2 Harmonic synthesis

So far, additive synthesis seems to be based around producing a specific waveform from a series of sine waves. In practice, the 'shape' of a waveform is not a good guide to its harmonic content, since minor changes to the shape can produce large changes in the harmonic content. Conversely, simple changes of phase for the harmonics can produce major changes in the shape of the waveform. In fact, although the human ear is mainly concerned with the harmonic content, the relative phase of the harmonics can be very important at low frequencies. For frequencies above 440 Hz, you can change the phase of a harmonic and thus alter the resulting shape of the waveform, but the basic timbre will sound the same. Control over phase is thus useful under some circumstances and is found in some additive synthesizers.

The harmonic content of waveshapes is a useful starting point for examining this relationship between shape and perception. Mathematically and harmonically, the 'simplest' waveshape is the sine wave. Sine waves sound clean and pure, and perhaps even a little bit boring. Adding in small amounts of odd-numbered harmonics produces a triangular waveshape, which has enough harmonic content to stop it sounding quite as pure as the sine wave (Figure 3.4.1).

A square wave contains only odd harmonics. It has a characteristic 'hollow' sound, and the absence of the second harmonic is particularly noticeable if a square wave is compared with a sawtooth wave (Figure 3.4.2).

A square wave that has been produced with a phase change in the second harmonic no longer looks like a 'square' wave, and yet the harmonic content is the same (Figure 3.4.3).

A sawtooth wave contains both odd and even harmonics. It sounds bright, although many pulse and 'super-sawtooth' waveshapes can contain greater levels of harmonics. Again, a sawtooth wave with a phase change in the second harmonic does not look like a sawtooth, although it still sounds like one to the ear (Figure 3.4.3).

Pulse waves contain more and more harmonics as the pulse width narrows (or widens) from square. A 10% pulse has the same spectrum as a 90% pulse and it also sounds the same to the ear. One special case is the square wave, where the even harmonics are missing completely. Pulse widths of anything other than 50% include the second harmonic, and this can usually be clearly heard as the pulse width is varied away from the 50% value.

Finally, there is the 'even harmonic' wave. If a sawtooth contains both odd and even harmonics and a square wave contains just the odd harmonics, then what does a wave containing just the even harmonics look like? Actually, it is just another square wave, but one octave higher in pitch, and with a fundamental frequency of $2f$!

In practice, adding together sine waves produces waveforms that have some of the characteristics of the mathematically perfect ideal waveforms, but not all. Producing square edges on a square wave would require large numbers of harmonics – an infinite number for a 'perfect' square wave. Using just

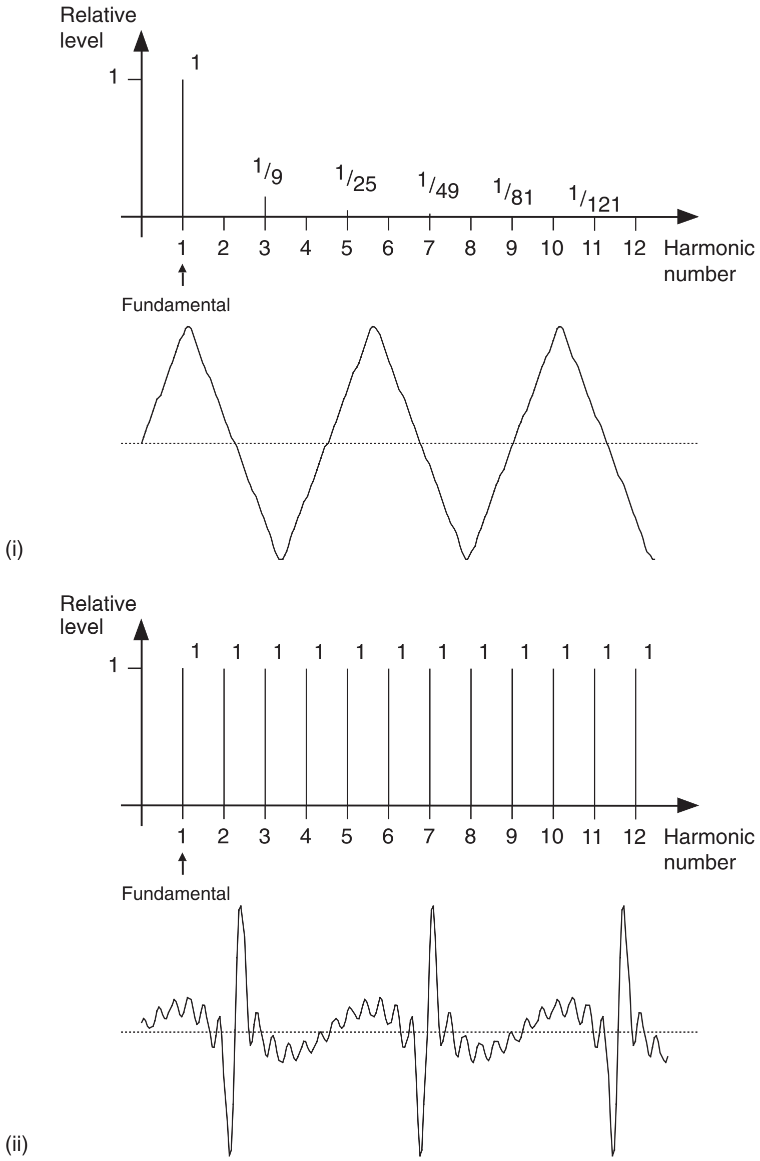


FIGURE 3.4.1
(i) A triangle waveform constructed from six sine wave harmonics is very different from a sine wave, even though the fundamental is by far the strongest component. (ii) A combination of equal amounts of the first 12 harmonics produces a waveform which looks (and sounds) like a type of pulse waveshape.

a few harmonics can produce waveforms that have enough of the harmonic content to produce the correct type of timbre, even though the shape of the waveform may not be exactly as expected.

3.4.3 Harmonic analysis

In order to produce useful timbres, an additive synthesizer user really needs to know about the harmonic content of real instruments, rather than mathematically derived waveforms. The main method of determining this information is Fourier analysis, which reverses the concept of making any waveform out

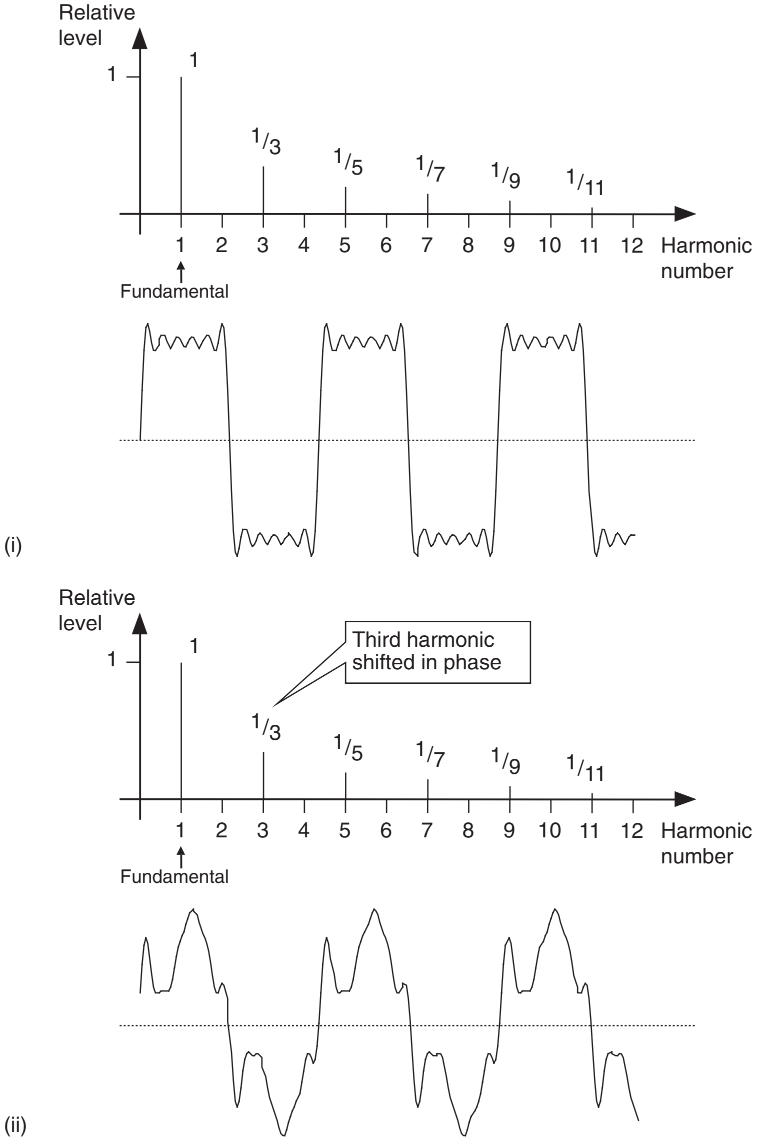


FIGURE 3.4.2
(i) A square waveform constructed from six sine wave harmonics has a close approximation to the ideal waveshape.
(ii) Changing the phase of the third harmonic radically alters the shape of the waveform.

of sine waves and uses the idea that any waveform can be split into a series of sine waves.

The basic concept behind Fourier analysis is quite simple, although the practical implementation is usually very complicated. If an audio signal is passed through a very narrow band-pass filter that sweeps through the audio range, then the output of the filter will indicate the level of each band of frequencies which are present in the signal (Figure 3.4.4). The width of this band-pass filter determines how accurate the analysis of the frequency content will

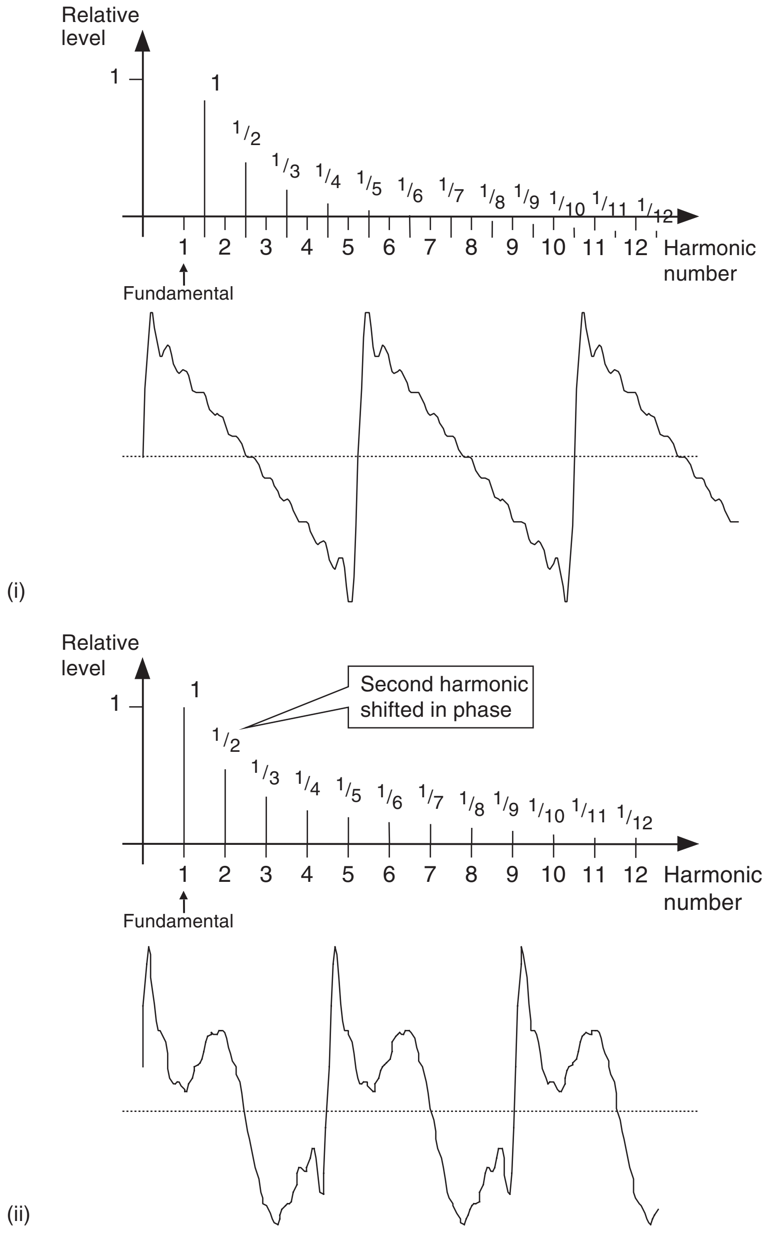


FIGURE 3.4.3
(i) A sawtooth waveform constructed from 12 sine wave harmonics has a close approximation to the ideal waveshape.
(ii) Changing the phase of the second harmonic radically alters the shape of the waveform.

be: if it is 100 Hz wide, then the output can only be used to a resolution of 100 Hz, whereas if the band-pass filter has a 1-Hz bandwidth, then it will be able to indicate individual frequencies to a resolution of 1 Hz.

For simple musical sounds that contain mostly harmonics of the fundamental frequency, the resolution required for Fourier analysis is not very high. The more complex the sound, the higher the required resolution. For sounds that have a simple structure consisting of a fundamental and harmonics, a rough 'rule of thumb' is to make the bandwidth of the filter less than the fundamental

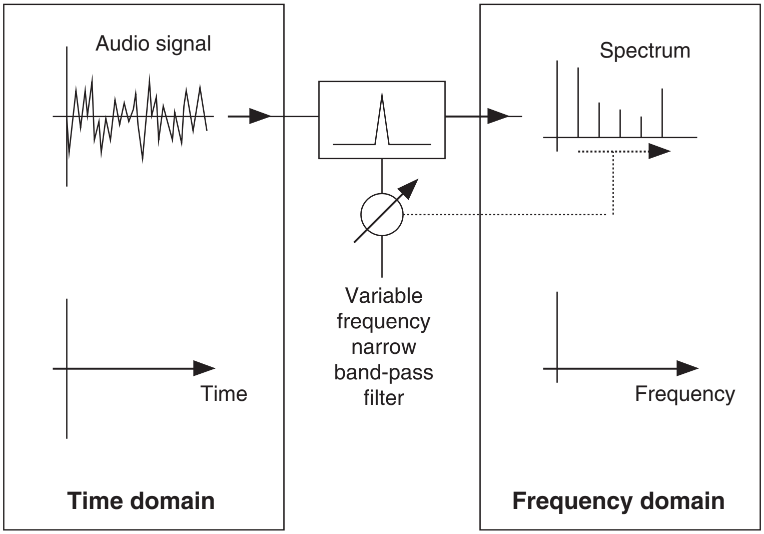


FIGURE 3.4.4
Sweeping the center frequency of a narrow band-pass filter can convert an audio signal into a spectrum: from the time domain to the frequency domain.

frequency, since the harmonics will be spaced at frequency intervals of the fundamental frequency. Having 1-Hz resolution in order to discover that there are five harmonics spaced at 1-kHz intervals is extravagant. Smaller bandwidths require more complicated filters, and this can increase the cost, size and processing time, depending on how the filters are implemented. Fourier analysis can be achieved using analogue filters, but it is frequently carried out by using digital technology (see Section 5.8).

Numbers of harmonics

How many separate sine waves are needed in an additive synthesizer? Supposing that the lowest fundamental frequency which will be required to be produced is a low A at 55 Hz, then the harmonics will be at 110, 165, 220, 275, 330, 385, 440 Hz,… The 32nd harmonic will be at 1760 Hz and the 64th harmonic at 3520 Hz.

An A at 440 Hz has a 45th harmonic of 19,800 Hz. Most additive synthesizers seem to use between 32 and 64 harmonics (Table 3.4.2).

Harmonic and inharmonic content

Real-world sounds are not usually deterministic: they do not contain just simple harmonics of the fundamental frequency. Instead, they also have additional frequencies that are not simple integer multiples of the fundamental frequency. The following are several types of these unpredictable ‘inharmonic’ frequencies:

- noise
- beat frequencies
- sidebands
- inharmonics.

Noise has, by definition, no harmonic structure, although it may be present only in specific parts of the spectrum: colored noise. So any noise which is present in a sound will appear as random additional frequencies within those bands, and whose level and phase are also random.

Table 3.4.2 Additive Frequencies and Harmonics

Frequency				Harmonic fundamental
55	110	220	440	
110	220	440	880	2
165	330	660	1,320	3
220	440	880	1,760	4
275	550	1,100	2,200	5
330	660	1,320	2,640	6
385	770	1,540	3,080	7
440	880	1,760	3,520	8
495	990	1,980	3,960	9
550	1,100	2,200	4,400	10
605	1,210	2,420	4,840	11
660	1,320	2,640	5,280	12
715	1,430	2,860	5,720	13
770	1,540	3,080	6,160	14
825	1,650	3,300	6,600	15
880	1,760	3,520	7,040	16
935	1,870	3,740	7,480	17
990	1,980	3,960	7,920	18
1,045	2,090	4,180	8,360	19
1,100	2,200	4,400	8,800	20
1,155	2,310	4,620	9,240	21
1,210	2,420	4,840	9,680	22
1,265	2,530	5,060	10,120	23
1,320	2,640	5,280	10,560	24
1,375	2,750	5,500	11,000	25
1,430	2,860	5,720	11,440	26
1,485	2,970	5,940	11,880	27
1,540	3,080	6,160	12,320	28
1,595	3,190	6,380	12,760	29
1,650	3,300	6,600	13,200	30
1,705	3,410	6,820	13,640	31
1,760	3,520	7,040	14,080	32
1,815	3,630	7,260	14,520	33
1,870	3,740	7,480	14,960	34
1,925	3,850	7,700	15,400	35

Table 3.4.2 (Continued)

Frequency				Harmonic fundamental
1,980	3,960	7,920	15,840	36
2,035	4,070	8,140	16,280	37
2,090	4,180	8,360	16,720	38
2,145	4,290	8,580	17,160	39
2,200	4,400	8,800	17,600	40
2,255	4,510	9,020	18,040	41
2,310	4,620	9,240	18,480	42
2,365	4,730	9,460	18,920	43
2,420	4,840	9,680	19,360	44
2,475	4,950	9,900	19,800	45
2,530	5,060	10,120	20,240	46
2,585	5,170	10,340	20,680	47
2,640	5,280	10,560	21,120	48
2,695	5,390	10,780	21,560	49
2,750	5,500	11,000	22,000	50
2,805	5,610	11,220	22,440	51
2,860	5,720	11,440	22,880	52
2,915	5,830	11,660	23,320	53
2,970	5,940	11,880	23,760	54
3,025	6,050	12,100	24,200	55
3,080	6,160	12,320	24,640	56
3,135	6,270	12,540	25,080	57
3,190	6,380	12,760	25,520	58
3,245	6,490	12,980	25,960	59
3,300	6,600	13,200	26,400	60
3,355	6,710	13,420	26,840	61
3,410	6,820	13,640	27,280	62
3,465	6,930	13,860	27,720	63
3,520	7,040	14,080	28,160	64

Beat frequencies arise when the harmonics in a sound are not perfectly in tune with each other. 'Perfect' waveshapes are always assumed to have harmonics at exact multiples of the fundamental, whereas this is not always the case in real-world sounds. If a harmonic is slightly detuned from its mathematically 'correct' position, then additional harmonics may be produced at the beat frequency, so if a harmonic is 1 Hz too high in pitch relative to the fundamental, then a frequency of 1 Hz will be present in the spectrum.

Sidebands occur when the frequency stability of a harmonic is imperfect, or when the sound itself is frequency modulated. Both cases result in pairs of frequencies which mirror around the 'ideal' frequency. So a 1-kHz sine wave which is frequency modulated with a few hertz will have a spectrum that contains frequencies on either side of 1 kHz, and the exact content will depend on the depth of modulation and its frequency. See Section 3.5.1 for more details.

Inharmonics are additional frequencies that are structured in some way, and so are not noise, but which do not have the simple integer multiple relationship with the fundamental frequency. Timbres that contain inharmonics typically sound like a 'bell' or 'gong'.

Many additive synthesizers only attempt to produce the harmonic frequencies, with perhaps a simple noise generator, as well. This deterministic approach limits the range of sounds which are possible, since it ignores many stochastic, probabilistic or random elements which make up real-world sounds.

3.4.4 Envelopes

The control of the level of each harmonic over time uses EGs and VCAs. Ideally, one EG and one VCA should be provided for each harmonic. This would mean that the overall envelope of the final sound was the result of adding together the individual envelopes for each of the harmonics, and so there would be no overall control over the envelope of the complete sound. Adding an overall EG and VCA to the sum of the individual harmonics allows quick modifications to be made to the final output (Figure 3.4.5).

In order to minimize the number of controls and the complexity, the EGs need to be as simple as possible without compromising the flexibility. Delayed ADR (DADR) envelopes are amongst the easiest of EGs to implement in discrete analogue circuitry, since the gate signal can be used to control a simple capacitor charge and discharge circuit to produce the ADR envelope voltage. DADR envelopes also require only four controls (delay time, attack time, decay

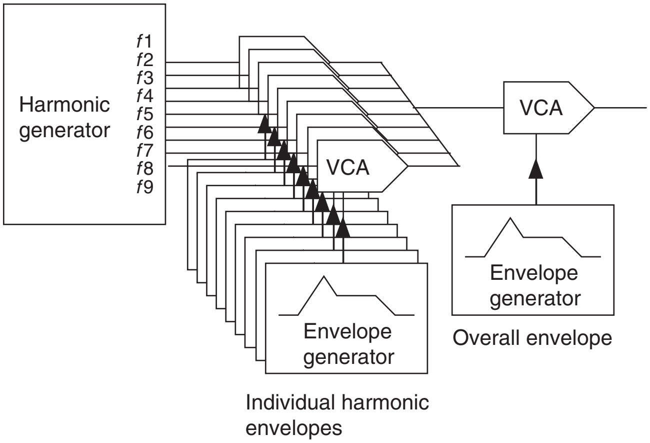


FIGURE 3.4.5 Individual envelopes are used to control the harmonics, but an overall envelope allows easy control over the whole sound which is produced.

time and release time), whereas a DADSR would require five controls and more complex circuitry. If integrated circuit (IC) EGs are used, then the ADSR envelope would probably be used, since most custom synthesizer chips provide ADSR functionality.

Control grouping and ganging

With large numbers of harmonics, having separate envelopes for each harmonic can become very unwieldy and awkward to control. The ability to assign a smaller number of envelopes to harmonics can reduce the complexity of an additive synthesizer considerably. This is only effective if the envelopes of groups of harmonics are similar enough to allow a 'common' envelope to be determined. Similarly, ganging together controls for the level of groups of harmonics can make it easy to make rapid changes to timbres – altering individual harmonics can be very time consuming. Simple groupings such as 'all of the odd' or 'all of the even' harmonics, can be useful starting points for this technique.

A more advanced use for grouping involves using keyboard voltages to give pitch-dependent envelope controls. This can be used to create the effect of fixed resonances or 'formants' at specific frequencies.

Filter simulation/emulation

Filters modify the harmonic content of a sound. In the case of an additive synthesizer, there are two ways that this can be carried out: with a filter or with a filter emulation. As with the overall envelope control mentioned earlier, there are advantages to having a single control for the combined harmonics, and a VCF could be added just before the VCA. Such a filter would only provide crude filtering of the sound, in exactly the same way as in subtractive synthesis.

Filter emulation uses the individual EGs for the harmonics to 'synthesize' a filter by altering the envelopes. For example, if the envelopes of higher harmonics are set to have progressively shorter decay times, then when a note is played, the high harmonics will decay the first (Figure 3.4.6). This has an audible effect which is very similar to a low-pass filter being controlled by a decaying envelope. The difference is that the 'filter' is the result of the action of all the envelopes, rather than one envelope. Consequently, individual envelopes can be changed, which then allow control over harmonics that would not be possible using a single VCF.

As with the envelope control ganging and grouping, similar facilities can be used to make filter emulation easier to use, although the implementation of this is much easier in a fully digital additive instrument.

3.4.5 Practical problems

Analogue additive synthesis suffers from a number of design difficulties. Generating a large number of stable, high-purity sine waves simultaneously can be very complex, especially if they are not harmonically related. Providing sufficient controls for the large number of available parameters is also a problem.

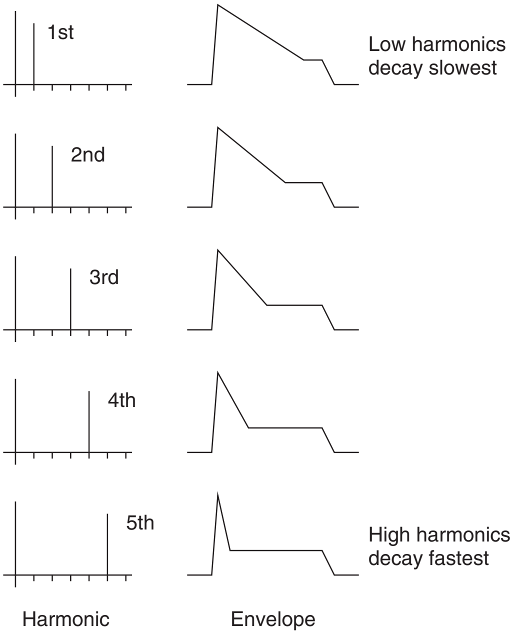


FIGURE 3.4.6 By using different envelopes for each harmonic, a filter can be 'synthesized'. This example shows the equivalent of a low-pass filter being produced by a number of different decaying envelopes.

Depending on the complexity of the design, an additive synthesizer might have the following parameters repeated for each harmonic:

- frequency (fixed harmonic or variable inharmonic)
- phase
- level
- envelope (DADR, DADSR or multi-segment – four or more controls).

For a 32-harmonic additive synthesizer, these eight parameters give a total of just over 250 separate controls, ignoring any additional controls for ganging and filter emulation. Although it is possible to assemble an additive synthesizer using analogue design techniques, practical realizations of additive synthesizers have tended to be digital in nature, where the generation and control problems are much more easily solved.

Spectrum plots

The subtractive and additive sections in this chapter have both shown plots of the harmonic content of waveforms, showing a frequency axis plotted against level. This 'harmonic content' graph is called a spectrum, and it shows the relative levels of the frequencies in an audio signal. Whereas a waveform is a way of showing the shape of a waveform as its value changes with time, a spectrum is a way of showing the harmonic content of a sound. The shape of a waveform is not a very good indication of the harmonic content of a sound, whereas a spectrum is – by definition.

Spectra (the plural of the Latin-derived word ‘spectrum’) are not very good at showing any changes in the harmonic content of a sound – in much the same way that a single cycle of a PWM waveform does not convey the way that the width of the pulse is changing over time. To show changes in spectra, a ‘waterfall’ or ‘mountain’ graph is used, which effectively ‘stacks’ several spectra together. The resulting 3D-like representation can be used to show how the frequency content changes with time (Figure 3.4.7).

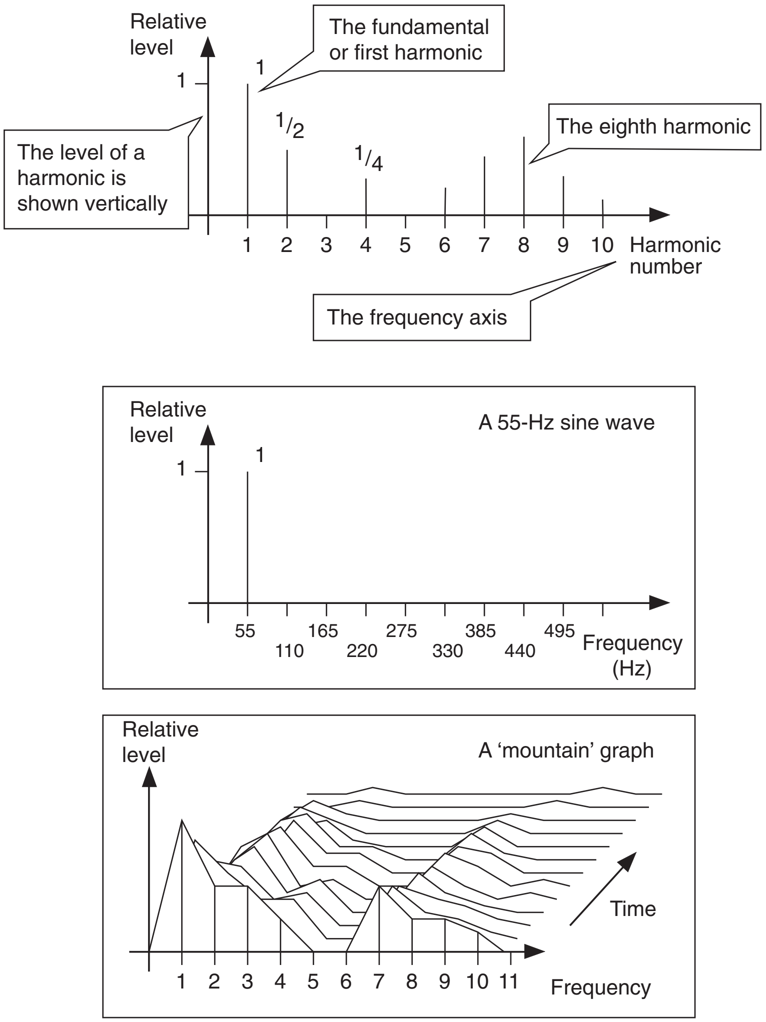


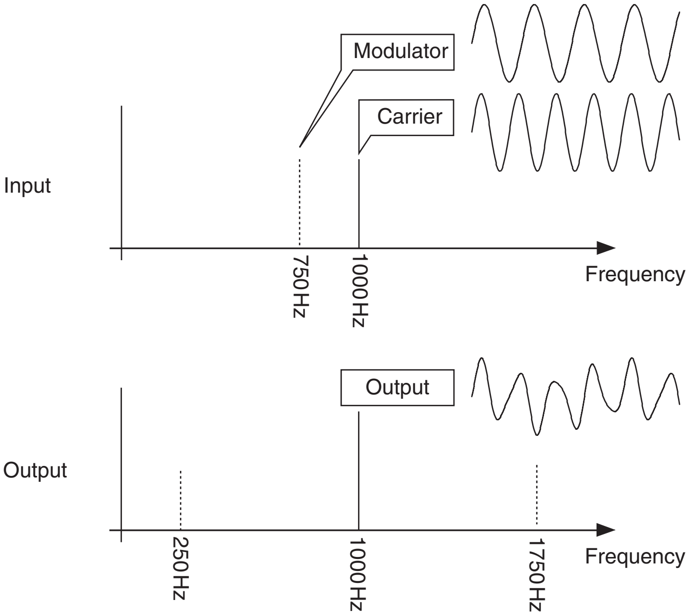
FIGURE 3.4.7 A spectrum is a plot of frequency against level. It thus shows the harmonic content of an audio signal. In most of the examples in this book, the horizontal axis is normally shown with harmonic numbers instead of frequencies – the 55-Hz sine wave spectrum shows the correspondence with frequency. When a spectrum changes with time, then a ‘mountain’ graph may be used to show the changes in the shape.

3.5 Other methods of analogue synthesis

3.5.1 Amplitude modulation

AM is a variation on one method used to transmit radio broadcasts. AM radio works by using a high-frequency signal as the 'carrier' of the audio signal as a radio wave. The carrier signal on its own conveys no information – it is the modulation of the carrier by the audio signal that provides the information by changing the level of the carrier. In the simplest case, a sine wave audio signal is used to change (or modulate) the level of the carrier signal. The resulting output signal contains not only the original carrier frequency but also the sum and the difference of the carrier and audio frequencies; these are called sidebands, because they are on either side of the carrier.

For audio AM, the two frequencies are both in the audio range, but the same principles apply – the output consists of the carrier frequency, and the sum of the two frequencies and difference between the two frequencies (Figure 3.5.1). So with a carrier of 1000 Hz and a modulator of 750 Hz, the output sideband frequencies will be 1000, 1750 and 250 Hz. Note that the modulating frequency is not present in the output. For 100% modulation, the sidebands have half the amplitude of the carrier.



Amplitude modulation	Outputs	
Carrier 1000 Hz	1000 Hz	
Modulator: 750 Hz	250 Hz	1750 Hz

FIGURE 3.5.1
AM with two sine waves produces outputs at the sum and difference of the two input frequencies.

For AM with waveforms other than sine waves, each component frequency is treated separately. So for a sine carrier and a non-sinusoidal wave modulator, there are actually the equivalent of several modulator frequencies: one for each harmonic in the modulator. For a sawtooth modulator wave, this means that there will be integer multiples of the modulator frequency at decreasing levels. Each of these harmonics will produce sidebands around the carrier. The carrier frequency of 1000 Hz will also be present in the output. Again, with 100% modulation, the sidebands will have half the amplitude of the carrier.

With a non-sinusoidal carrier of 1000 Hz and a sine wave modulator of 750 Hz, it is the equivalent of several carrier frequencies, and each carrier produces its own set of sidebands from the modulation frequency. For a sawtooth carrier, this means that there will be the equivalent of a carrier at each integer multiple of the carrier frequency, and each will produce sidebands from the modulator frequency. With 100% modulation the sidebands will have half the amplitude of the carrier. All of the harmonics in the carrier wave will also be present in the output (Figure 3.5.2).

For the case of two non-sinusoidal waves, AM produces a set of sidebands for each carrier harmonic, using each modulator harmonic. AM is thus a simple way of producing complex sounds with a number of harmonics that are not related to the fundamental (inharmonics) (Figure 3.5.3).

In an analogue synthesizer, AM is produced by connecting a VCO to the modulation control input of a VCA which is processing the output of another VCO. If the modulating frequency is lower than about 25 Hz, then AM is called tremolo and it is perceived as a rapid cyclic change in the amplitude.

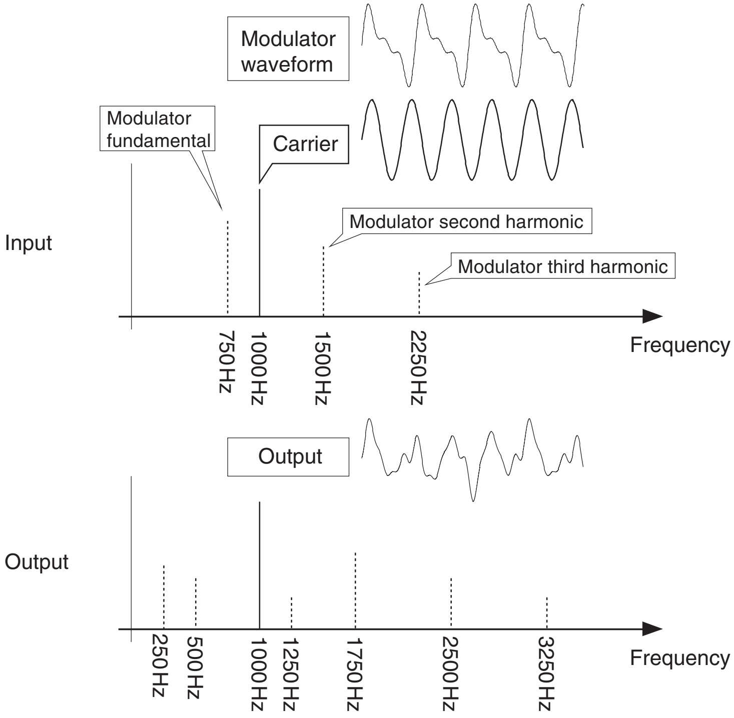
3.5.2 Frequency modulation

FM also employs another method which is normally used for the transmission of radio broadcasts. FM radio again uses a high-frequency signal as the 'carrier' of the audio signal. The modulation of the carrier signal by the audio signal 'carries' the information by changing the frequency of the carrier. The simplest case is where a sine wave audio signal is used to change (or modulate) the frequency of the carrier signal. The amount of frequency change is called the deviation, f_c, and instead of producing just one pair of sideband frequencies, FM can produce many sidebands, where the extra sidebands are similar to the harmonics in the sawtooth AM case described in Section 3.5.1, and this is just for sine wave carrier and modulator frequencies.

The number of sidebands that are produced can be determined by using the modulation index which is a measure of the amount of modulation and is being applied to the carrier. The modulation index is given by dividing the deviation by the modulator frequency, f_m:

$$\text{Modulation index} = \delta f_c \,/\, f_m$$

Note that the modulation index is dependent not only on how much the carrier frequency is changed but also on the modulator frequency. The resulting



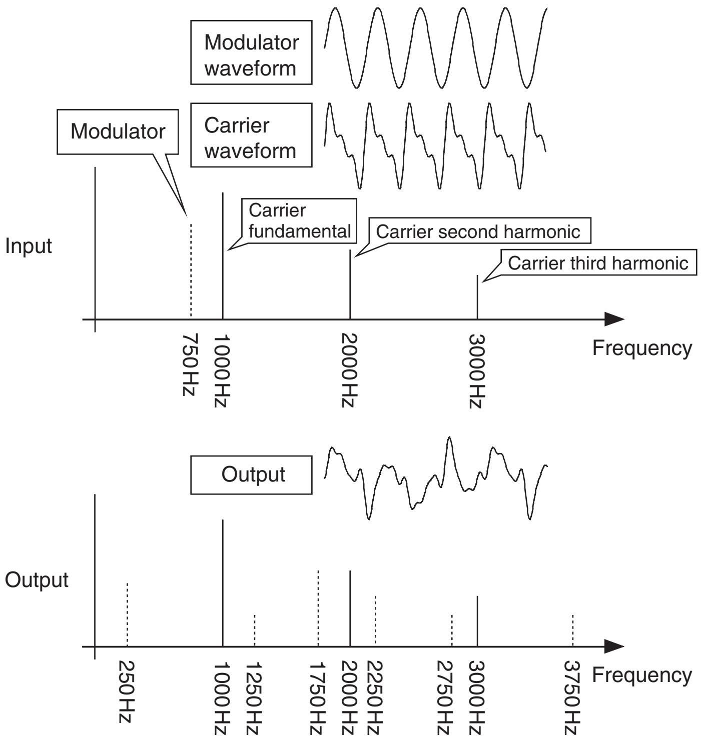
Amplitude modulation	Outputs	
Carrier 1000 Hz	1000 Hz	
Modulator: 750 Hz	250 Hz	1750 Hz
Modulator: 1500 Hz	500 Hz	2500 Hz
Modulator: 2250 Hz	1250 Hz	3250 Hz

FIGURE 3.5.2
If the modulator is a non-sinusoidal waveform, then each of the harmonics of the modulator produces a pair of sum and difference frequencies in the output.

output signal contains not only the original carrier frequency, but also the sum and difference sidebands for each of the multiples of the modulator frequency.

For audio FM with two sine waves, the output consists of the carrier frequency and sidebands made up from the sum and difference frequencies of the carrier and multiples of the modulator frequency. The number of sidebands depends on the modulation index (Figure 3.5.4), and a rough approximation is that there are two more than the modulation index. The modulating frequency is not present in the output. The amplitudes of the sideband frequencies are determined by a set of curves called Bessel functions (Chowning and Bristow, 1986).

For FM with waveforms other than sine waves, each component frequency is treated separately. So for a sawtooth carrier and a sine wave modulator, the



Amplitude modulation	Outputs	
Carrier 1000 Hz	1000 Hz	
Modulator: 750 Hz	250 Hz	1750 Hz
Modulator: 1500 Hz	1250 Hz	2750 Hz
Modulator: 2250 Hz	2250 Hz	3750 Hz

FIGURE 3.5.3
If the carrier is a non-sinusoidal waveform, then each carrier harmonic appears in the output and also produces a pair of sum and difference frequencies.

output is similar to the sawtooth AM case, but there are many more sidebands produced. FM is thus a very powerful technique for producing complex spectra, but in an analogue synthesizer it suffers from problems related to the frequency stability of the carrier and modulator VCOs, and the response of the carrier VCO to FM at audio frequencies.

In an analogue synthesizer, FM is produced by connecting one VCO to the frequency control input of another VCO. If the modulating frequency is lower than about 25 Hz, then FM is known as vibrato, and it is perceived as a cyclic change in pitch. FM is described in more detail in Section 5.1.

3.5.3 Ring modulation

Ring modulation takes two audio signals and combines them together in a way that produces additional harmonics. It uses a circuit known as a 'balanced

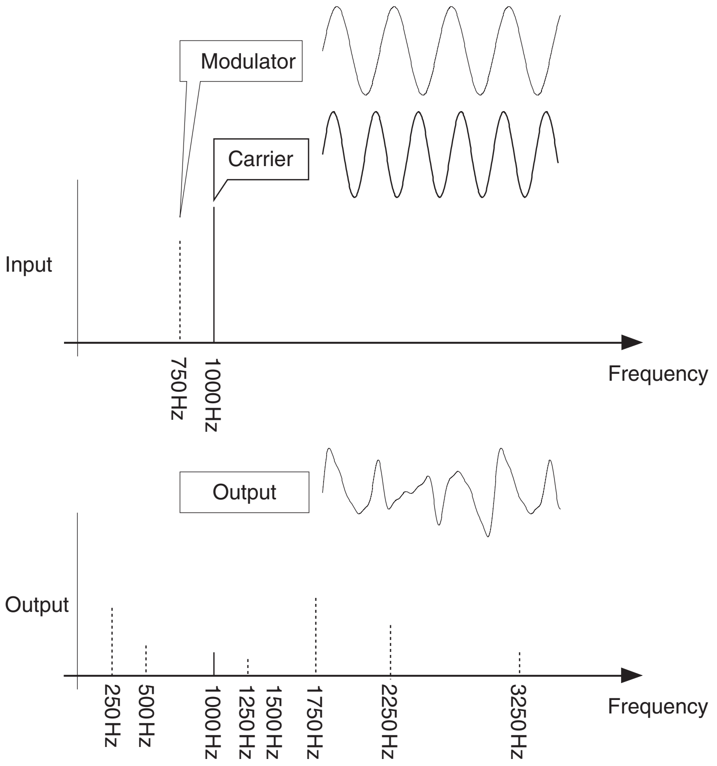
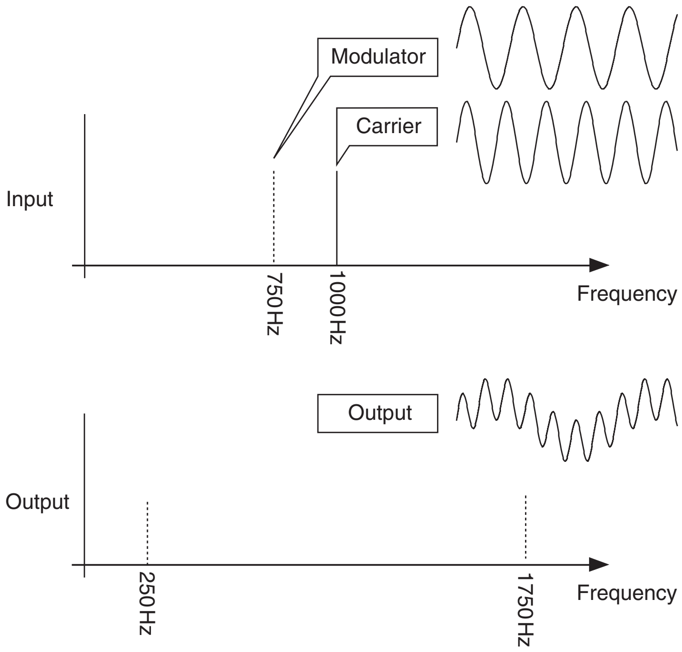


FIGURE 3.5.4
FM depends on the depth of modulation as well as the input frequencies. The number of sidebands that are produced depends on the modulation index.

Frequency modulation,	Outputs	
Carrier 1000 Hz, Modulator 750 Hz, Modulation index = 2	1000 Hz	
1st sidebands	250 Hz	1750 Hz
2nd sidebands	500 Hz	2500 Hz
3rd sidebands	1250 Hz	3250 Hz

modulator' to produce a single output from two inputs: the output consists of the sum of the two input frequencies and the difference between the two input frequencies. The original inputs are not present in the output signal (Figure 3.5.5).

This is similar to AM, except that it is only the additional frequencies that are generated which are present at the output: only the sidebands are heard, not the carrier or the modulator. This means that ring modulation can be useful where the original pitch information needs to be lost, which makes it useful for pitch transposition, especially where one of the sets of extra frequencies can be filtered out. In an analogue synthesizer, ring modulation is produced by a special modifier circuit.



Ring modulation	Outputs	
Carrier 1000 Hz		
Modulator: 750 Hz	250 Hz	1750 Hz

FIGURE 3.5.5
Ring modulation produces only the sum and difference frequencies – neither the carrier nor the modulator frequencies are present at the output.

Table 3.5.1 Modulation Summary

AM	carrier in output	Simple sidebands for sine waves
FM	carrier in output	Multiple sidebands for sine waves
RM	no carrier in output	Simple sidebands for sine waves

Modulation summary

Modulation summary is given in Table 3.5.1.

3.5.4 Formant synthesis

Formant synthesis is intended to emulate the strong resonant structure of many real instruments, where the spectrum of the output sound is dominated by one or more formants. Some analogue synthesizers have a simple high-pass filter after the low-pass filter to give some additional control over the bandwidth of sounds, and thus a simple type of formant.

In a formant synthesizer, this extra filtering is extended further: a graphic equalizer or complex filter is used to provide control over the bandwidth of the sound in addition to a VCF and VCA. Several parallel sections may be used to

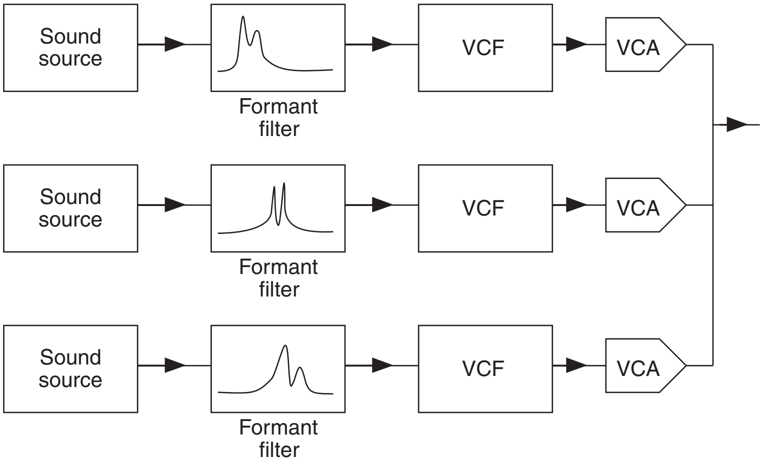


FIGURE 3.5.6 A formant synth is intended to emulate the resonance found in real instruments. This can be achieved by using formant filters in addition to VCFs and VCAs.

enable more detailed control over the individual formant areas of the sound (Figure 3.5.6).

3.5.5 Damped oscillators and ringing filters (drum sounds)

Circuits that have a strong resonance at a specific frequency can be made to oscillate if a sudden input causes them to self-oscillate. This 'ringing' is usually a sine wave and it dies away at a rate which is dependent on how close to self-oscillation the circuit is. The nearer it is to oscillating, the longer the ringing will last. Some VCFs can be made to self-oscillate if their *Q* or resonance is high enough, and at *Q* values just below this, they will ring. Conversely, an oscillator can be 'damped' so that it does not self-oscillate, but it will then ring. Filters and oscillators are just different applications of resonant circuits.

Decaying sine waves are very useful for producing percussive sounds, and many of the drum sounds produced by rhythm machines in the 1970s and early 1980s were produced by using ringing circuits (Figure 3.5.7).

3.5.6 Organ technologies

Most traditional organs are based around additive synthesis techniques, where a large number of sine waves are produced from a master oscillator, and then individual notes select mixes of sine waves through drawbar or other controls for the harmonic content (Figure 3.5.8).

Unlike additive synthesizers, until the middle of the 1980s, organs tended not to have envelope control over the individual harmonics which make up the sounds. The advent of digital technology and sampling has made organs much more closely related to sample and synthesis synthesizers. Chapter 3 gives

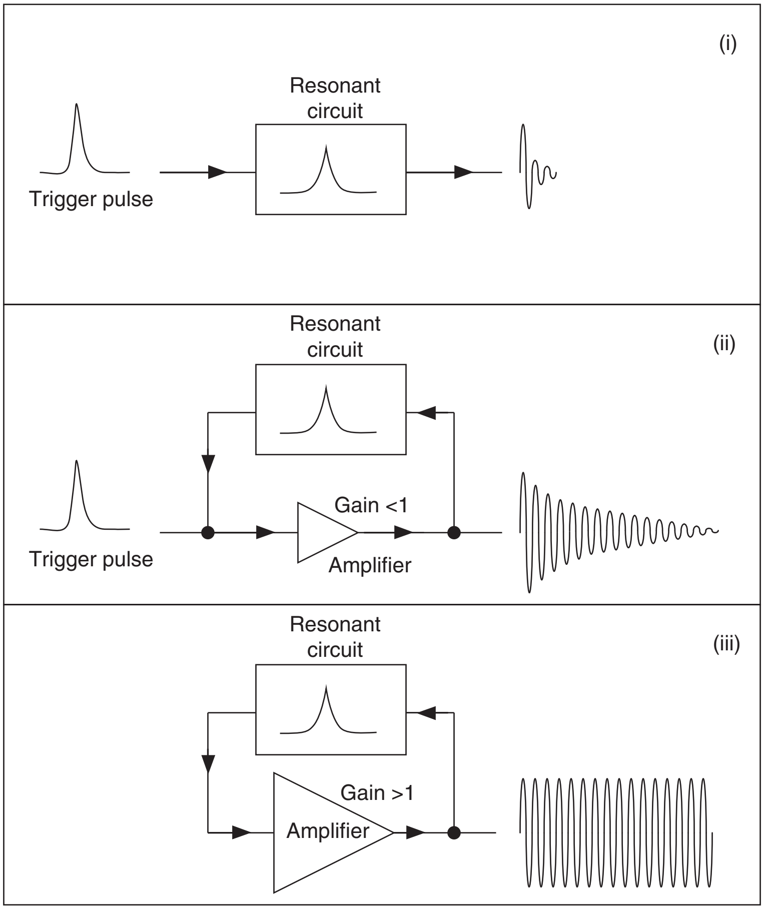


FIGURE 3.5.7 (i) A resonant circuit can produce some ringing when a trigger pulse is applied. (ii) When a resonant circuit is placed in the feedback loop of an amplifier with a gain of less than one, then the ringing of the resonant circuit is enhanced. (iii) If the gain of the amplifier is greater than one, then the circuit will oscillate at the frequency of the least attenuation in the resonant circuit.

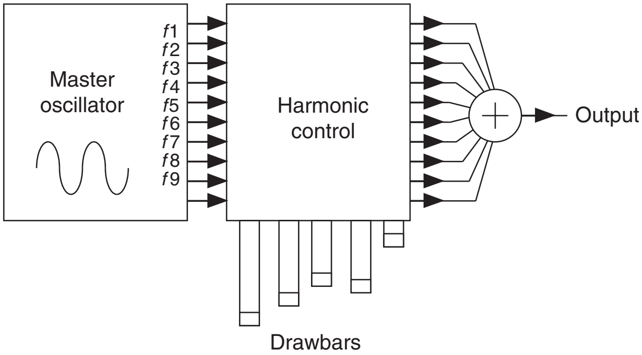


FIGURE 3.5.8 Organs typically produce sounds by the addition of sine waves. The methods of producing the sine waves can be mechanical, electromechanical and electronic.

further details of digital master oscillators, whilst Chapter 4 describes sample and synthesis in more detail.

3.5.7 Piano technologies

Before digital sampling technology, piano-type sounds were produced by taking square or rectangular waveforms, often derived from a master oscillator by a divider technique, and then applying a percussive envelope and filtering. This produces a completely polyphonic instrument, although the sound suffers from the same lack of dynamic individual harmonic control as organs of the same time period. By using narrow pulse waveforms and different envelopes, the same techniques can be used to produce string-like sounds, and this was used in many 1970s 'string machines'. Section 4.5.3 describes 'beehive noise', a side effect of this sound generation technique.

By the mid-1980s, separate 'stand-alone' dedicated string machines had been replaced by polyphonic synthesizers, with the typical electronic piano becoming a specialized sample-replay device by the end of the 1980s (Figure 3.5.9).

3.5.8 Combinations

Some analogue synthesizers use a combination of synthesis techniques, for example, where several oscillators are used (additive style) to provide the sound source, although this is then followed by a conventional subtractive synthesis modifier section. Ring modulation is another method which sometimes appears in otherwise straightforward implementations of subtractive synthesizers – perhaps because it is relatively simple to implement, and yet allows a large range of bell-like timbres that contrast well with the often more melodic subtractive synthesis timbres. Some 'string machines' in the 1970s added a

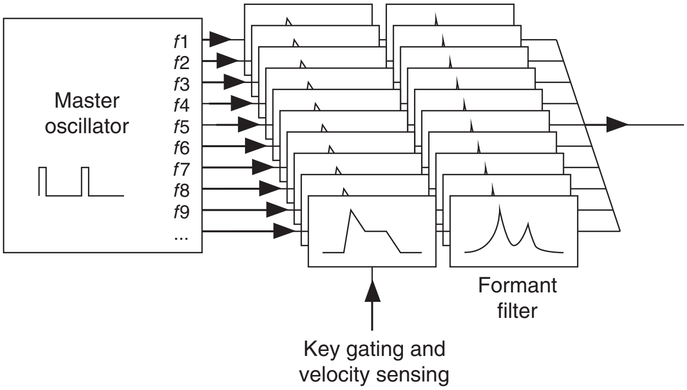


FIGURE 3.5.9 Simple 'piano' and 'string' type sounds can be produced by gating and filtering pulse waveforms which are derived from a master oscillator.

VCF and ADSR EG section to provide 'synth brass' capabilities. Such combinations can provide additional control and creative potential, although their additions rarely become adopted generally.

3.5.9 Tape techniques

Perhaps the most straightforward method of analogue synthesis is the use of the tape recorder. By recording sounds onto magnetic tape, they can be stored permanently for later modification and manipulation. The raw sounds used can be either natural or synthetic. Chapter 4 details the use of tape as a recording medium, whilst Chapter 1 outlines some of the creative possibilities of using tape as a synthesis tool.

3.5.10 Optical techniques

Whilst tape offers a large number of possibilities for manipulating sound once it has been recorded on the tape, it does not allow the user to generate or control a sound directly. The audio signals are recorded onto the tape as changes to the magnetic fields stored on the iron oxide coating of the plastic tape, and so cannot be seen or changed, other than by recording a new sound over the previously recorded audio signal.

In contrast, by using the optical soundtracks that are often used in film projectors, it is possible to directly input the raw sound itself. Modern film projectors can use magnetic or digital techniques as well, but the basic method uses a light source and an optical sensor on either side of the film. When the soundtrack is clear, all the light passes through the film to the sensor, and conversely, when the film is dark, then no light passes to the sensor. By varying the amount of light that can pass through the film to the light sensor, the output of the sensor can be controlled. If the film soundtrack varies at a fast enough rate, then audio signals can be produced at the output of the sensor. Film soundtracks usually control the amount of light by altering the width of the clear part of the film – the wider the gap, the more light passes through to the sensor. The part of the film used to record this 'sound' track is by the side of the picture, and looks much like an oscilloscopic view of an audio signal, except that it is mirrored around the long axis (Figure 3.5.10).

By taking film that has no sound recorded onto it, and then drawing onto the film soundtrack with an opaque ink, it is possible to create sounds that will only be heard when the film is played. Sounds can thus be drawn or painted directly onto film. Although this sounds like an effective marriage of art and science, it turns out that the process of drawing sounds by hand is a slow and tedious one, and the precision required to obtain consistent timbres is very high. The rough '30-dB rule of thumb' that says that a drawn audio waveform represents only the most significant 30 dB of the harmonics is very relevant here. Combining the drawing skills of optical sound creation with the tape manipulation processes of music concrete can offer a much more versatile technique. In this case, only short segments of film soundtrack need to be drawn,

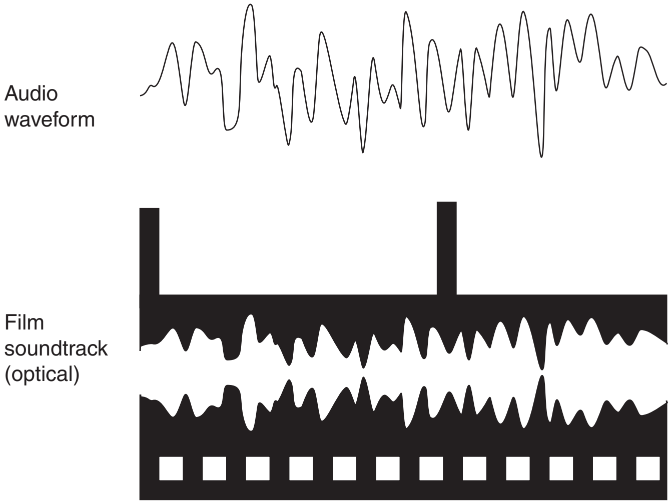


FIGURE 3.5.10 A film soundtrack uses the amount of light passing through the film to represent the audio waveform.

since the resulting short sounds can be recorded onto tape, copied many times to provide longer sounds and then manipulated using tape techniques.

3.5.11 Sound effects

Perhaps the ultimate 'analogue' method of synthesizing sounds is the work of the 'sound effects' team in a film or television studio. Using a floor covered with squares containing various surfaces, and a large selection of props, 'Foley' artistes produce many of the everyday sounds that accompany film and television programs. For more unusual 'spot' effects, specialized props or prerecorded sound effects may be used. Choreographing the sound effects for a detailed scene can be a very complex and time-consuming task, very similar to controlling an orchestra!

3.5.12 Disk techniques

Using a turntable, slipmat and a robust cartridge can also be a flexible and versatile analogue method of sound generation. Since the 1980s, the use of the vinyl disk as a source of complex sound effects, rhythms and musical phrases has become increasingly significant, and this has happened alongside the use of samplers (see also Chapter 8).

3.6 Topology

How do the component parts of a synthesizer fit together? This section starts by looking at typical arrangements of VCOs, VCFs, VCAs and EGs. It then looks at categorizing types of synthesizers: the main divisions in type are between monophonic and polyphonic synthesizers, performance and modular synthesizers, and alternative controllers. This section deals with the topology of the modules that make up a typical synthesizer – how they are arranged and ordered. Although this information is fundamental to the actual construction of analogue synthesizers, the theory behind it is also relevant to some digital instruments, even though digital synthesizers often have no physical realization of the separate modules at all.

3.6.1 Typical arrangements

The most common arrangement of analogue synthesizer modules is based on the 'source and modifier' or 'excitation and filter' model. This uses one or more VCOs plus a noise generator as the sources of the raw timbre. It then uses a VCF and VCA controlled by one or more EGs to shape and refine the final timbre. An LFO is used to provide cyclic modulation: usually of the VCO pitch (Figure 3.6.1).

This basic arrangement of modules is used so often by manufacturers that it has become permanently hard-wired into many designs, even some modular systems! The use of 'normalized' jack sockets allows for this type of preset wiring where the insertion of a plug into the socket opens the switch and removes the hard-wiring and thus allowing it to be overridden and replaced. 'Hard-wiring' is also used in many digital designs where there is no need for a rigid arrangement of modules because they are implemented in software.

One alternative method of subtractive synthesis replaces the single VCF with several. This enables more specific control of portions of the sound spectrum

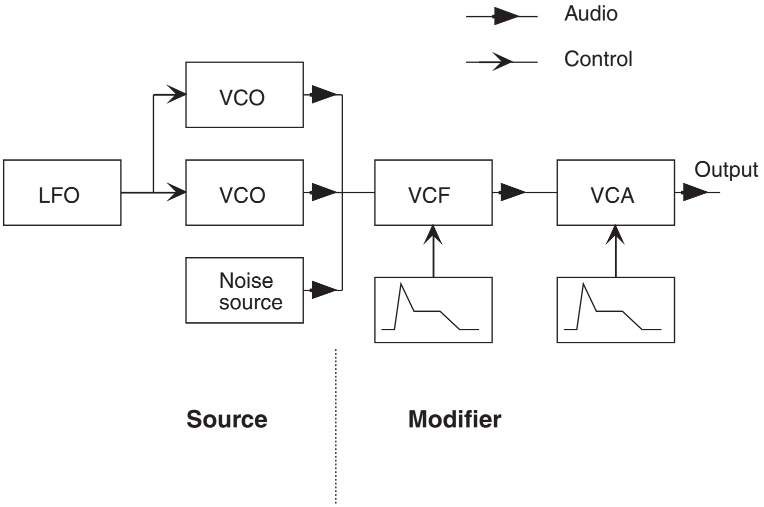


FIGURE 3.6.1 The basic synthesizer patch uses one or more VCOs and a noise generator as the sound source, with an LFO to provide vibrato modulation. The modifier section comprises a VCF and a VCA, both controlled by one or more EGs.

and is often associated with the use of band-pass rather than low-pass filters. Because having separate filters for the oscillators enables them to be used as components of the final sound, rather than as a single source processed through a single modifier, this paralleling of facilities can be much more flexible in its creative possibilities. It is often used in formant synthesis, where the aim is to emulate the peaks in frequency response which characterize many real-world instruments, and particularly the human voice. Additive synthesis is an extension of this formant synthesis technique, where additional VCOs, VCFs and VCAs are added as required. Ganging of EGs by using voltage control of the EG parameters can make the control easier.

By using one VCO to modulate another, FM synthesis can be used, although the limitations of the VCO tuning stability and scaling accuracy limit its use. By using VCFs to process the outputs of each VCO, the FM can be dynamically changed from using sine waves to using more complex waveshapes by increasing the cut-off frequency of the VCF on the output of the modulation VCO. This is something which most commercial digital FM synthesizers cannot do!

The basic synthesizer patch varies between monophonic and polyphonic synthesizers. It is often simplified for use in polyphonic synthesizers: only one VCO and VCA, and often less controllable parameters. Custom 'synth-on-a-chip' ICs are often used to implement polyphonic synthesizer designs, and these chips are based on a minimalist approach to the provision of modules and parameters.

3.6.2 Monophonic synthesizers

Monophonic synthesizers tend to be performance-oriented instruments designed for playing melodies, solos or lead lines. Despite the name, many monophonic analogue instruments can actually play more than one note at once: many have a duophonic note memory that allows two different note pitches to be assigned to two VCOs. With only one or two notes capable of being played simultaneously, an assignment strategy is required so that any additional notes played can be dealt with in a predictable way.

Two common schemes are last-note and low-note priorities. Last-note priority is a time-based scheme, which always assigns the most recently played note to the synthesizer's voice circuitry, whilst low-note priority is a pitch-based scheme, which always assigns the lowest pitched note to the voice circuitry. Low-note priority can be a powerful performance feature; for example, the performer can play legato 'drone' notes with the thumb of their right hand and use the rest of the fingers to play runs on top, with staccato playing dropping back to the 'drone' note. This technique is most effective with envelopes that are not retriggerable; that is, they do not restart the attack segment each time a new key is pressed on the keyboard. See Chapter 7 for more details on keyboard design and note assignment.

Portamento is a gliding effect which happens between notes. On a monophonic synthesizer it is normally used as a performance effect to give a contrast between the sudden pitch transition between notes and the slow change

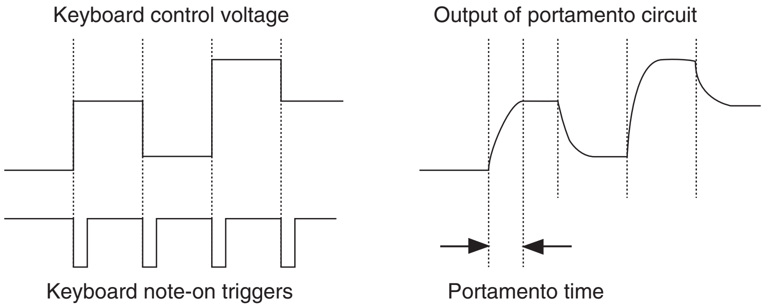


FIGURE 3.6.2 Portamento provides a smooth transition between successive pitches from the VCOs. The time taken for the keyboard CV to change from the previous value to the new value is called the portamento time.

of a portamento. The portamento circuits in analogue synthesizers work by restricting the rate at which a CV can change. Normally, the pitch CV from a keyboard will change rapidly when a new note is selected. A portamento circuit changes the slope of the transition between the two voltages. It thus takes time for the note to move from the existing pitch to the new pitch (Figure 3.6.2).

Glissando is a rapid movement from one note to another where the pitch changes chromatically through all the notes in between. At fast speeds, glissandos sound similar to portamento.

Monophonic synthesizers normally arrange the front panel controls so that they form a logical arrangement, often mimicking the topology of the modules inside. The front panel is normally arranged so that sources and controllers are on the left, with modifiers and the final output on the right. Early analogue monophonic synthesizers, and most modular systems, do not have any form of memory for the positions and settings of the front panel controls, and so a clear and functional arrangement of controls can aid the user in remembering settings. The process of using such a synthesizer requires a lot of practice to become thoroughly familiar with the workings of the instrument. Recalling a sound is often achieved iteratively, with adjustments of the controls gradually homing in on the required sound. Individuals who have mastered a synthesizer in this way have many similarities to a classically trained instrumentalist, where the way to produce a sound from the instrument requires dexterity, skill and a degree of coaxing.

By the end of the 1970s, memory stores for the rapid recall of front panel settings had begun to appear, and by the end of the 1980s almost all monophonic synthesizers were equipped with memories. Front panels began to reflect this change by concentrating more on simplifying both the recall of memories and making simple minor edits to them. Many synthesizers became simply replay machines for preset sounds and for many users, their programming changed from being part of the performance art to being an unwanted chore. By the late 1990s, live editing of sounds had become fashionable again, and synthesizer design

Oberheim's OB1 monophonic synthesizer from 1978 had memories, but it is perhaps more famous for allegedly inspiring a character name from the first (IV) 'Star Wars' movie.

reflected this with an increasing number of designs that included more controllers. In the first years of the twenty-first century, a number of manufacturers released synthesizers which were modern recreations of their own instruments from about 20 years before, but with memories and additional performance controls.

The performance controls on monophonic analogue synthesizers are mono-oriented: pitch-bend (often set to an interval of a fifth or an octave); octave switch (up or down, one or two octaves: often to compensate for a small keyboard span); modulation (normally vibrato) and occasionally, after-touch (normally controlling vibrato). For those instruments that do have front panel controls, they can be used as an additional method of control: real-time changes to sounds can be made 'live'. This usage of front panel and performance controls arises from the monophonic nature of the keyboard. For a right-handed player, the right hand is used to play the keyboard, whilst the left hand is used to provide additional expression by manipulating the pitch-bend and modulation wheels. 'Classical' two-handed static position techniques for playing monophonic melodies are rarely seen; and instead, a flowing right-hand movement with lots of crossovers is used, thus freeing the left hand for the performance controls. Left-handed versions of monophonic synthesizers are very rare indeed: the placement of the performance controls is invariably on the left side of the keyboard (Figure 3.6.3).

3.6.3 Polyphonic synthesizers

Polyphonic analogue synthesizers are often implemented as several monophonic synthesis 'engines' or 'voices' connected to a common polyphonic keyboard. Each of these 'voices' receives monophonic note-pitch voltage, gate and trigger information, and performance controller information. It is usual for each voice to produce the same sound or timbre: multi-timbrality is normally

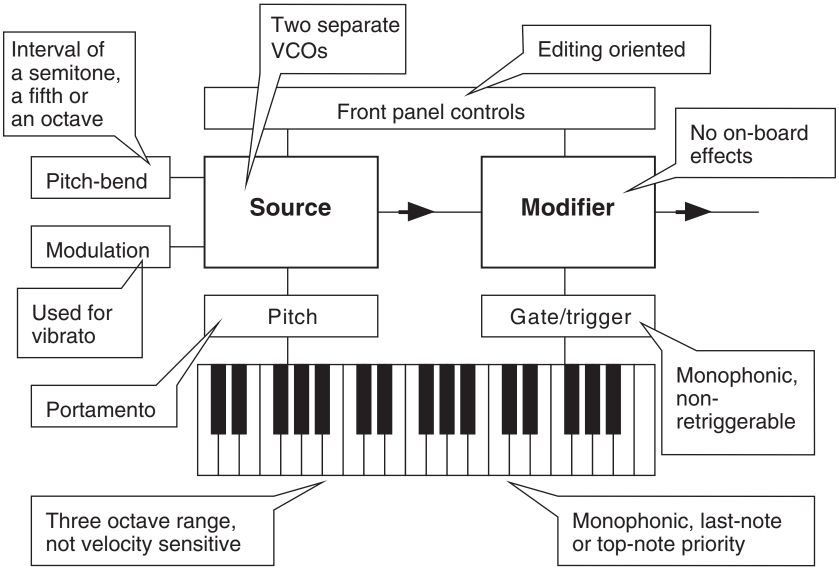

Two separate VCOs
Interval of a semitone, a fifth or an octave
Editing oriented
Front panel controls
No on-board effects
Pitch-bend
Source
Modifier
Modulation
Used for vibrato
Pitch
Gate/trigger
Portamento
Monophonic, non-retriggerable
Three octave range, not velocity sensitive
Monophonic, last-note or top-note priority

FIGURE 3.6.3
A summary of the main features of a typical analogue monophonic synthesizer of the 1970s.

commonly found in digital instruments only. The assignment of the voices to the keys which are played on the keyboard is carried out by key assignment circuitry or software in the polyphonic keyboard. This deals with the reassignment of notes which are playing (note stealing) and the method of assigning notes to the voices (last-note priority, etc.). (More details of keyboards can be found in Chapter 7.)

Controlling portamento on a polyphonic synthesizer is much more complex than on a monophonic synthesizer. The transitions between several notes can be made using several portamento algorithms. These are often named according to the effective polyphony that they produce, although in practice, only short portamento times are used to give a slight movement of pitch at the beginning of notes: this is frequently used for vocal, brass and string sounds.

Longer portamento times do not suit polyphonic keyboard technique, except for special effect usage with block chords, and often a glissando is more musically useful – where all the notes in between the last-note played and the next are played in sequence.

The Roland Juno-6 and Juno-60 memory versions illustrate this well, since the follow-up model the Juno-106 was only available with memories.

Memory stores seem to be more widespread in early polyphonic analogue synthesizers than in their monophonic equivalents. Initially, some manufacturers produced low-cost polyphonic synthesizers without memories, but these were not very popular in comparison with their more expensive memory-equipped versions. The designers of polyphonic synthesizers seem to have placed more emphasis on the accessibility of the memory recall controls than on the front panel controls for programming the synthesizer voices. The Yamaha CS-80 demonstrates this principle in its design: the programmable memories are hidden underneath a flap, and have tiny controls, whilst the large, colorful memory recall buttons are handily placed right at the front of the control panel.

The performance controls on polyphonic analogue synthesizer tend to be optimized for polyphonic playing techniques. Pitch-bend is normally only a semitone, and can often be applied to only the top note or the last note which has been played on the keyboard. The modulation wheel is often replaced or paralleled by a foot pedal and often controls timbre through the VCF cut-off rather than vibrato. After-touch is almost invariably used to control vibrato or tremolo, and some instruments provide polyphonic after-touch pressure sensing instead of the easier-to-implement global version. Some instruments have an LFO which is common to all the voices, and so vibrato or tremolo modulation is applied at exactly the same frequency and phase to all the voices. In contrast, instruments which use separate LFOs for each voice circuit will have slightly different frequencies and phases, and this can greatly improve string and vocal sounds.

Real-time changes to the timbres are normally made using additional controllers: foot pedals, foot switches and breath controllers.

Manipulating front panel controls whilst playing with both hands on a polyphonic keyboard seems to be unpopular, and if front panel controls are used then the playing technique used often reverts to monophonic usage, as described earlier – although polyphonic keyboards almost always have retriggered envelopes, which restrict some performance techniques. The performance

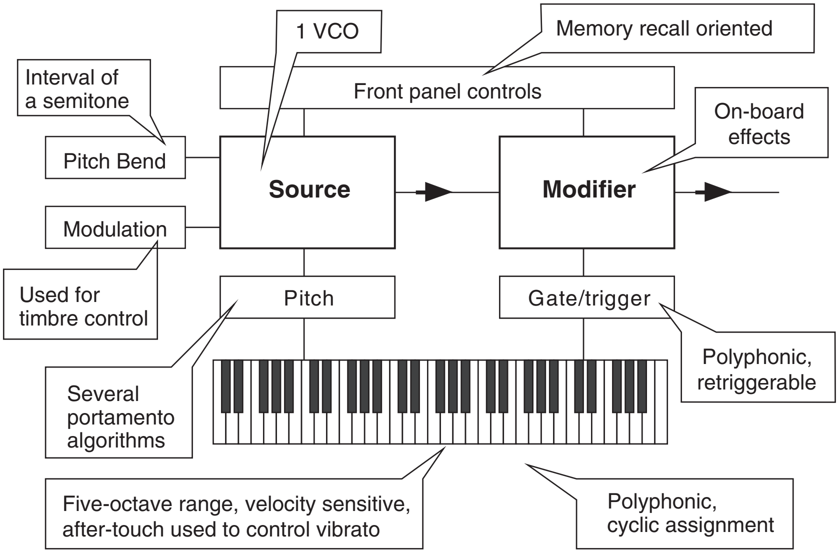


FIGURE 3.6.4
A summary of the main features of a typical analogue polyphonic synthesizer of the 1980s.

controls are placed on the left-hand side of the keyboard, just as with monophonic synthesizers (Figure 3.6.4).

3.6.4 Performance versus modular synthesizers

Performance synthesizers (monophonic or polyphonic) need a simplified and ordered control panel in order to make them usable in live performance. For this reason they usually have a fixed topology of modules: VCO, VCF and VCA.

Analogue modular synthesizers are not arranged in a logical order because there is no way to anticipate what they will be used for, except for the simplest cases. The most usual arrangement has the oscillators and other sound sources grouped together, usually on the top or on the left, with the modifiers (filters and amplifiers) in the center or middle and the EGs on the right or bottom.

Performance instruments have memories that can be used to store and recall sounds or timbres quickly. They are often used as replay machines for a series of presets.

Modular synthesizers normally have no memory facilities, or very simple generic ones which do not have the immediacy of those found in polyphonic instruments (Figure 3.6.5).

Performance synthesizers have modules arranged in a way that enables quick results: VCO–VCF–VCA with EGs. Modular synthesizers have few preset connections, if any, and so whilst it is quick and easy to connect a VCO to an amplifier and get a sound which will play until you turn the VCO or amplifier off, it can take some time to get a sound from a modular synthesizer which can be used in conventional performance.

It has been said that modular synthesizers are the ultimate synthesizers and that it is only time that limits people's use of them. Actually, modular

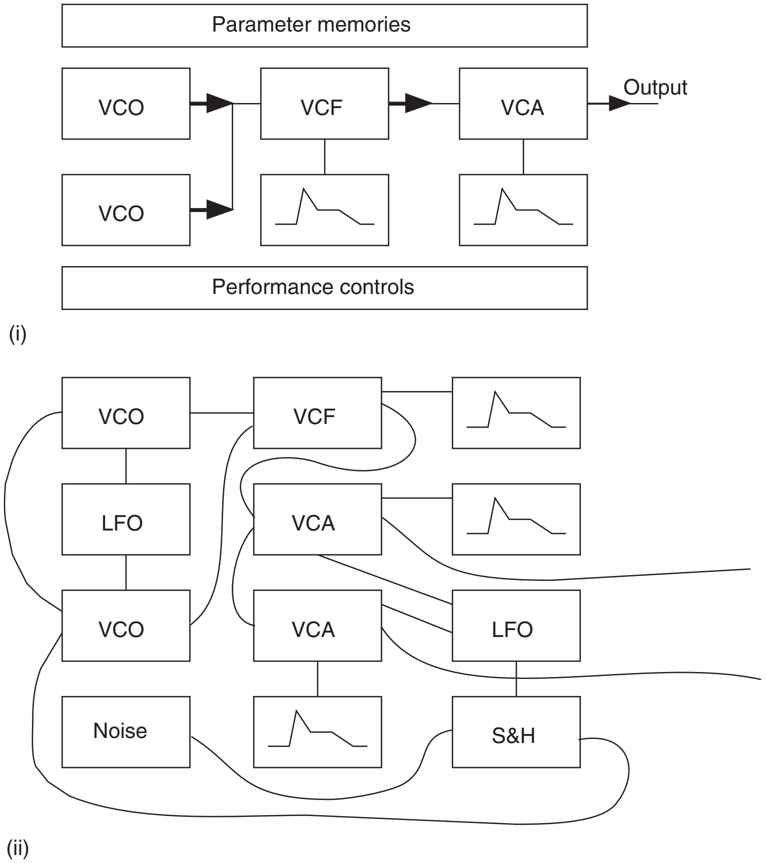


FIGURE 3.6.5 (i) A performance-oriented synthesizer is designed to rapidly recall stored sounds and allow detailed performance effects to be applied with a range of specialized controllers. (ii) A modular synthesizer provides a wide range of modules which provide great flexibility, but at the expense of complexity and ease-of-use.

synthesizers are severely limited by a combination of the design and the user. The design is limited by the problems of trying to cope with patch-leads and lots of controls underneath. The user is fully occupied trying to hold everything about what is happening in their head: a simple VCO–VCF–VCA setup with a couple of EGs can be spread over more than a dozen modules and 20 or more patch-leads. The limitations are all too evident: no programmability, a confusing and obscure user interface and lots of scribbled sheets noting down settings and patches. They are also often write-only devices – once the user has produced a patch, coming back 3 months later and trying to figure out what is happening is almost impossible. It is often much faster to start all over again.

Modular synthesizers are very good for appearance. Large panels covered with knobs, switches and patch-cords can look very impressive on stage. In reality, modular synthesizers are very good at producing lots of variations on a very specific set of sounds, and not very much outside of that set. Some

FM-type sounds can be produced, but not very usefully, since the VCO modulation at audio frequencies is often less than ideal. Filter sweeps have a nasty habit of getting bored too, and it is very easy to fall into the 'lots of synth brass sounds' cliché. And do not forget that beyond about 20 patch-cords, most people lose track of what is connected to what! Modular synthesizers can be considered as almost 'write-only' devices, where trying to work out what a patch does can be very difficult, especially if someone else did the patching. It is also often forgotten that despite the large number of modules which are available in many modular synthesizers, their polyphony is very limited: two or three notes and frequently only one note! Modular synthesizers are really not designed for polyphonic use, and trying to keep several separate sets of modules with anything like the same parameter settings is almost impossible. Although sampling the sounds that are produced using a modular synthesizer can be one way of producing a polyphonic sound and having programmability, given the synthesis power of many hardware and software samplers, the modular synthesizer is almost redundant even for this application.

Perhaps the most persuasive argument for the limited timbre palette of modular synthesizers is stored forever in recordings of the early 1970s. The problems of keeping track of patch-cords, the very limited polyphony, trying to avoid sweeping filter clichés, attempting to stay in control of the sound, the complete lack of any memory facilities and other limitations all conspire to make modular synthesizers an expensive chore.

Of course, from a very different viewpoint, modular synthesizers are collectable and may well be sought very well after in the future as 'technological antiques'.

How to do individual vibrato on specific notes? Take two synthesizers and set them to the same sound. Now play the non-vibrato notes on one keyboard and the notes requiring vibrato on the other, using after-touch to bring in the vibrato (or just set the modulation wheel with a preset amount of vibrato) – simple but effective and a challenging test of two-handed playing technique (or sequencer programming). The inventive reader is encouraged to find other 'two-synth' solutions to 'And you can't do that on a synthesizer!' challenges.

3.6.5 Keyboards versus other controllers

Most synthesizers come with a keyboard. Most expander modules are equipped with MIDI input, which is a strongly keyboard-oriented interface. Many of the controls on a typical synthesizer are monophonic keyboard-oriented: pitch-bend, modulation, keyboard tracking, after-touch, key scaling,…

Alternative controllers often have different parameters available which are not keyboard related. Stringed instruments such as violin and cello have control over the pressure of the bow on the string in a way which is analogous to velocity and after-touch combined. Guitars enable the performer to use vibrato on specific notes: something which is very difficult on most keyboard-based synthesizers. Woodwind instruments have a number of performance techniques that do not have a keyboard equivalent – like pitch-bending, changing the timbre or producing harmonics, all by using extra breath pressure and lip techniques. (Additional information on controllers can be found in Chapter 7.)

3.7 Early versus modern implementations

Electronics is always changing. Components, circuits, design techniques, standards and production processes may become obsolete over time. This means that the design and construction of electronic equipment will continuously change as these new criteria are met. The continuing trend seems to be for smaller packaging, lower power, higher performance and lower cost but at the price of increasing complexity, embedded software, difficulty of repair and rapid obsolescence. Over the last 25 years, the basic technology has changed from valves and transistors towards microprocessors and custom ICs.

3.7.1 Tuning and stability

The analogue synthesizers of the late 1960s and early 1970s are infamous for their tuning problems. But then so are many acoustic instruments!

In fact, it was only the very earliest synthesizers that had major tuning problems. The first Moog VCOs were relatively simple circuits built at the limits of the available knowledge and technology – no one had ever built analogue synthesizers before. The designs were thus refined prototypes which had not been subjected to the rigorous trials of extended serious musical use. It is worth noting that the process of converting laboratory prototypes into rugged, 'road-worthy' equipment is still very difficult; and at the time, valve amplifiers and electromechanical devices such as tape echo machines were the dominant technology. Modular synthesizers were the first 'all-electronic' devices to become musical instruments that actually left the laboratory.

The oscillators in early synthesizers were affected by temperature changes because they used diodes or transistors to generate the required exponential control law, and these change their characteristics with temperature (diodes or transistors can be used as temperature sensors!). Once the problem was identified, it was quickly realized that there was a need for temperature compensation. A special temperature compensation resistor called a 'Q81' was frequently used – they have a negative temperature coefficient which exactly matches the positive temperature coefficient of the transistor.

Eventually circuit designers devised methods of providing temperature compensation, which did not require esoteric resistors, usually based around differential pairs of matched transistors. Developments of these principles into custom synthesizer chips have effectively removed the need for additional temperature compensation.

Unfortunately, the tuning problems had created a characteristic sound, which is one reason why the 'beating oscillator' sounds heard on vintage analogue synthesizers are emulated in fully digital instruments that have an excellent temperature stability.

Tuning problems fall into four categories:

1. overall tuning
2. scaling
3. high-frequency tracking
4. controllers.

Tuning polyphonic synthesizers requires patience and an understanding of the way that key assignment works (see Section 6.5.3). The tuner needs to know which VCO is making the sound (sometimes indicated by a light emitting diode (LED) or by a custom circuit add-on), as well as how to cycle through the remaining VCOs – often by holding one note down with a weight or a little wedge and then pressing and holding additional notes.

Because of the differences in the response of components to temperature, the tuning of an analogue synthesizer can change as it warms up to the operating temperature. This can be compensated manually by adjusting the frequency CV or automatically using an 'auto-tune' circuit (see later). Some synthesizers used temperature-controlled chips to try and provide elevated but constant temperature conditions for the most critical components: usually the transistors or diodes in the exponential converter circuits. These 'ovens' have been largely replaced in modern designs by careful compensation for temperature changes.

Temperature drift of the octave interval is the problem that most people mean when they say that analogue synthesizers go out of tune. Trying to match two exponential curves means that two interdependent parameters need to be changed: the *offset* and the *scaling*. The *offset* sets the lowest frequency that the VCO will produce, whilst the *scaling* sets the octave interval to get the doubling of frequency for each successive octave. On a monophonic instrument this is not so hard, and any slight errors only help to make it sound lively and interesting. For polyphonic analogue synthesizers, this process can be very time consuming and very tedious. With lots of VCOs to try and adjust, the problem can begin to approach piano tuning in its complexity.

One method used to provide an 'automatic' tuning facility for polyphonic analogue synthesizers was introduced in the late 1970s. A microprocessor was used to measure the frequencies generated by each VCO at several points in its range and then work out the offset and scaling correction CVs. Because of the complexity of this type of tuning correction, and its dependence on a closed system, it has never been successfully applied to a modular synthesizer. (Auto-tuning is covered in more detail in Section 4.3.)

High-frequency tracking is the tendency of analogue VCOs to go 'flat' in pitch at the upper end of their range. This is normally most noticeable when two or more VCOs are tuned several octaves apart, and although often present in a single VCO synthesizers, is only apparent when they are used in conjunction with other instruments. Most VCOs use a constant current source and an integrator circuit to generate a rising voltage and resetting the integrator when the output reaches a given voltage. This produces a 'sawtooth' waveform. The higher the current, the faster the voltage rises, and the sooner it will be reset, which produces a higher-frequency sawtooth waveform. At low frequencies, the time it takes to reset the integrator is not significant in comparison with the time for the voltage to rise. But at high frequencies the reset time becomes more significant until eventually the waveform can become triangular in shape, which means that only one part of the waveform is actually controlled by the current source, and so the oscillator is not producing enough high frequency (Figure 3.7.1). Some VCO designs generate a triangular waveform as the basic waveform and so do not suffer from this problem.

Controllers are another source of tuning instability. The stability of the pitch produced by a VCO is dependent on the CVs that it receives. So

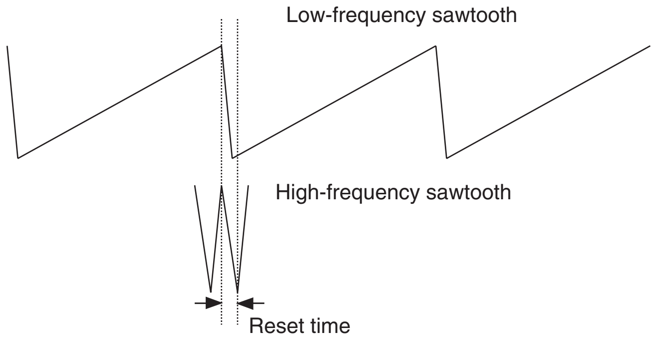


FIGURE 3.7.1 At low frequencies, the rising part of the sawtooth waveform is much longer than the fixed reset time. But at higher frequencies, the reset time becomes a significant proportion of the cycle time of the waveform and so the frequency is lower than it should be. This high-frequency tracking problem needs to be compensated for in the CV circuitry of the VCO.

anything mechanical that produces a CV can be a source of problem. Slider controls are one example of a mechanical control, which can be prone to movement with vibration, whilst pitch-bend devices with poor detents can cause similar 'mechanical' tuning problems. The detent mechanism varies. One popular method involves using the pitch-bend wheel itself – it has two of the finger notches opposite to each other. One is used to help the user's fingers grip the wheel, whilst the other is used to provide the detent – a spring steel cam follower clicks into place when it is in the detent and pops out again when the wheel is moved. This can wear, and produce wheels which do not click into position very reliably, which can mean that the whole instrument is then put out of tune.

3.7.2 Voltage control

As has already been mentioned several times, despite the name, most of the electronic circuitry used in synthesizers is actually controlled by currents, not voltage! The voltages that are visible in the patch-cords in the outside world are converted into currents inside the synthesizers and the control is achieved using these currents. Two 'standards' are in common usage:

1. 1 volt/octave
2. Exponential.

1 volt/octave

The 1-volt/octave system uses a linear relationship between the CV and pitch, which in practice means that there is a logarithmic relationship between voltage and frequency. This means that small changes in voltage become more significant at higher frequencies – just where small changes in pitch might become significant and audible as tuning problems. A 0- to 15-volt control signal can be used to control a pitch change of 15 octaves.

Exponential

The exponential system uses a linear relationship between the CV and the frequency. Because this method provides more resolution at high frequencies, it can be argued that it is a superior method to the 1-volt/octave system, since minor tuning errors at low frequencies are less objectionable. If the highest CV is 15 volts, then one octave down is 7.5, then 3.45, 1.875, 0.9375, 0.468,75, 0.234,375, 0.117,187,5, 0.058,593,75, and so on …, halving each time. Note that just eight octaves down a voltage change of 58 millivolts is equivalent to an octave of pitch change.

Despite the apparent advantage of the exponential system, the most popular method was the 1-volt/octave system. Conversion boxes that enabled interworking between these two systems were available in the 1970s and 1980s, but they are very rare now.

3.7.3 Circuits

VCO

The basic oscillator circuit for a VCO uses a current to charge a capacitor. When the voltage across the capacitor reaches a preset limit, then it is discharged, and the charging process can start again. This 'relaxation' oscillator produces a crude sawtooth output, which can then be shaped to produce other waveforms (Figure 3.7.2). By varying the current that is used to charge the capacitor, the time it takes to reach the limit then changes, and so the frequency of the

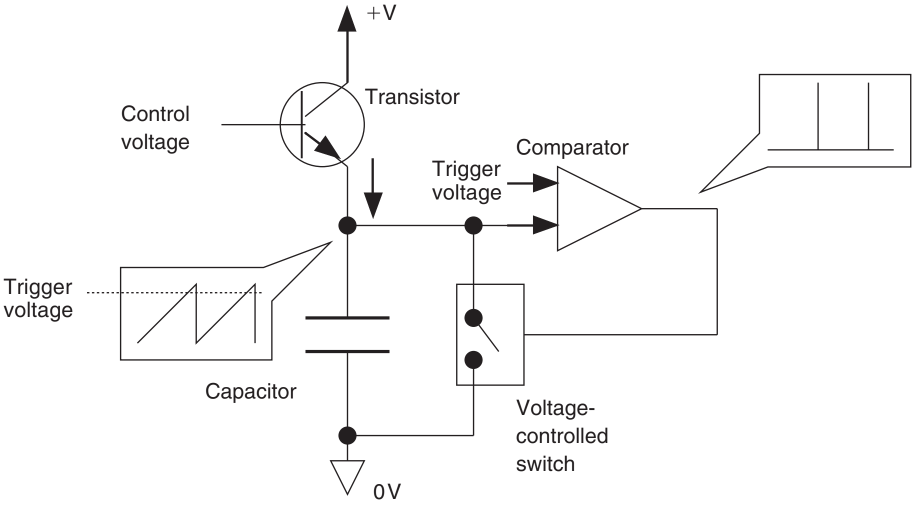


FIGURE 3.7.2 A relaxation oscillator circuit consists of a capacitor which is charged by a current, *i*, and discharged by a switch when the voltage across the capacitor reaches the point at which the comparator triggers. Two output waveforms are available: the sawtooth voltage from the capacitor and the reset pulses from the comparator.

oscillator changes. By using a voltage to control the current, perhaps with a transistor, the oscillator then becomes voltage controlled. This type of circuit forms the basis of many VCOs.

VCF

Simple low-pass filters use RC networks to attenuate high frequencies. By making the resistor variable, it is possible to alter the cut-off frequency. This RC network forms a single-pole filter, which has poor performance in terms of cut-off slope. Two- or four-pole filters improve the performance, but require more resistors and capacitors. This requires separate buffer stages and multiple variable resistors. One way to produce several variable resistors uses the variation in impedance of a transistor or diode as the current through it is varied. By arranging a cascade of RC networks, where the transistors or diodes have the 'voltage-controlled' current flowing through them, it is possible to make a low-pass filter whose cut-off frequency is controlled by the current that flows through the chain of transistors. This is the principle behind the 'ladder' filters used in Moog synthesizers.

The basic Moog-type filter uses two sets of transistors or diodes in a 'ladder' arrangement (Figure 3.7.3). The important parts of the filter are the base–emitter junction resistance and the capacitors that connect the two sides of the ladder. Current flows down the ladder, and the input signal is injected into one side of the ladder. Since the resistance of the junctions is determined by the current which is flowing, the RC network thus formed changes its cut-off frequency as the current changes. This gives the voltage (actually current) control over the filter.

Another type of filter which is found in analogue synthesizers is the 'state variable' filter. This configuration had been used in analogue computers since valve days to solve differential equations. Once op-amps were developed, making a state variable filter was considerably easier, and by using field effect transistors (FETs) or transconductance amplifiers the cut-off frequency of the filter could easily be changed by a CV.

A typical state variable filter is made in the form of a loop of three op-amps (Figure 3.7.4). It is a constant-*Q* filter. Three outputs are available: low-pass, high-pass and band-pass (a band-reject can be produced by adding a fourth op-amp). Other types of multiple op-amp filters can be made: the bi-quad is one example whose circuit looks similar to a state variable, but the minor changes make it a constant-bandwidth filter and it only has low-pass and band-pass outputs.

3.7.4 Envelopes

It has been said that the more complex the envelope, the better the creative possibilities. The history of 'the envelope' is one of the continuous evolution. The beginning lies with organ technology, where RC networks were used to try and damp out the clicks caused by keying sine waves on and off, and then the

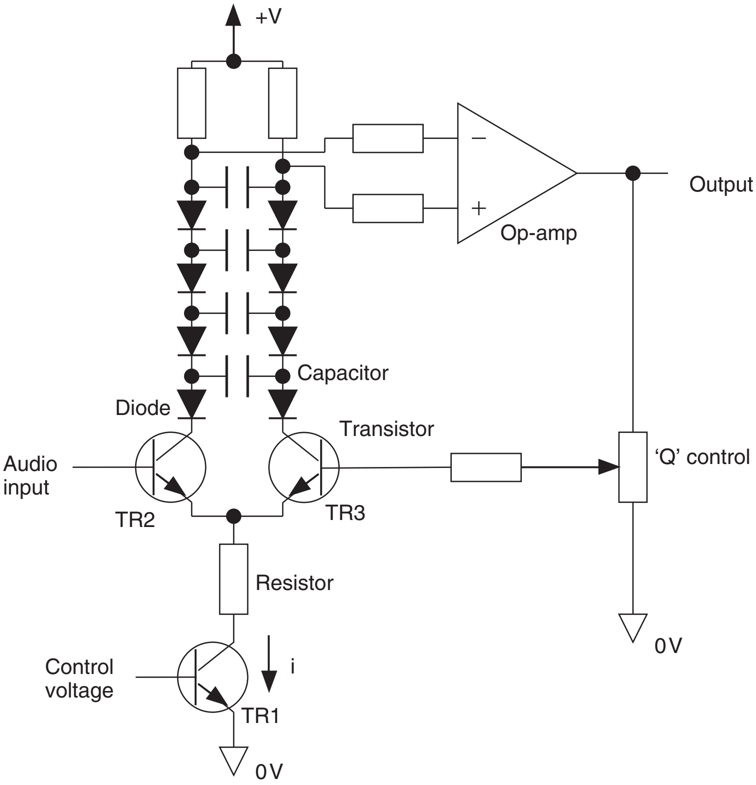


FIGURE 3.7.3 A typical 'ladder' filter. The current flows through the CV transistor, TR-1, and then through the two chains of diodes. The diodes and the connecting capacitors form RC networks which produce the filtering effect, with the diodes acting as variable resistors. The op-amp amplifies the difference between the two chains of diodes and feeds back this signal, thus producing a resonance or '*Q*' control.

clicks ended up being generated deliberately so that they could be added back in as 'key click'.

Trapezoidal waveform generators followed, which provided control over the start and finish of the envelope. ADSR-type envelopes, and their many variants, were used for the majority of the analogue synthesizers of the 1970s and 1980s. The advent of digital synthesizers with complex multi-segment EGs has made the ADSR appear unsophisticated, and analogue synthesizers designed in the 1990s have tended to emulate the multi-segment envelopes by adding additional break-points to ADSR envelopes.

The suitability of an envelope has very little to do with the number of segments, rates, times or levels. Instead, it is connected with the way that things happen in the real world. There are two things to consider:

1. Many instruments have envelopes with exponential attacks rather than the much easier to produce linear slopes which many analogue synthesizers use. One solution to this is to add in two or more attack

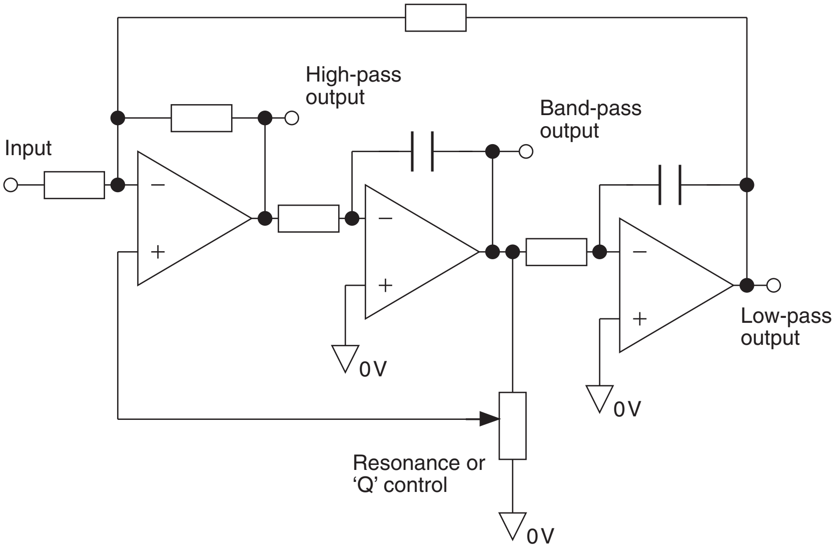


FIGURE 3.7.4 A typical 'two-pole' state variable filter. This produces three simultaneous outputs: high-pass, band-pass and low-pass.

segments and so produce a rough approximation of an exponential envelope. This is much easier to achieve in a digital instrument than in analogue circuitry.

2. Envelopes often change their shape and their timing in ways that are related to the note's pitch and the velocity with which it was played. Most modular and monophonic analogue synthesizers are not velocity sensitive, and so instruments that depend on this sort of performance technique tend to suffer (e.g. pianos). Changing the attack times with pitch can be quite complex in an analogue synthesizer – you need an EG with voltage-controlled time parameters, and this can require a large number of additional patch-cords and control knobs (Figure 3.7.5).

Sophisticated multi-segment envelopes suffer from being harder for the user to visualize the shape of the envelope being produced. Probably the best compromise is an ADSR with a couple of attack, decay and release segments, and control over the slopes: 'function' generators meeting this sort of design criteria are beginning to appear. EG design research is still ongoing.

3.7.5 Discrete versus integration

Early analogue synthesizers used individual transistors to build up their circuits. This 'discrete' method of construction was gradually replaced by ICs, usually op-amps for the majority of the analogue processing. Custom chips began to integrate large blocks of circuitry into single chips: a VCO or VCF, for example. Finally, by the mid-1980s, complete VCO, VCF, VCA, LFO and EG circuits could be placed on a single 'voice' chip intended for use in polyphonic

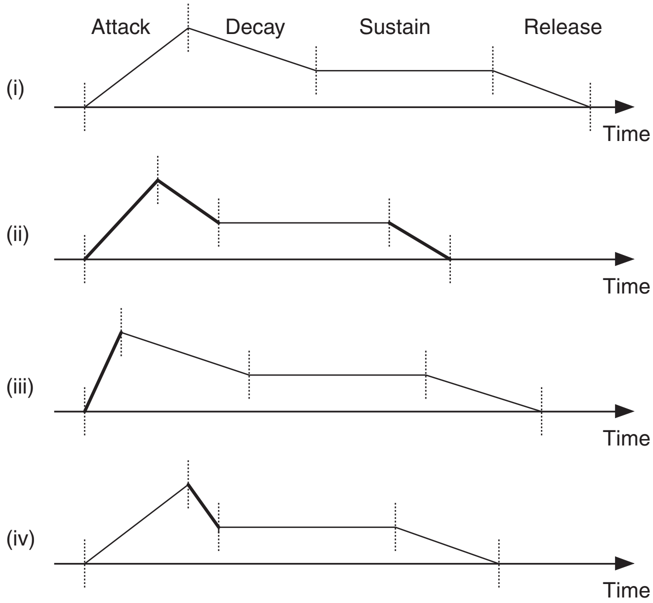


FIGURE 3.7.5 Envelope scaling using voltage control. (i) An ADSR envelope. (ii) The same envelope with the attack, decay and release times reduced proportionally. (iii) The same envelope with just the attack time reduced. (iv) The same envelope with just the decay time reduced. In order to produce each of these envelopes, a voltage-controlled EG would need both ganged (all time altered equally) and individual controls.

analogue synthesizers. In the 1990s, the VCO would probably be replaced by digital generation techniques, with analogue filtering and enveloping from VCF and VCA chips. The specialist chips that are used can become collectors' items, particularly some of the older and rarer designs.

3.7.6 Pre- and post-MIDI

The development of MIDI signaled a major change in synthesizer technology (Rumsey, 1994). At a stroke, many of the incompatibility problems of analogue synthesizers were solved. CVs; gates and trigger pulses were replaced by digital data. The note-control and control parameters, sound data, pitch-bend and modulation controls were later standardized, and instruments could be easily interconnected.

Before MIDI, manufacturers were relatively free to use any method to provide interconnections between the instruments they produced, if at all. Commercial interests dictated that if a manufacturer used a different CV, gate and trigger pulse system, then purchasers would only be able to easily interconnect to other products within the manufacturers range. As a result, with a few exceptions, any interfacing between synthesizers from different manufacturers would require the conversion of voltages or currents. In addition, the performance controls were not fixed. Some manufacturers provided pitch-bend controls and multiple modulation controls, whilst others only had switched modulation: on or off. If an instrument was programmable, then the sound data was normally stored on data cassettes – again in proprietary formats.

MIDI was intended to enable the interchange and control of musical events with and by electronic musical instruments. It replaced the analogue voltages, currents and pulses with digital numbers, and so provided a simple way to assemble simple instruments into a larger unit. The layering of one sound with another changed from requiring two tracks on a multi-track tape recorder, to being a simple case of connecting two instruments together with a MIDI cable.

The introduction of MIDI had a profound and lasting effect on synthesizer design. Because the MIDI specification included a standard set of performance controllers, it effectively froze the pitch-bend and modulation wheel permanently into the specification of a synthesizer. MIDI is also biased towards a keyboard-oriented way of providing control: monophonic pressure is one example of this. MIDI also provided a standardized way of saving sound data by using system exclusive messages, and the possibility of editing front panel controls remotely.

The uniformity of many aspects of synthesizer design post-MIDI has meant that the emphasis has been placed onto the method of sound generation, rather than the functional design of the instrument. Although this has provided a wide variety of sounds, it has also meant that alternative controllers for synthesizers have tended to be largely ignored: the guitar synthesizer being one example.

3.7.7 Before and after microprocessors

The adoption of MIDI was also accompanied by a consolidation in the use of microprocessors. Microprocessors had begun to be used in polyphonic synthesizers to provide memory functions for storing sounds, but MIDI made the use of a microprocessor almost obligatory.

Before microprocessors, analogue synthesizers did not typically have autotune facilities or memories for sounds. Interfacing was through analogue voltages and the complexity meant that only two or three instruments would be connected together. Front panel controls actually produced the CVs that controlled the synthesizer sound circuitry.

Once microprocessors were incorporated in synthesizer designs, then autotuning was introduced for polyphonic synthesizers. Memories for sounds, and storage on floppy disk, data cassette or through MIDI system exclusive messages were possible. MIDI cables could be used to connect many instruments together. Front panel controls were scanned by the microprocessor to determine their position and thus produce a CV, or the front panel controls were replaced by a parameter system using buttons and a single control to select a parameter and edit it.

The changes in synthesizer design post-MIDI and post-microprocessors are most evident in rack-mounting synthesizer modules, which have very little in common with the exterior appearance of analogue synthesizers of the late 1970s: no keyboard, few or no control knobs, no data cassette, no CV sockets and no performance controls – MIDI is totally essential to their production and control of sounds.

Once the idea of having sound generation separate from the keyboard and performance controls had become established, then moving the synthesizer module from the rack to inside the computer itself was readily accepted.

Environment

For brevity, this section will use the phrase 'analogue synthesizers' to mean analogue synthesizers of the monophonic, polyphonic and modular varieties, as well as string synthesizers, electronic pianos, bass pedal synthesizers and other analogue electronic musical instruments.

3.8 Sampling in an analogue environment

3.8.1 Tape-based

Audio recording and playback (in this context: 'sampling') based on tape recording techniques has a long history. The first 'tape' recorders did not use tape at all, but used wire instead. Plastic tape covered with a thin layer of iron oxide is much easier and safer to handle than reels of wire, and far easier to cut and splice!

Tape recording

The underlying idea behind how a tape recorder works is very simple. The sound signal is converted into an electrical signal in a microphone, and this signal is then amplified, converted into a changing magnetic field and stored onto tape. By passing this magnetized tape past a replay head, the changes in the magnetic field are picked up, amplified and converted back into sound again.

Magnetic tape is made up from two parts:

1. A plastic material which is chosen for its strength, wear resistance and temperature stability.
2. Magnetic coating which is chosen for its magnetic properties.

It is actually possible to record and replay sounds using a fine layer of iron oxide dust placed onto the sticky side of an adhesive tape, although this is not recommended as a practical demonstration. The commercial versions of recording tape are just more sophisticated versions of this 'rust on tape' idea.

A tape recorder is a mixture of mechanical and electronic engineering. The mechanical system has to handle long lengths of fragile tape, pulling it across the record and replay heads at a constant speed, and ensuring that the tape is then wound onto the spool neatly. This requires a complex mixture of motors, clutches and brakes to achieve. The pulling of the tape across the heads is achieved by pressing the tape against a small rotating rod called the capstan. The tape is held onto the capstan with a rubber wheel called the pinch roller. The spool that is supplying the tape is arranged so that it provides enough friction to provide sufficient tension in the tape to press it against the record and replay heads as it is pulled past. Once past the capstan and roller, the tape is then wound onto the other tape spool. When the tape is wound forwards or backwards, the pinch roller is moved away so that the tape no longer presses against the capstan or the heads, and the spools can then be moved at speed (Figure 3.8.1).

The electronic part of a tape recorder has two sections: record and replay. The record part amplifies the incoming audio signal and then drives the record head with the amplified signal plus a high-frequency 'bias' signal. The combination of the two signals allows the response of the magnetic tape to be 'linearized'. Without the bias, the tape recorder would produce large amounts of distortion. The replay section merely amplifies the signal from the replay head (no bias is required for replay).

Mellotrons

The word 'Mellotron' is a trade-marked name for one type of sample-replay musical instrument which uses short lengths of magnetic tape. The concept is simple, the practicalities are rather more involved. The basic idea is to have a tape replayer for each key on the keyboard. A long capstan stretches across the whole of the keyboard. Pressing a key pushes the tape down onto the capstan and pulls it across the replay head. The tape is held in a bin with a spring and pulley arrangement to pull it back when the key is released. The length of the tape is thus fixed and so the key can only be held down for a limited time. Loops of tape cannot be used because the start of the sound would not be synchronized with the pressing of the key; that is, by arranging for the tape to be pulled back into the bin each time the key is released, it automatically goes back to the start point of the sound, ready for the next press of the key. There have been several other variants on the same idea from other manufacturers, but the Mellotron is the best known (Figure 3.8.2).

Because the capstan is the same size for each key, the tape for each key needs to be recorded separately, with each tape producing just one note (although several tracks are available on each tape, with a different sound on each track). The tapes are thus multi-sampled at 1-note intervals. Recording user samples

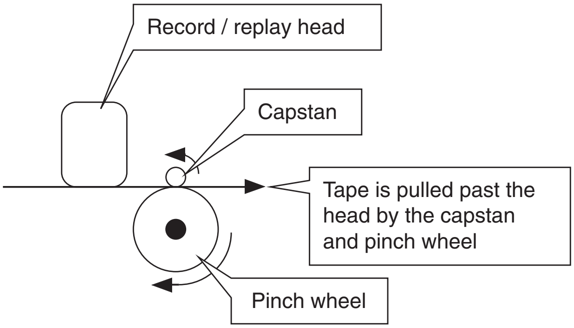


FIGURE 3.8.1 A tape recorder/player pulls the tape past the record/replay head. The capstan revolves at a constant rate and the tape is held against the capstan by the pinch wheel.

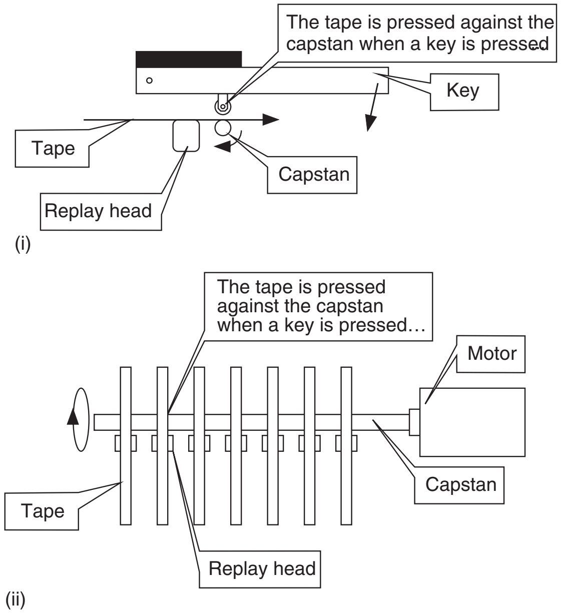

The tape is pressed against the capstan when a key is pressed
Key
Tape
Capstan
Replay head
(i)
The tape is pressed against the capstan when a key is pressed…
Motor
Tape
Capstan
Replay head
(ii)

FIGURE 3.8.2 (i) Side view and (ii) top view of a tape sample-replay instrument. The capstan spans the whole of the keyboard and revolves continuously. When a key is pressed down, this presses the tape against the capstan, which pulls the tape across the replay head.

for such a machine requires time, patience and attention to detail: the levels of the sounds must be consistent across all the tapes, for example. The 'frames' that contain complete key-sets of the tape bins can be changed, but this is not a quick operation. Because of the difficulty of recording your own sounds onto a tape, these tape samplers can almost be regarded as being sample-replay instruments rather than true samplers.

Tape loops

By looping a piece of tape around and joining the end to the beginning with splicing tape, it is possible to create a continuous loop of tape which will play the same piece or recorded material repeatedly. The only limitation on the size of the tape is physical: short loops may not fit around the tape recorder head and capstan, whilst long loops can be difficult to handle as they can easily become tangled.

The repetition of a sequence of sounds produces a characteristic rhythmic sound, which can be used as the basis of a composition. As with the Mellotron tape player, synchronizing the playback of the start of a loop is difficult, and synchronizing two loops requires them to be exactly the same length, or to have very accurate capstan motor speed control. Tape loops are thus usually used for asynchronous sound generation purposes.

The Watkins (WEM) CopyCat echo unit consists of a loop of tape and several replay heads, but the addition of the record head changes the function!

Pitch changes

Analogue tape recorders have one fundamental 'built-in' method of modifying the sound: speed control. Changing the speed at which the tape passes through the machine alters the pitch of the sound when it is played back. This can be either during the record or the replay process.

For example, if a sound is recorded using 15 inches per second (ips), and replayed at 7.5 ips, then it will be played back at half the speed, and thus will be shifted down in pitch by one octave. Conversely, sounds that are recorded at 7.5 ips and replayed at 15 ips will be played at twice the normal speed and will thus be shifted up in pitch by one octave. Note that the pitch and time are linked: as the pitch goes up, the time shortens, whilst lower pitch means longer time. The 'length' of a sound is exactly the length of the piece of tape on which it is recorded. If the tape is played back faster, then the tape passes over the replay head faster, and so the sound lasts for a shorter time. (The same is not necessarily true for digital samplers…)

This 'pitch halving and time doubling' was used to a great effect by guitarist Les Paul in the 1950s. Using the technique of recording low-pitched notes at a slow tape speed, and then replaying at a faster tape speed, he was able to achieve astonishingly fast and complex performances on guitar. The same technique is still a powerful way of changing the pitch of sounds, or for enabling virtuoso performances at slow tempos.

3.8.2 Analogue sampling

Analogue sampling covers any method which does not use tape or digital methods to store the audio signals.

'Bucket-brigade' delay lines

The most common technology which met these requirements in the 1970s was the 'bucket-brigade' delay line or analogue delay line. This used the charge on a series of capacitors to represent the audio signal, rather than the magnetic field used in tape systems or the numbers used in digital systems. The sampling process was merely the opening of an electronic switch to charge up the first capacitor in the delay line. The size of the voltage determined the amount of charge that was transferred to the capacitor: the higher the voltage which was being sampled, the more the charge which was stored in the capacitor. Effectively, the capacitor acted as a store for the voltage, since the presence of the charge in the capacitor was shown by the voltage across the capacitor. The switch then opened and the charge was held in the capacitor since there was no significant leakage path. Another switch was then used to transfer the charge to the next capacitor in the delay line, where it again produced a voltage. The original capacitor was then available to sample the next point on the incoming audio signal. This process continued, with the sample voltages moving along the delay line formed by the capacitors; hence the term 'bucket-brigade' delay lines (Figure 3.8.3).

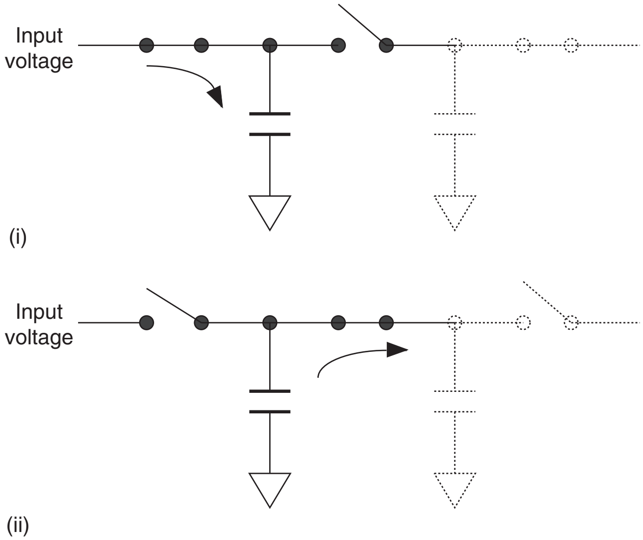


FIGURE 3.8.3 An analogue delay line moves charge along a series of capacitors connected by switches. (i) The input voltage is stored on the first capacitor. (ii) The charge is then transferred to the next capacitor. This repeats for the entire chain and so the input voltages move along the capacitors.

Because each section of the delay line is just a capacitor and some electronic switches, it was easy to fabricate, and so several thousands could be placed on a single IC chip. The sampling and transfer of charges required a relatively high-frequency clock signal, but the control circuitry was straightforward. This simplicity of control and application made analogue 'bucket-brigade' delay lines popular in the 1970s and early 1980s for producing echo, chorus and reverberation effects. At least one monophonic sampler was produced using analogue delay lines in the early 1980s, but it was rapidly superseded by digital versions.

The limitations of the analogue delay line technique are many fold: first, the capacitors and switches are not perfect, so some of the charge leaks away causing signal loss, distortion and noise; but more importantly, the high-frequency clock signals tend to become superimposed on the output audio signals and this degrades the usable dynamic range of the delay line. Also, because they sample the audio, the high-frequency sample clock needs to be low in order to achieve long time delays, but then the clock rate interferes with the audio signal. At high clock rates, the delay time is short. And so they acquired a reputation for poor high-frequency response, which was a direct result of designs that sampled at too low a frequency in order to try and maximize the delay time. Because of these problems, digital sampling technology has replaced analogue delay lines and modern equivalents can easily put an (analogue-to-digital converter ADC), (digital-to-analogue converter DAC) and storage onto a single chip.

As with many synthesizer-related analogue chips, some bucket-brigade delay line chips are now rare and can sometimes attract high prices when they are needed to repair old guitar flanger/chorus/echo units.

Delay lines

An alternative to bucket-brigade delay lines moving charge around is to use metal springs or metal plates to carry the sound signals acoustically/mechanically. Sounds are transferred to the metal using modified loudspeaker drivers, and the delayed sound signals are recovered with contact microphones. The physical size of these acoustic delay lines can be large, and the 'spring lines' and 'plate echoes' of the 1960s and 1970s have again been largely replaced by digital alternatives, including many emulations! Acoustic delay lines have the advantage that they are not a sampling system, but are more suited to reverberation effects than pure sampling – they are not suited to storing a sound and subsequently replaying it, instead they simply store a sound for short time.

Optical

One alternative sampling method uses a technique which is similar in principle to tape recording. Optical film soundtracks are a light-based variation of tape recording. Instead of storing the audio as a changing magnetic field, the film soundtrack uses the amount of light passing through the film to store the audio signal. This is normally achieved by arranging for large audio signal levels to allow a large amount of light to pass through the film, whilst small signals allow less light through. A photodetector and lamp are used to convert the transmission of light into an audio signal.

This modulation of light by an audio signal is normally achieved by using the audio waveform to control the width of a slot, and so the amount of light that passes through the film. Variable density (opacity) film can also be used, but this is rare for film use, although it has been used for experimental systems where film is used to produce sound by literally painting onto it to control the amount of light that passes through it at any given instant. By passing the resulting film through a lamp and photodetector, the optical version of the audio can be converted into sound. Although flexible, the complexity of producing the required degree of detail is enormous and very time consuming.

At least one manufacturer produced an optical sample-replay machine in the 1980s, but as with all analogue methods, this was not a success against the digital competitors.

3.9 Sequencing

Human musicians can be used for sequencing analogue synthesizers. Left-hand walking bass patterns are one example of a learned pattern that can move from a conscious control to an unconscious control. But sequencing in the context of analogue synthesizers is normally taken to refer to two different types of sequence:

1. Step sequencers
2. CV and gate.

1. Step sequencers

Step sequencers produce pattern loops that are normally 16 notes long, with 8-, 12-, 24- or 32-note variants in some circumstances. The sequences loop continuously once started, playing 16 notes in order, although sometimes they can be stopped with CVs or gates. The typical arrangement of controls is a row of rotary (or linear slider) controls with another row of LEDs above that 'scan' across. The controls are used for setting the pitch by setting the CV that is output when the associated LED is lit. Slider controls effectively give a 'pitch graph' or map of the notes being played. Sixteen step sequencers are often found on modular synthesizers, particularly for live performance (the scanning LEDs) and for some genres of electronic music (e.g. Tangerine Dream in the 1970s). Step sequencers are normally 1 volt/ octave, although there were exponential variants and converters between the two types.

The most useful musical feature is a quantiser circuit, which turns the continuous CV from the controls into discrete semitones. Without a quantiser, you should not use a step sequencer if you have perfect pitch. One feature of step sequencers is that they normally play a note for each step of the sequence: rests are unusual and usually are provided by adding a third row of switches to control the output of gate signals. If there are no gate controls, then one technique is to simulate rests by programming in very low notes.

When a modular synthesizer is being controlled by a step sequencer, it is common to patch in a keyboard and perhaps a sample/hold circuit so that notes played on the keyboard will transpose the step sequence. Without this addition, step sequencers can severely restrict the harmonic progression of the music.

2. CV and gate

CV and gate sequencing were features of some modular synthesizers (e.g. the large EMS systems and the EMS Poly-Synthi) and are more generic variant of the step sequencer, often using a computer to store the CVs, note durations and rest durations. One notable stand-alone example was Roland's MC-8 MicroComposer sequencer, which was introduce in 1977. This allowed the typing in of music as a series of numbers for pitch, note duration and rest duration. This exacting process, particularly for polyphonic music, could be very time consuming, and editing was primitive with a display that showed just the note time position, pitch, gate and CV details for one note at a time. Storage was on tape cassettes. Simpler stand-alone dedicated CV and gate sequencers followed, but difficulties with interfacing computers to CV- and gate-based analogue synthesizers meant that it was not until MIDI that general-purpose computers really started to play a role as sequencers. Once MIDI has become widely adopted, and computer-based MIDI sequencers were developed, then MIDI-to-CV/gate converters were used to enable analogue synthesizers to be controlled by a MIDI sequencer.

3.9.1 Wiring

It is worth considering the number of cables and converters that may be encountered in an analogue synthesizer sequencing environment. The synthesizers will probably have a power supply cable, plus one or more audio output cables. CV and gate cables might be augmented with additional CVs to affect filter cutoff or envelope decay/release time. Synchronization of a sequencer with a tape recorder, video playback, drum machine or other sequencers might require the use of standards like DIN-Sync 24, which was used before MIDI to provide synchronization with 24 pulse-per-quarter-note timing signals, plus a start/ stop signal, or MIDI Time Code or conversion between them. One volt/ octave and exponential CV systems might require conversion, and there were several different 'standards' for what constituted a gate signal, with corresponding converters.

3.10 Recording

Recording analogue synthesizers needs to take into account a number of challenges. First, because of all the cabling, it is very easy to get ground loops which can cause hum. Tuning stability can also be a problem, and so waiting for internal temperatures inside the synthesizers to stabilize after power-up, and then frequent tuning, may be required, even in a temperature-controlled environment. Most analogue synthesizers have mono outputs and so need to be panned or fed into two sets of comb filters to provide positional information in a mix, and they may sometimes require gating to prevent noise from escaping into a mix. In addition, the wide usage of low-pass filters in subtractive synthesizers can result in a mix becoming bass heavy, and a little high-pass filtering can help to remove this.

To produce polyphonic sounds from monophonic analogue synthesizers, you need either several synthesizers or to record the same one several times (tuning!). This can have unexpected side effects: slightly different rates of glissando, portamento or LFO modulation can sound very impressive.

Analogue synthesizers also have either limited effects (chorus in string synths) or none at all. Adding external effects to a synthesizer can produce a number of effects: echoes set to almost the clock rate of a step sequencer will produce syncopated rhythms that almost repeat an interesting contrast to the exact and predictable timing produced by digital synthesizers or computers with tempo-synchronized effects. Using just the pre-echoes and turning off the rest of the reverb, or vice versa, can be interesting too. Adding distortion to monosynths (polysynth chords tend to just produce noise) and playing guitar-influenced melody lines can produce a very distinctive sound.

3.11 Performing

To be played in context, synthesizers should be arranged in stacks, with a synthesizer on top of a string machine, on top of an organ or electric piano. Two-handed playing on different keyboards was much more common than split keyboards, except for the lower-cost multi-keyboards which mixed strings synths a VCF-based brass effect with a monophonic bass. Having two separate sounds and no restriction about which hand plays high or low parts (or both simultaneously) can be an interesting challenge, and one that can undo the legacy of piano lessons.

3.11.1 Memories

Memories were often very limited: the Yamaha CS-80 had four 'user' memories which were actually tiny control panels.

Early analogue synthesizers do not have memories for the sounds, and so the performer needs either to have multiple synthesizers or needs to change the sounds during performance. Given the cost of analogue synthesizers at the time, performers learned to change the controls to create different sounds. This required practice and a good familiarity with the synthesizer's layout and controls. Commonly changed parameters for these 'fast edits' include the VCO waveforms, VCO2 detune, VCF cut-off frequency and resonance, attack time and decay time. Because analogue synthesizers normally have live controls, parameters would often be changed during the performance, and so if any of the settings were not right, they would be changed with one hand whilst playing with the other. MIDI controller boxes and DJ controllers are the modern equivalent of this live parameter adjustment from the 1970s.

3.11.2 Sounds

Analogue synthesizers abound in clichéd sounds (some might say nothing but), although fashion and retro are cyclical, and if this is seen as bad, then waiting awhile may reverse the situation. Clichéd sounds can be used to advantage by avoiding the other clichés contextual sounds of the time: syndrum sweeps, spring-line reverbs, classic electronic drum sounds and 16-step sequencer bass lines (or by deliberately using all of these). Monosynth melody lines have some characteristic patterns of clusters of note playing followed by a held note being bent upwards or vibrato added (not unlike some guitar-solo clichés), and there are many examples on keyboard-oriented albums of the late 1970s and early 1980s that can be used as tutorials.

3.12 Example instruments

3.12.1 Moog modular (1965)

The Moog modular synthesizers comprise a number of modules which are placed in a frame which provides their power. Connections between modules are made using ¼-inch front panel jack connectors. Models were available where the number and choice of modules were pre-determined, or the user could make their own selection. The system shown here (Figure 3.12.1) provides enough facilities for a powerful monophonic instrument, although producing polyphonic sounds does require a large number of modules, and can be very awkward to control.

Note the logical arrangement of the panels: the controls are at the top and the sockets at the bottom. Although with two rows of modules, the patch-cords do tend to obscure the lower set of mostly VCO controls.

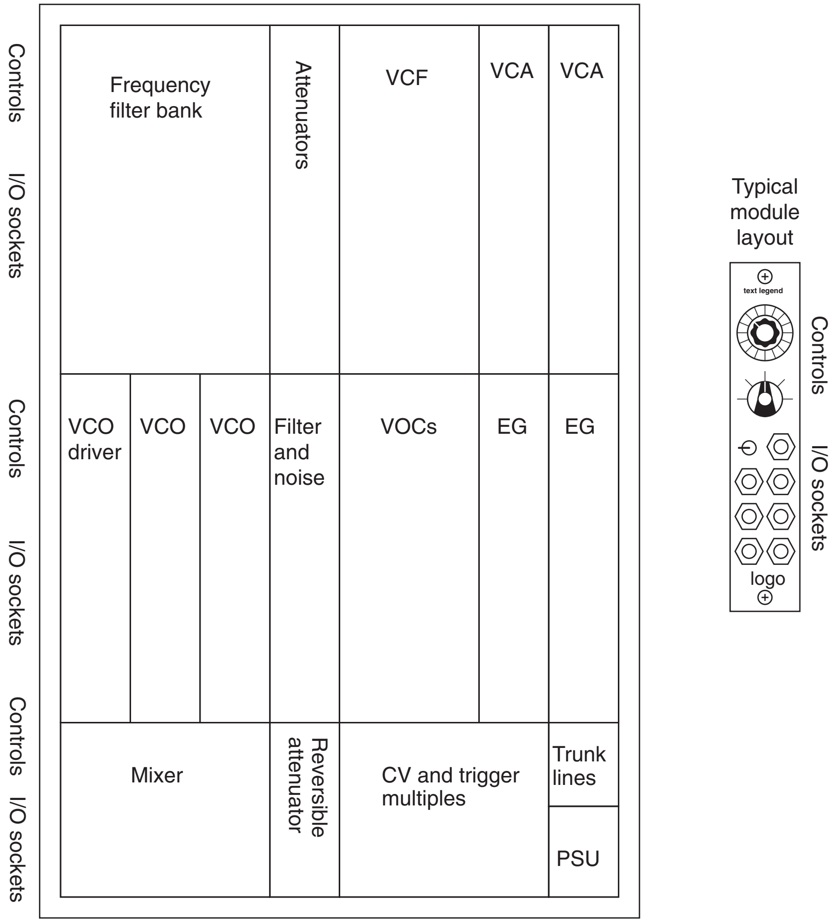


FIGURE 3.12.1
Moog modular.

3.12.2 Minimoog (1969)

The Minimoog was intended to provide a portable monophonic performance instrument (the Sonic Six repackaged similar electronics in a different case for educational purposes). It provides a hard-wired arrangement of synthesizer modules: VCOs, VCF, VCA, with two ADS EGs. This topology has since become the de facto 'basic' synthesizer 'voice' circuit, and can be found in many monophonic and polyphonic synthesizers, as well as custom 'synth-on-a-chip' ICs (Figure 3.12.2).

3.12.3 Yamaha CS-80 (1978)

The Yamaha CS-80 was an early polyphonic synthesizer made up from eight sets of cards, each comprising a dual VCO/VCF/VCA/ADSR type of synthesizer 'voice' circuit. Comprehensive performance controls made this a versatile and expressive instrument, if it was little bulky and heavy. Preset sounds were

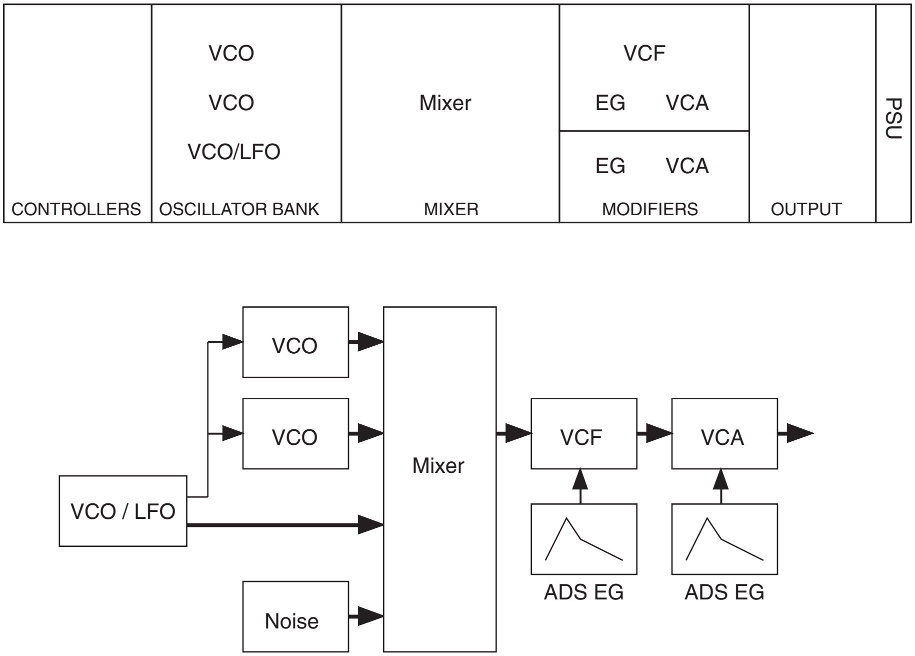


FIGURE 3.12.2
Minimoog.

provided and these could be layered in pairs. Four user memories were provided, these used miniature sliders and switches which echoed the arrangement of the front panel controls, which provided another two user memories. The presets could be altered only by changing the resistor values on a circuit board inside the instrument (Figure 3.12.3).

3.12.4 Sequential Prophet 5 (1979)

The Prophet 5 was essentially five 'Minimoog'-like synthesizer voice cards connected to a polyphonic keyboard controller. The major innovation was the provision of digital storage for sounds, although the ability to use one VCO to modulate the other, called 'poly-mod' by sequential, allowed the production of unusual FM sounds (Figure 3.12.4).

3.12.5 Roland SH-101(1982)

The SH-101 was intended for live performance and contained a simplified basic synthesizer 'voice' circuit. The instrument casing was designed so that it could be adapted for on-stage use by slinging it over the shoulder of the performer – a special hand grip add-on provided pitch-bend and modulation controls (Figure 3.12.5).

3.12.6 Oberheim Matrix-12 (1985)

The Matrix-12 (and the smaller Matrix-6) was a modular synthesizer in a case which was more typical of a performance synthesizer. The front panel extends the use of displays, which was pioneered in earlier OB-X models – this time

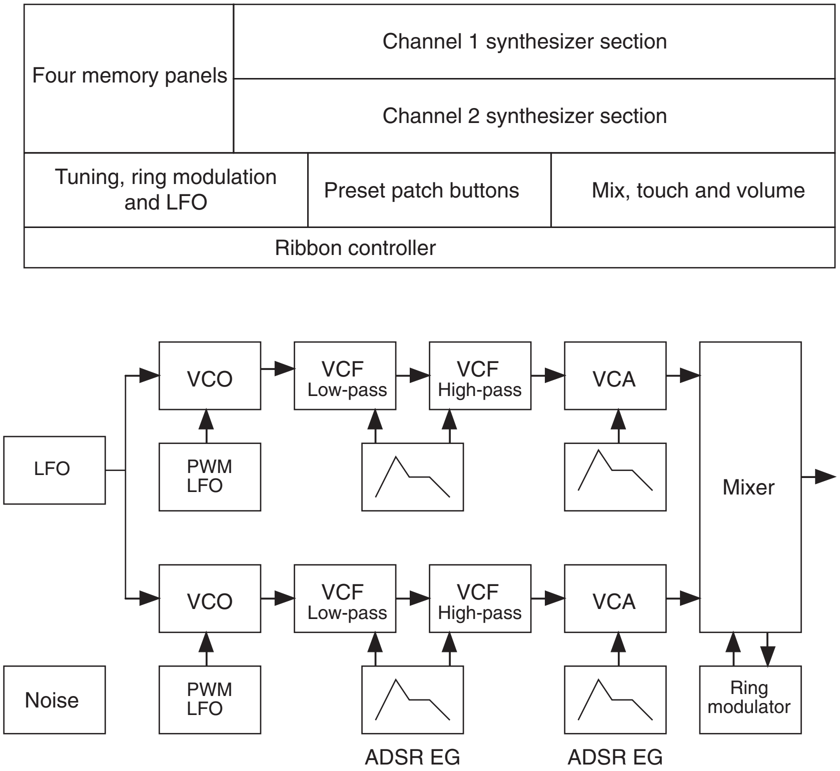


FIGURE 3.12.3
CS-80.

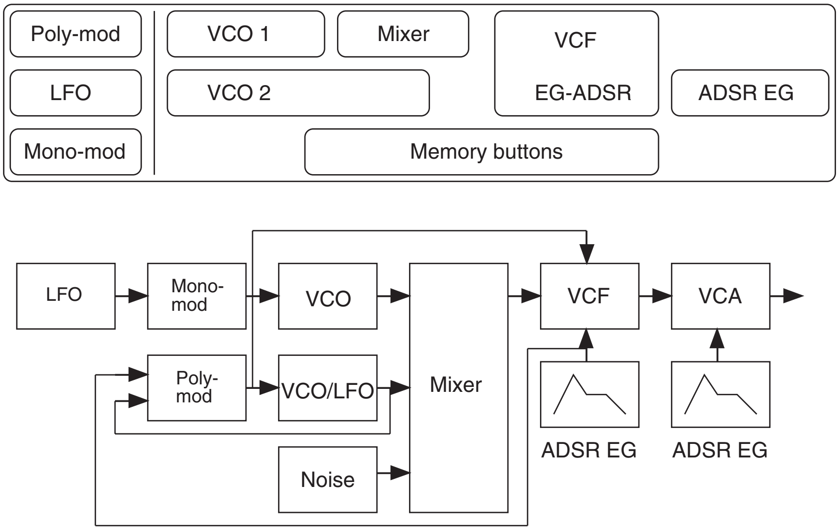


FIGURE 3.12.4
Prophet 5.

using green cold-cathode displays, to provide reassignable front panel controls. The wide range of processing modules made this a versatile and powerful instrument. Only 1 voice from the 12 available is shown in Figure 3.12.6.

3.13 Questions

1. Name three ways of producing sound electronically using analogue synthesis and briefly outline how they work.
2. Describe the 'source and modifier' model for sound synthesis.

FIGURE 3.12.5
SH-101.

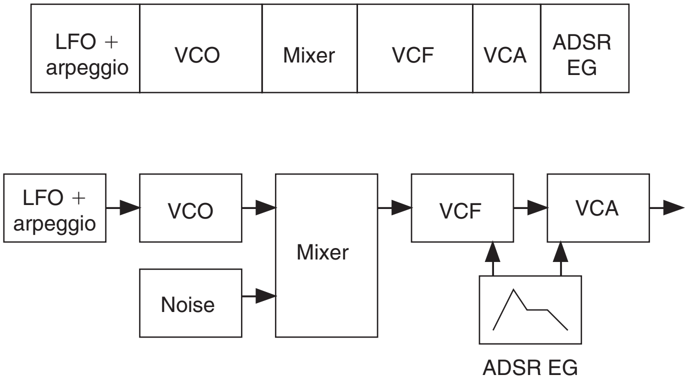
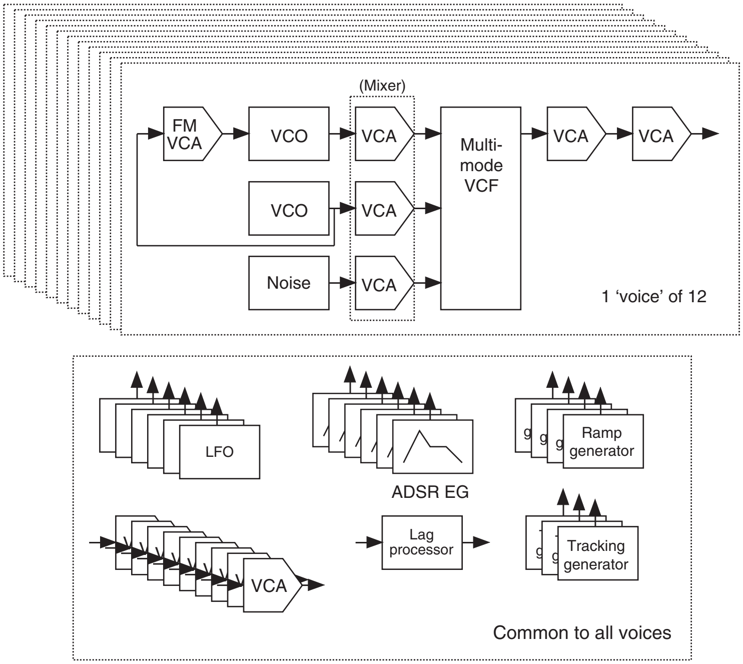


FIGURE 3.12.6
Matrix-12.



3. What are the basic analogue synthesizer source waveforms and their harmonic contents? To make the harmonic content increase as a waveform selector control is rotated, in what order should the waveforms be arranged?
4. What are the four major types of filter response curve? What effect do they have on an audio signal?
5. What are the main parts of an envelope? Include examples of the envelopes of real instruments.
6. How do vibrato and tremolo differ?
7. Why is it difficult to construct an analogue additive synthesizer?
8. What are the differences between AM, FM and ring modulation? Draw a spectrum for a 1-kHz carrier and 100-Hz modulator for each type of modulation.
9. Compare and contrast monophonic and polyphonic synthesizers.
10. Outline the effect of MIDI on synthesizer design. Suppose MIDI had been specified for guitar synthesizers instead of keyboards – how would it differ?

3.14 Timeline

Date	Name	Event	Notes
1500	Barrel Organ	The barrel organ. Pipe organ driven by barrel covered with metal spikes.	The forerunner of the synthesizer, sequencer and expander module!
1700	J. C. Denner	Invented the Clarinet.	Single reed woodwind instrument.
1700	Orchestrion	Orchestrions made in Germany. Complex combinations of barrel organs, reeds and percussion devices. Used for imitating orchestras.	
1804	Leonard Maelzel	Leonard Maelzel invented the Panharmonicon, another mechanical orchestral imitator.	
1807	Jean Baptiste Joseph Fourier	Fourier published details of his Theorem, which describes how any periodic waveform can be produced by using a series of sine waves.	The basis of additive synthesis.
1870	Gavioli	Fairground organs from Gavioli began to use real instruments to provide percussion sounds.	
1876	Alexander Graham Bell	Invented the telephone.	Start of the marriage between electronics and audio.
1877	Loudspeaker	Ernst Siemens patented the electrical loudspeaker.	Used for telephones.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1877	Thomas Edison	Thomas Alva Edison invented the cylinder audio recorder – the 'Phonograph'. Playing time was a couple of minutes!	Cylinder was brass with a thin foil surface – replaced with metal cylinder coated with wax for commercial release.
1878	David Hughes	Moving coil microphone invented.	
1887	Torakusu Yamaha	Torakusu Yamaha built his first organ.	
1888	Emile Berliner	First demonstration of a disk-based recording system – the 'Gramophone'.	Disk was made of zinc, and the groove was recorded by removing fat from the surface, and then acid etching the zinc.
1896–1906	Thaddeus Cahill	Invented the Telharmonium, which used electromagnetic principles to create tones.	Telephony.
1896	Thomas Edison	Motion picture invented.	
1903	Double-sided record	The Odeon label released the first double-sided record.	Two single-sided records stuck together?
1904–1915	Valve	Development of the Valve.	The first amplifying device – the beginnings of electronics.
1915	Lee de Forest	The first Valve-based oscillator.	
1920	Cinema organs	Cinema organs, used electrical connection between the console keyboard and the sound generation.	Also started to use real percussion and more: car horns, etc. – mainly to provide effects for silent movie accompaniment.
1920	Lev Theremin	The Theremin – patented in 1928 in the United States. Originally called the 'Aetherophone'.	Based on interfering radio waves.
1920	Microphone recordings	First major electrical recordings made using microphones.	Previously, many recordings were 'acoustic' – used large horns to capture the sound of the performers.
1928	Maurice Martenot and Ondes	Invented the Ondes Martenot – an early synthesizer.	Controlled by a ring on a wire – finger operated.
1930	Baldwin, Welte, Kimball and others	Opto-electric organ tone generators.	
1930	Bell Telephone Labs	Invented the Vocoder – a device for splitting sound into frequency bands for processing.	More musical uses than telephone uses!
1930	Friedrich Trautwein	Invented the Trautonium – an early electronic instrument.	Wire pressed onto metal rail. Original was monophonic. Later duophonic.

Timeline (*Continued*)

Date	Name	Event	Notes
1930	Ondes	Ondioline – an early synthesizer.	Used a relaxation oscillator as a sound source.
1930	Record groove direction	Some dictation machines recorded from the center out instead of edge in.	This pre-empts the CD 'center out' philosophy.
1930	Run-in grooves	Run-in grooves on records invented.	Previously, you put the needle into the 'silence' at the beginning of the track…
1933	Stelzhammer	Electrical instrument using electromagnets to produce a variety of timbres.	
1934	John Compton	UK patented for rotating loudspeaker.	
1934	Laurens Hammond	Hammond 'Tone Wheel' Organ used rotating iron gears and electromagnetic pickups.	Additive sine waves.
1935	AEG, Berlin	AEG in Germany used iron oxide backed plastic tapes produced by BASF to record and replay audio.	Previously, wire recorders had used wire instead of tape.
1937	Tape recorder	Magnetophon magnetic tape recorder developed in Germany.	The first true tape recorder.
1939	Hammond	Hammond Novachord – first fully electronic organ.	Used 'master oscillator plus divider' technology to produce notes.
1939	Hammond	Hammond Solovox – monophonic 'synthesizer'.	British Patent 541911, US Patent 209920.
1945	Metronome	First pocket metronome produced in Switzerland.	
1945	Ronald Leslie	Patents rotating speaker system.	
1947	Conn	Independent electromechanical generators used in organ.	
1948	Baldwin	Blocking divider system used in organ.	
1949	Allen	Organs using independent oscillators.	
1950	John Leslie	Reintroduction of Leslie speakers.	This time they were a success.
1951	Hammond	Melochord.	
1954	Milton Babbitt, H. F. Olsen and H. Belar	RCA Music Synthesizer mark I.	Only monophonic.
1957	RCA	RCA Music Synthesizer mark II.	Used punched paper tape to provide automation.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1958	Charlie Watkins	Charlie Watkins produced the CopyCat tape echo device.	
1958	RCA	RCA announced the first 'cassette' tape – a reel of tape in an enclosure.	Not a success.
1959	Yamaha	First 'Electone' organ.	
1960	Clavioline	Clavioline	British Patent 653340 and 643846.
1960	Mellotron	The Mellotron, which used tape to reproduce real sounds.	Tape-based sample playback machine.
1960	Wurlitzer, Korg	Mechanical rhythm units built into home organs by Wurlitzer and Korg.	
1963	Don Buchla	Simple VCO-, VCF- and VCA-based modular synthesizer: 'The Black Box'.	Not well publicized.
1963	Herb Deutsch	First meeting with Robert Moog. Initial discussions about voltage-controlled synthesizers.	
1963	Philips	Philips in Holland announced the 'Compact Cassette' – two reels plus tape in a single case.	A success well beyond the original expectations!
1964	Philips	The Compact Cassette was launched.	Tape made easy by hiding the reels away.
1965	Paul Ketoff	Built the 'Synket', a live performance analogue synthesizer for composer John Eaton.	Commercial examples like the Minimoog and Arp Odyssey, soon followed.
1965	Robert Moog	First Moog Synthesizer was hand-built.	Only limited interest at first.
1966	Don Buchla	Launched the Buchla Modular Electronic Music System – a solid-state, modular, analogue synthesizer.	Result of collaboration with Morton Subotnick and Ramon Sender.
1966	Rhythm machine	Rhythm machines appeared on electronic organs.	Non-programmable, and very simple rhythms.
1968	Ikutaro Kakehashi	First stand-alone drum machine, the 'Rhythm Ace FR-1'.	Designed by the future boss of Roland.
1968	Walter Carlos	'Switched On Bach', an album of 'electronic realizations' of classical music, became a best seller.	Moog synthesizers suddenly changed from obscurity to stardom.
1969	Peter Zinovief?	EMS produced the VCS-3, the UK's first affordable synthesizer.	The unmodified VCS-3 was notable for its tuning instability.

Timeline (*Continued*)

Date	Name	Event	Notes
1969	Robert Moog	Minimoog was launched. Simple, compact monophonic synthesizer intended for live performance use.	Hugely successful, although the learning curve was very steep for many musicians.
1970	ARP Instruments	ARP 2600 'Blue Meanie' modular-in-a-box released.	
1970	ARP Instruments, Alan Richard Pearlman	ARP 2500. Very large modular studio synthesizer.	Uses slider switches – a good idea, but suffered from crosstalk problems.
1971	ARP Instruments	The 2600, a performance-oriented modular monosynth in a distinctive wedge-shaped portable case.	
1972	Roland	Ikutaro Kakehashi found Roland in Japan, designed for R&D into electronic musical instruments.	First products are drum machines.
1972	Roland	TR-33, 55 and 77 preset drum machines launched.	
1978	Electronic Dream Plant	Wasp Synthesizer launched. Monophonic, all-plastic casing, very low cost, touch keyboard – but it sounded much more expensive.	Designed by Chris Huggett and Adrian Wagner.
1978	Roland	Roland launched the CR-78, the world's first programmable rhythm machine.	
1978	Sequential Circuits	Sequential Circuits Prophet 5 synthesizer – essentially five Minimoog-type synthesizers in a box.	A runaway best seller.
1978	Yamaha	Yamaha CS series of synthesizers (50, 60 and 80), the first mass-produced successful polyphonic synthesizers.	Korg, Oberheim and others also produced polyphonic synthesizers at about the same time.
1979	Roland	Boss 'Dr. Rhythm' programmable drum machine.	
1979	Roland	Roland Space–Echo launched – used long tape loop and had built-in spring-line reverb and chorus.	A classic device, used as the basis of several specialist guitar performance techniques (e.g. Robert Fripp).
1979	Roland	VP-330, the 'Vocoder Plus': a string/vocal chorus machine with a built-in vocoder.	Roland and Korg have both released twenty-first century mixes of synth plus vocoder…
1979	TASCAM	Introduced the 'Portastudio', a 4-track recorder and mixer for compact cassette.	Made 4-track recording at home affordable and convenient.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1980	Roland	Jupiter-8 polyphonic synthesizer.	8-note polyphonic, programmable poly-synth.
1980	Roland	Roland TR-808 launched. Classic analogue drum machine.	
1980	Roland	Jupiter-8 polyphonic synthesizer.	8-note polyphonic, programmable poly-synth.
1981	Moog	Robert Moog was presented with the last Minimoog at NAMM in Chicago.	The end of an era.
1981	Roland	Roland Jupiter-8. Analogue 8-note polyphonic synthesizer.	
1982	Moog	Memory Moog – six note polyphonic synthesizer with 100 user memories.	Cassette storage! Six Minimoogs in a box!
1982	Roland	Jupiter-6 launched – first Japanese MIDI synthesizer.	Very limited MIDI specification. 6-note polyphonic analogue synth.
1982	Sequential	Prophet 600 launched – first US MIDI synthesizer.	6-note polyphonic analogue synth – marred by a membrane numeric keypad.
1983	Oxford Synthesiser Company	Chris Huggett launched the Oscar, a sophisticated programmable monophonic synthesizer.	One of the few monosynths to have MIDI as standard.
1984	Sequential	Sequential launched the Max, an early attempt at mixing home computers and synthesizers.	A complete failure – too early for the market.
1984	Sequential Circuits	SixTrak. A multi-timbral synthesizer with a simple sequencer.	The first 'workstation'?
2001	Alesis	A6 Andromeda, 16-voice analogue synth.	Digitally controller oscillators, but analogue filters and lots of modulation facilities.
2005	Bob Moog	Bob Moog, synthesizer pioneer, died.	1934–2005 (pronounced to rhyme with 'vogue').

CHAPTER 4

Making Sounds with Hybrid Electronics

CONTENTS

Hybrid Synthesis

Environment

Hybrid synthesis is the name usually associated with methods of synthesis that are not completely analogue or digital. These borderline methods were most important during the changeover from analogue to digital sound generation in the early 1980s, but the underlying techniques have also become part of the all-digital synthesis methods. With the continuing increase of interest in 'analogue' synthesis that began in the 1990s, it is intriguing to note that very few of the instruments that are now being designed are truly analogue; in many ways they are actually hybrids, even if this is just for programmability.

Synthesis methods that combine more than one techniques or methods of synthesis to produce a composite sound are described as 'layered' or 'stacked' and are covered in Chapter 6. Although these methods are sometimes called 'hybrid' methods, the term 'composite' synthesis is preferred by the author.

It is possible to divide hybrid synthesizers into different classes. One possible division is based on the roles of the digital and analogue parts:

- Digital control of the parameters of analogue synthesis, as used in many programmable analogue monophonic synthesizers.
- Digital control of the oscillator (in other words, digitally controlled oscillator, DCO) with the remainder of the instrument analogue, perhaps with digital control of the parameters.
- Digital oscillator with analogue modifiers and with digital control of the analogue parameters. These are the forms that many of the mid-1990s 'retro' analogue synthesizers used, and examples are still being produced in the twenty-first century.

Another classification might be made on the method used to produce the sound. This section divides hybrid synthesizers using this method. Wavecycle, wavetable and DCO technologies are all discussed.

The predominant method of hybrid synthesis uses digital sound generation and control of parameters with analogue filtering and enveloping.

This uses the technology that is the most appropriate for the task. Hybrid synthesizers are often characterized by a more sophisticated raw sound as the output from the 'source' part, which is due to the use of digital technology, especially in wavetable-based synthesizers. In fact, this availability of additional waveshapes, beyond the 'traditional' analogue set of sine, triangle, sawtooth and rectangular waveforms, could be seen as the major differentiator between analogue and hybrid instruments. Although the idea of mixing analogue with digital was pioneered in the late 1970s, most notably with the Wasp synthesizer, it still forms the basis of many of the most successful hybrid (and hybrid masquerading as analogue) instruments. In fact, the recent trend of using digital circuitry and software to replace traditional analogue functions like filters or oscillators follows on from hybrid synthesis. In addition, the hybrid design philosophy of using a complicated oscillator and conventional 'subtractive' modifiers also forms the basis of all sample and synthesis (S&S) instruments.

4.1 Wavecycle

A wavecycle is another term for waveform, although it emphasizes the term cycle, which is very significant in this context. It is used here to emphasize the difference between the 'static, sample-based' replay-oriented wavecycle oscillators and the 'dynamic, loop-based' wavetable oscillators.

Analogue synthesizers incorporate voltage-controlled oscillators (VCOs) or oscillators that can typically produce a small number of different waveforms with fixed waveshapes, where each cycle is identical to those before and after it. The one exception to this is a pulse width modulation (PWM) waveform, where the shape of the pulse can be changed using a control voltage, often from a low-frequency oscillator (LFO) for cyclic changes of timbre.

Hybrid synthesizers that use wavecycle-based sound generation can use this single-cycle mode, but they can also produce additional waveshapes, and use more complex schemes where more than one cycle of the waveform is used before the shape is repeated. The logical conclusion to this is for there to be a large number of cycles, each different, and with no repetition at all; the result is then called a sample. The only really important differences between these examples are the length of the audio sample and the amount of repetition. Each method has its own strengths and weaknesses.

4.1.1 Single-cycle

Single-cycle oscillators produce fixed waveforms, somewhat like an analogue synthesizer, although the selection of waveshapes is often much larger. The method of producing the waveform is often a mixture of analogue and digital circuitry; digital technology has fallen in price, the use of digital circuit design has increased predominantly.

Possibly, the simplest method of controlling a waveshape is the pulse width control, which is sometimes found in analogue synthesizers. With a single control, a variety of timbres can be produced: from the 'hollow'-sounding square wave with the missing second harmonic through to narrow pulses with a rich harmonic content and a thin, 'reedy' sound.

Pulse waveforms usually have only two levels, although there are variants that have three, where the pulses are positive and negative with respect to a central zero value. Multiple levels were used in one of the first 'user-programmable' waveforms, called 'slider scanning'. In this method, the oscillator runs at several times the required frequency and is used to drive a counter circuit, which then controls an electronic switch called a multiplexer. The multiplexer 'scans' across several slider controls and thus creates a single waveform cycle where the voltage output for each of the stages is equivalent to the positions of the relevant slider. By setting half of the sliders to the maximum voltage position, and the remainder to the minimum, a square waveform is produced, but a large number of other waveforms are also possible (Figure 4.1.1).

Slider scanning oscillators are limited by the number of sliders that they provide. Eight or sixteen sliders are often used, and this means that the oscillator is running at 8 or 16 times the frequency of a VCO producing the same note conventionally. The counter is normally arranged so that it switches in each slider in turn, and when all of the sliders have been scanned, it returns to the first slider again. The sliders thus represent one cycle of the waveform. This type of counter is called a Johnson counter; although it is possible to use counters that scan back and forth along the sliders, the relationship between the sliders and the cycle of the waveform is less obvious. Although it appears to be only half of a cycle, the reversal of the scan direction merely adds in a time-reversed version of the sliders, and this sounds like a second cycle of a

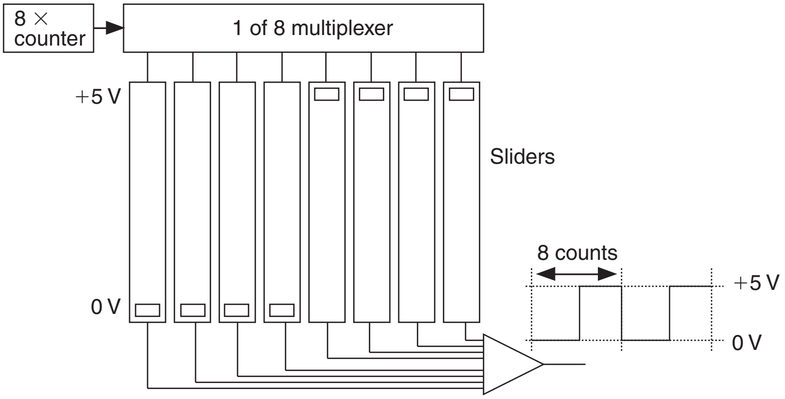


FIGURE 4.1.1 This eight slider scanner circuit runs a counter at 8 times the required frequency. The counter causes a '1 of 8' multiplexer to sequentially activate each of the 8 slider controls, which produce a voltage dependent on their position. The slider outputs are summed together to produce the output waveform.

two-cycle waveform. The pitch is thus unaltered. Slider scanners thus provide single-cycle (or two-cycle) waveforms where the shape of the waveform is static (unless the slider positions are changed) and repeated continuously.

For detailed control over the waveform, the obvious solution is to add more sliders. Providing many more than 16 separate sliders quickly becomes very cumbersome, and it makes rapid selection of different waveforms almost impossible. One alternative is to provide pre-stored values for the slider positions in a memory chip (read-only memory (ROM) or random-access memory (RAM)) and use these values to produce the output waveform. In this way, many more 'sliders' can be used, and the waveform can be formed from a large number of separate values, instead of just 8 or 16. The difficulty lies in producing the values to put in the memory – preset values can be provided – but sliders or other means of user control of the values are preferable. A minimalist approach might be to provide two displays: one for the 'slider' number and another for the value at that slider position.

The Fairlight Computer Musical Instrument (CMI) allowed users to draw waveforms on a screen using a light-pen. Unfortunately, the waveform is not a good guide to the timbre that will be produced, see Figures 3.4.2 and 3.4.3.

Drawing the values on a computer display screen has been used as one method of providing a more sophisticated user interface to large numbers of sliders, but this can be very tedious to use and difficult to achieve the desired results. Trying to set the positions of several hundred sliders on a screen to produce a particular timbre is also hampered by the relationship between the shape drawn and the timbre produced, which is not intuitive for most users, and requires considerable practice and experience before a specific timbre can be quickly set up. The simpler oscillators that scan through values in a memory chip are very economical in their usage of memory, and a large number of waveforms can be made available in this way. As with analogue synthesizers, selecting a waveform is easier if they are arranged in some sort of order – either a gradually increasing harmonic content or else in groups with variations of specific timbres: pulses, multiple sine waves added together, and soon.

Scanning across a series of voltages, which are set by slider positions, using a multiplexer is straightforward. Replacing the slider positions with numbers then requires some way of converting from a number to a voltage. This is achieved using a circuit called a 'digital-to-analogue converter' (DAC), which converts a digital number into a voltage. By sequentially presenting a series of numbers at the input to the DAC, a corresponding set of output voltages will be produced (Figure 4.1.2).

By storing the numbers in ROM, a large number of preset waveforms can be provided, merely by sending different sets of numbers from the ROM to the DAC. This is easily accomplished by having additional control signals that set the area of the memory that is being used. One simple way to achieve this is to use the low-order bits from the counter to cycle through the memory, whilst the high-order bits can be used to access different cycles, and thus, the different stored waveforms. For user-definable waveforms, RAM chips are used, where values can also be stored in the chip instead of merely recalled. Often, a mixture of ROM and RAM is used to provide fixed and user-programmable

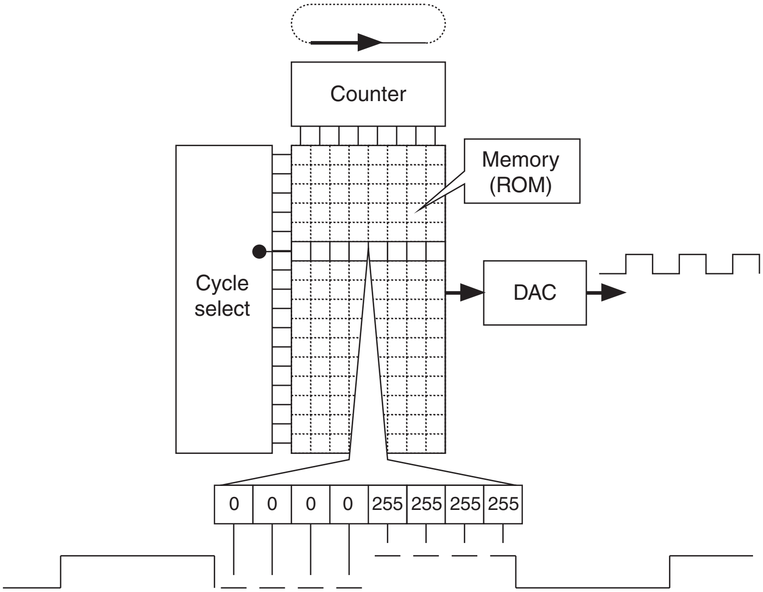


FIGURE 4.1.2 When a wavecycle is stored in memory, then a counter can be used to successively read each of the values and output them to the DAC, and so produce the desired waveform. This repeats for each cycle of the waveform. The location in the memory which is selected by the 'cycle select' logic determines the cycle shape. The 8-bit values are shown here only for brevity: 16-bit representations became widely adopted in the late 1980s.

waveforms, but there are alternatives to using memory chips. By using the output of a counter as the input to the DAC, a number of different waveforms can be produced: it depends on the way that the counter operates. If assumed that the DAC converts a simple binary number representation, then a simple binary counter would produce a sawtooth-like staircase waveform. There are a large number of types of counter that could be used for this purpose, although dynamically changing the type of counter is not straightforward. Some other types that might be used include the up–down and Johnson counters mentioned earlier and the Gray-code counters.

By using digital feedback between the stages of a counter, it is possible to produce a counter that does not just produce a short sequence of numbers in sequence, but a very much longer sequence of numbers in a fixed but relatively unpredictable order. These are called pseudo-random sequence generators, and they can be used to produce noise-like waveforms from a DAC. Actually, this is a multi-cycle waveform (see later) rather than a single-cycle. But by deliberately using the wrong feedback paths (or by resetting the count), it is possible to shorten the length of the sequence so that it produces sounds with a definite pitch, where the length of the sequence is related to the basic pitch that is produced. In effect, the length of the sequence becomes the length of one cycle of the waveform. Slight changes in the feedback paths or the initial conditions of

Whilst ingenious synthesis techniques sometimes find their way into commercial instruments, the long-term trend has always been for straightforward metaphors and user interfaces, especially when referenced to 'real-world' circuits. Digitally modeled analogue synthesis is one example of how strong this bias is.

the counter can produce a wide variety of waveforms from a relatively small amount of circuitry with simple (if unintuitive) controls, and no memory is required (unless the paths and initial conditions need to be stored, of course!).

Since digital circuitry is concerned with only two values, on and off or one and zero, the square or pulse waveform is a basic digital waveform, in much the same way as sine waves are the underlying basis of analogue. By taking several square or pulse waveforms at different rates, and adding them together, they can be used to produce waveshapes in much the same way as adding several sine waves together (see Chapter 4). These are called Walsh functions, and although conceptually very different from the systems that read out values from a memory chip, the simplest method of producing them is to calculate the values and store them in memory. Providing a user interface to a Walsh function driven waveform generator would require comprehensive control over many pulse waves, and as with drawing a waveform on a screen, it suffers from the same problems of complexity and detail, without any intuitive method of determining the settings (Figure 4.1.3).

Filtering of outputs

All of the methods of producing arbitrary waveshapes using digital circuitry described earlier produce outputs that tend to have flat segments connected by sudden transitions. Since these rapid changes produce additional (often unwanted) harmonics, the outputs need to be filtered so that the final output is 'smooth', usually with a low-pass filter whose cut-off frequency is set to be at the highest required frequency in the output. Because the frequency of the output waveform from an oscillator can change, the filter needs to track the changes in frequency, which means that the VCO needs to be coupled to a voltage-controlled filter (VCF), and set up so that the cut-off frequency follows the oscillator frequency, usually by connecting the same control voltage to the VCO and VCF. In some circumstances, the additional frequencies are deliberately allowed to pass through the filter. Because these frequencies are linked to

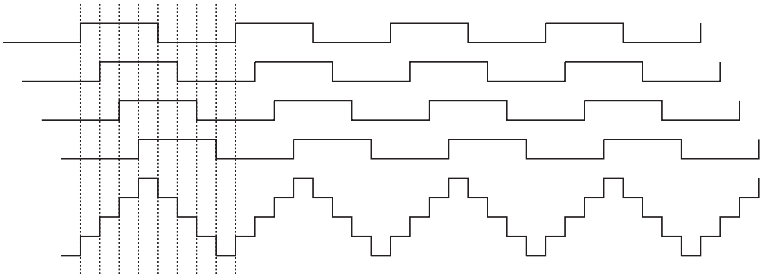


FIGURE 4.1.3 Walsh functions combine square or pulse waveforms to produce more complex waveforms. In this example, four square waves are added together to produce a crude 'triangle' type of waveform. Each position in the output is produced by adding together the level of each of the component waveforms.

the oscillator frequency, they are actually harmonics of it, albeit high harmonics. Removing the filter means that a waveform, which might appear to be a sine wave from the slider positions, is actually a sine wave with extra harmonics. The ability to switch the filter in and out, together with knowledge of how the waveforms are being produced, is very useful if the most is to be made of the potential of single-cycle oscillators (Figure 4.1.4).

Waveshapes

In summary, single-cycle oscillators normally have a selection of waveshapes based on the following types:

- Mathematical shapes: sine, triangle, square and sawtooth.
- Additions of sine wave harmonics (organ 'drawbar' emulations).
- Additions of square or pulse waves (Walsh functions).
- Random: single-cycle 'noise' waveforms from pseudo-random sequence generators tend to be very non-white in character and often have large amounts of high harmonics.

Single-cycle oscillators normally have a characteristic fixed timbre – each cycle is the same as the previous one and the next one. This means that subsequent processing through modifiers is often used to make the sound more interesting to the ear.

One alternative modifier possibility is to make the waveform vary with the velocity of the note played. This is known as 'velocity switching'. Therefore a note played hard, with a high-velocity value, would produce a bright sound, whilst a note played more softly, with a lower velocity value, would produce a duller sound. The simple case with two levels gives an abrupt change of timbre, but with more levels, more gentle and subtle variations (and the opposite) are possible. This waveform modification might give the same end effect as a filter controlled by velocity, but it also allows other effects that are more complex. For example, the change in timbre might not be just a simple change in brightness and could be changes in just a few harmonics, leaving the remainder unchanged. More complex variations might change the timbre several times with velocity, in ways that are not possible with simple filtering. This velocity switching moves some aspects of modification into the sound source and can produce very dynamic sounds from modest synthesis capability.

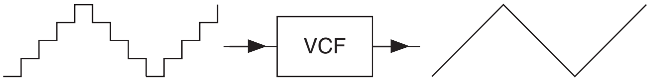


FIGURE 4.1.4 Filtering the output of a generated wavecycle waveform can smooth out the abrupt transitions and produce the required shape.

4.1.2 Multi-cycle

The important thing about multi-cycle oscillators is that although they can have many cycles of waveforms that they output in sequence, the same set of cycles repeats continuously. This is different to a sample, where the sample may only be played through in its entirety once. The technology for producing multi-cycle waveforms is very similar to single-cycle oscillators, although the user interface is often restricted to merely choosing the specific waveforms for each cycle, rather than providing large numbers of slider or other controls. The basic method uses a memory chip and DAC, just as with single-cycle oscillators. The difference is that the area of memory that is being cycled can also be controlled dynamically. A simple example might have two areas of ROM – one containing a square waveform and the other a sawtooth waveform (Figure 4.1.5). By setting the ROM to output first the square, then the sawtooth, and then repeating the process, the output will be a series of interspersed square and sawtooth cycles. This two-cycle waveshape has a harmonic structure that incorporates some elements from each of the two types of waveforms, but also has additional lower frequency harmonics that are related to half of the basic cycle frequency. This is because the complete cycle repeats at half the fundamental cycle rate.

By concatenating more waveforms together, more complex sequences of waveforms can be produced. As the length of the sequence increases, the extra low-frequency harmonics also drop in frequency.

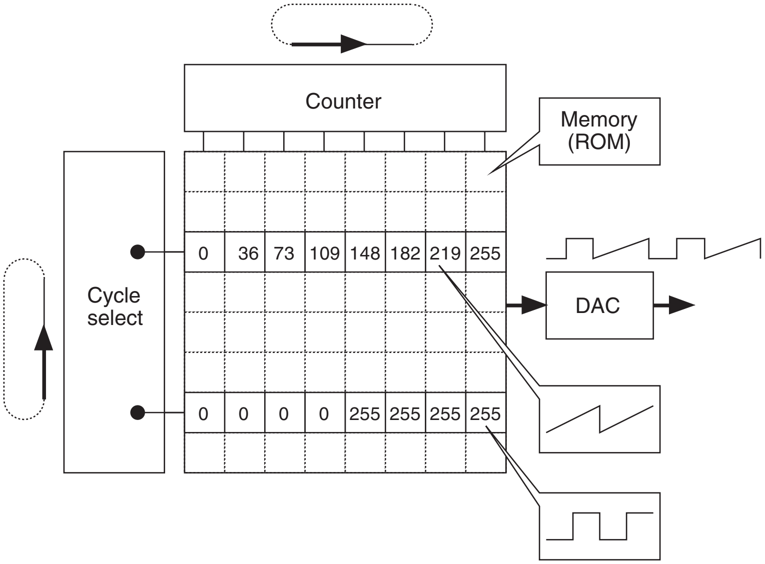


FIGURE 4.1.5 By addressing two (or more) different parts of the waveform ROM, the output waveform can be changed on a 'per cycle' basis. Here, 8-bit representations of a square and sawtooth waveforms are present in the ROM, and are sequenced cyclically to form a composite output waveform.

To take an extreme example, imagine one cycle of a square waveform followed by three cycles of a silence waveform. The equivalent is a pulse waveform with a frequency of a quarter of the square wave cycle – two octaves below the pitch that was intended. With eight cycles in the sequence, then the lowest harmonic component will be three octaves down, and with 16 cycles, the frequency will be four octaves down. If the length of the sequence is not a square of two, then the frequency that is produced may not be related to the cycle frequency with intervals of octaves.

For example, if a square wave cycle is followed by two cycles of silence, then the effective frequency of the pulse waveform is a third of the basic cycle frequency, which means that the lowest frequency will be an octave and a fifth down, and the ear will interpret this as the fundamental frequency, and therefore the oscillator has apparently been pitch shifted by an octave and a fifth. With longer sequences of single-cycle waveforms, this pitch change can be harmonically unrelated to the basic pitch of the oscillator. With more than one cycle of non-silence, the resulting harmonic structures can be very complicated. This can produce sounds that have complex and often unrelated sets of harmonics, which gives a bell-like or clangorous timbre. The pseudo-random sequence generators mentioned earlier are one alternative method of producing sequences of single-cycle waveforms, and exactly the same length-related pitch-shifting effect happens. Chapter 4 looks at these effects in more detail.

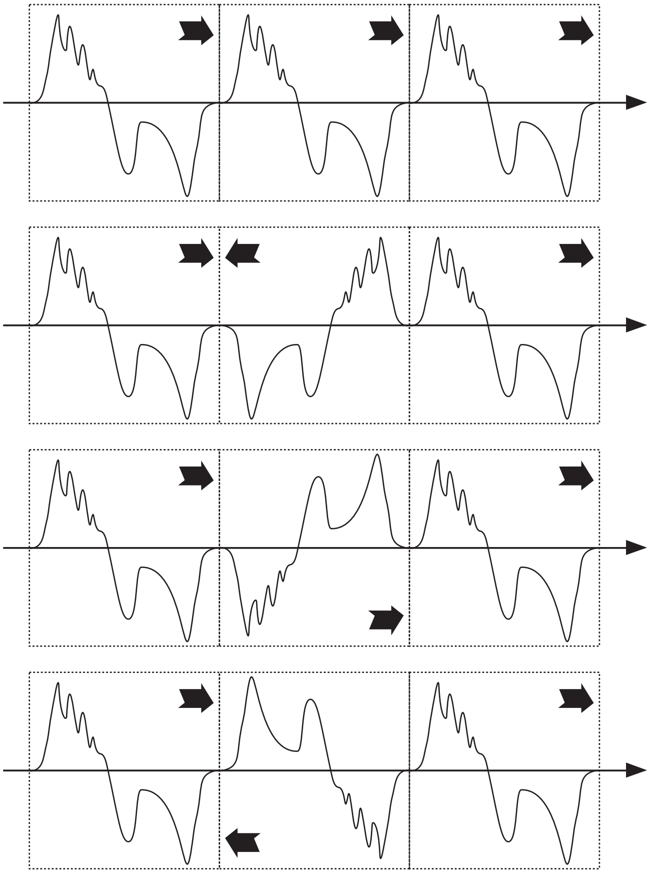
For very long sequences of cycles, the pitch-shift can become so large that the frequency becomes too low to be heard, and it is then only the individual cycles that are heard. This means that by concatenating a series of pulse waveforms that gradually change their pulse width, it is possible to produce a repeated multi-cycle waveform that sounds like a single-cycle PWM waveform. (By altering the number of repeated cycles of each different pulse width waveform, it is possible to change the effective speed of PWM. In fact, this is exactly how wavetable oscillators work; see Section 3.2.)

There are two methods for reading out the values in a wavecycle memory. When the values are accessed by a rising number, then the shape is merely repeated, whilst by accessing the same values using a counter that counts up and down, then each alternate cycle is reversed in time. This can be a powerful technique for producing additional multi-cycle waveforms from a small wavecycle memory. If repeats of the cycles can be inverted as well, then even more possibilities are available. All of these variations on a single cycle can produce changes in the spectrum of the sound: from minor detail through to major additional harmonics where the transition between the repeats is not smooth (see Figure 4.1.6 for more details).

Fixed sequences of single-cycles can be thought of as short samples, the PWM waveform is one example where a complete 'cycle' of PWM is repeated to give the same audible effect as a pulse waveform that is being modulated. Many other dynamically changing multi-cycle waveforms can be produced.

FIGURE 4.1.6

Pairs of wavecycles can be arranged in different ways by exploiting the symmetry (or lack) of the waveshapes. These four examples show a wavecycle followed by the four possibilities of reversing or inversion. The transitions between the wavecycles can be smooth or abrupt depending on the shape of the wavecycle.



Multi-cycle oscillators can also be likened to granular synthesis, since they both concatenate cycles of waveforms, although granular synthesis normally works on groups of cycles rather than individual cycles (see Chapter 5). Roland's 'RS-PCM' and many other sound-cards and computer sound generators all often use loops of multi-cycle 'samples' to provide the sustain and the release portions of an enveloped sound, the and sometimes even the attack portion. This technique is equivalent to changing the waveform of a multi-cycle oscillator dynamically, which is an advanced form of wavetable synthesis (see Section 4.2).

Multi-cycle oscillators normally have a selection of single-cycle waveshapes plus the following additional types:

- Concatenations of mathematical shapes meaning sine, triangle, square and sawtooth cycles in sequences.
- Symmetry variations of mathematical and other shapes.

- PWM waveshapes that change their harmonic content with time.
- Waveshapes that change their harmonic content with time, but not in a regular sequence (i.e. not progressively as in PWM).
- Shapes with additional non-harmonic frequencies (clangs, chimes and vocal sounds).
- Noise in other words, more cycles mean that the noise produced can be more 'white' in character than from single-cycle oscillators.

Interpolation is a method of producing gradual changes from one wavecycle shape to another, rather than the abrupt changes that occur when wavecycles are concatenated. Section 4.2 deals with this in more detail.

Multi-cycle oscillators can also be used with the velocity switching technique where different waveshapes can be mapped to the velocity with which notes are played or to any other controller. This can produce a wide variety of timbre changes, ranging from subtle to harsh, and can significantly enhance the synthesis power of a multi-cycle oscillator.

4.1.3 Samples

For very long sequences of single cycles, the complete sequence may not repeat whilst a note is being played, and it then becomes a sample rather than a multi-cycle waveform. Samples are usually held in either ROM or RAM, and because of the length, the amount of memory used can be quite large. For example, for 16-bit values, where there are 44,100 values output per second (the same as a single channel of a compact disc (CD) player) 705,600 bits (just over 86 kB) are required to store just 1 second of monophonic audio. This means that it requires a megabyte of memory to store just below 12 seconds of monophonic audio sample. Obviously, long samples will require large quantities of memory, and stereo samples will double the memory requirements. Because of this, most hybrid synthesizers of the 1970s and 1980s used very short samples, and it was only with the availability of low-cost memory in the 1990s that sampling techniques became more widespread, and this was in an all-digital form.

Trying to reduce the amount of memory that is required to store cycles affects the quality of the audio. Hybrid wavecycle synthesizers suffer from the resolution limitations of their storage. At low frequencies, there are not enough sample points to adequately define the waveshape, whilst at high frequencies the circuitry may not run fast enough. For example, suppose that a single cycle of a waveform is represented by 1024 values. At 100 Hz, this means that the VCO needs to run at 1024 times 100 Hz, which is 102.4 kHz. But at 1000 Hz, the VCO needs to run at 1.204 MHz and at 10 kHz, the VCO is oscillating at 10.24 MHz. Accurate VCOs with wide ranges, good temperature stability and excellent linearity at these frequencies are more normally found in very high-quality radio receivers. More importantly, affordable late 1970s memory technology began to run out of speed at a few megahertz. Reducing the number of

Memory size is closely related to date. In the 1980s, a megabyte was large: the first external small computer system interface (SCSI) hard drive for the 128-kB RAM-equipped Macintosh Plus computer had a capacity of only 20 MB. At the time, this was a huge amount of storage – the operating system files for a Mac would fit easily onto a 400-kB 3.5-inch single-sided floppy disk.

bits that are used to represent the waveform cycles also reduces the quality by introducing noise and distortion. The whole of the sampling process is covered in more detail in Chapters 1 and 4.

It is a common fallacy that the ability to produce complex waveshapes is all that is required to recreate any sound. In practice, most methods of producing a waveshape do not have the required resolution to provide enough control over the sound over time. Harmonics are often a more reliable guide to the sound, and it is normally the change of harmonics over time that provides the interest in most sounds. The effect of harmonics on waveshape is covered in more detail in Section 3.4.

Modifiers

When the oscillator has produced the raw source waveform, then most hybrid synthesizers pass it through a modifier section that is typical of those found on analogue synthesizers. This is usually a VCF and voltage-controlled amplifier (VCA), with associated envelope generator (EG) control. Curiously, whereas most hybrid synthesizers attempt to improve on the selection of available waveforms for the oscillator, the VCF is often still just a simple resonant low-pass filter. This puts a great deal of emphasis on the oscillator as the prime source of the timbre, and it means that the possibilities for the changing of the timbre by the modifiers are just the same as an analogue synthesizer. This means that the resonant filter sweep sound remains an audio cliché for both analogue and hybrid synthesizers. The filtering capabilities have only been enhanced significantly in some of the all-digital S&S synthesizers.

In a historical perspective, hybrid synthesizers reached a peak of popularity in the early 1980s, after polyphonic analogue synthesizers and just before the all-digital synthesizers. Many of the 'analogue' synthesizers of the mid-1990s' 'retro' revival of analogue technology are often not truly analogue, but are actually more modern hybrids, where the 'VCOs' are actually sophisticated wholly digital DCOs that use the methods described earlier to produce their waveforms, but coupled with a conventional analogue modifier section in a standard hybrid synthesis way. These updated hybrids have, in turn, been incorporated into all-digital instruments that use a mixture of synthesis methods. By the start of the twenty-first century, stand-alone hybrid synthesis had been almost entirely replaced by digital synthesis, often using modeling techniques, although there was also a new 'retro' revival for 'pure' analogue, where 'pure' often means wrapping digital retuning circuitry and chips around analogue VCOs to keep them in tune.

Manufacturers such as PPG and Waldorf were the main commercial exploiters of hardware wavetable synthesis.

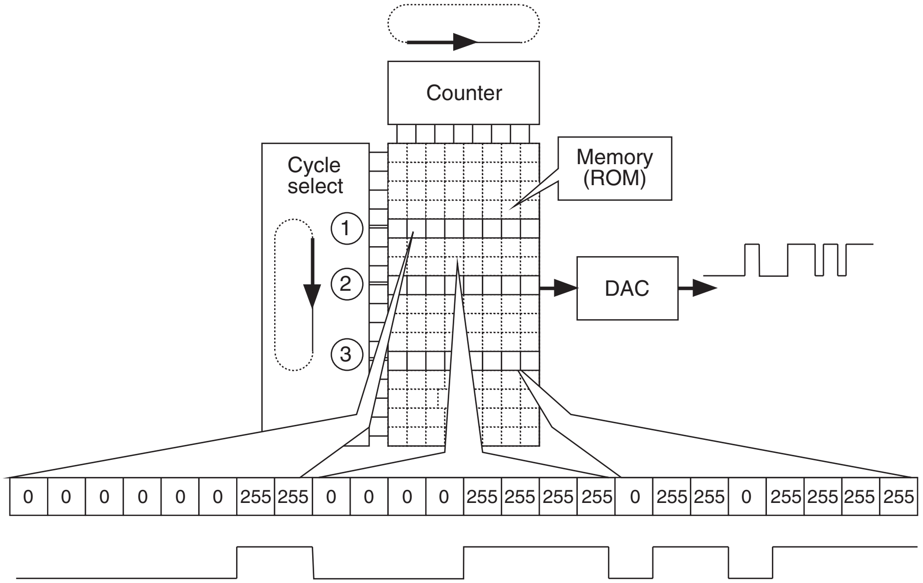
4.2 Wavetable

Initially, wavetable synthesis might appear to be very similar to multi-cycle wavecycle synthesis. Both methods use sequences of cycles to produce complex waveshapes. The major difference lies in the way the cycles are controlled. In multi-cycle wavecycle synthesis, the chosen sequence of cycles is repeated continuously, whereas in wavetable synthesis, the actual waveform that will be used can be chosen on a cycle-by-cycle basis. This is a very significant difference and makes wavetable synthesis very powerful, and more like granular synthesis than wavecycle synthesis or sample replay. Curiously, despite this flexibility, wavetable synthesis has seen only limited commercial success, and sampling is often seen as making it redundant, when this is not actually the case – wavetable synthesis is arguably the general case of which sampling is a special case.

4.2.1 Memory

Wavetable synthesis is based on memory even more strongly than wavecycle synthesis. In wavecycle synthesis, there are a few methods that do not use large quantities of memory, for example, pseudo-random sequence-based waveform generators. But wavetable synthesis uses the memory as an integral part of the synthesis process, since the cycle being used is dynamically selected by controlling the memory.

Just as with single-cycle wavecycle synthesis, a cycle of a waveform is stored in a memory chip, and successive values are retrieved from the memory and sent to a DAC where they produce the output waveform (Figure 4.2.1). The values are retrieved in order by using a counter that steps through the memory



0	0	0	0	0	0	255	255	0	0	0	0	255	255	255	255	0	255	255	0	255	255	255	255

FIGURE 4.2.1 A wavetable synthesizer uses several wavecycle locations in the memory: accessing each in turn. In this example, the cycle select logic sequentially selects wavecycles 1, 2 and 3, and then repeats this continuously. The output thus consists of three concatenated wavecycles. For simplicity the values shown are just 0s and 1s, but they could be 8-, 12- or 16-bit values, depending on the required precision.

in an ascending sequence of memory locations. Determining which values the counter steps through is set by controlling which part of the memory is used. In a single-cycle wavecycle oscillator, the control signals are set to point to the specific single-cycle waveform, and the oscillator then outputs that waveshape continuously.

In a wavetable oscillator, the control signals that determine where the waveform information is stored can be changed dynamically as the oscillator is outputting the waveshape. Normally the changes are made as one cycle ends and another begins, so that the waveshape does not change mid-way through a cycle. The control signals that set where the cycle is retrieved from can be thought of as modulating the shape of the output waveform, although they are really just pointing to different parts of the memory. The name 'wavetable' comes from the way that the memory can be thought of as being a table of values, and therefore the control signals just point to cycles within that wavetable. There are two basic ways that the cycle being used in the table can be changed: swept and random-access.

Swept

One common usage for swept wavetables is to emulate a waveform that has been passed through an enveloped resonant filter.

By incrementing the pointer so that it points to successive cycles in the wavetable, the control signals effectively 'sweep' the resulting waveshape through a series of waveforms. The fastest rate at which this can happen is when only one cycle of each waveform is used before moving to the next waveform, although by omitting waveforms the sweep speed can be increased. Wavetables that are intended to be swept in this fashion are normally arranged so that the waveforms are stored in an order where similar sounding waveforms are close together. This produces a 'smooth' sounding change of waveshape and harmonics as the table is swept. Large changes of harmonics or sudden changes of waveshape can produce rich sets of harmonics, and this is catered for by allowing sweeps to occur over the boundaries between these groups of similar timbres. For example, a series of added sine waves might be followed by a group of pulse waves in the table, and a sweep that crossed over between the two groups would have large changes between the two sections.

Random-access

Swept wavetables with large numbers of tables in the series begin to approach sampling, where a long sample has much the same ability to produce a waveform that changes with time. But wavetable is not restricted to this type of sweep. By allowing the pointer to be set to point to anywhere in the table for each successive cycle, any cycle can be followed by any other cycle from the wavetable. This is called random-access, since any randomly chosen cycle in the table can be accessed. By supplying a series of pointers, the waveform can be swept, and therefore a sweep is in fact a special case of structured, 'random' access. More normally, a series of values are used to make the pointer access a sequence of waveform cycles. This can be a fixed sequence, in which case the wavetable behaves as a multi-cycle wavecycle oscillator, or a dynamically changing sequence of pointer locations, in which case the modulation of the waveform is characteristically that of a wavetable (Figure 4.2.2).

The ability of wavetable synthesis to control exactly which cycle is played at any time is closely related in many ways to granular synthesis (see Section 5.6).

4.2.2 Table storage

The actual storage of the waveforms inside the wavetable can be of several forms. Some hybrid oscillators only have one of these types. Others provide two types or all three. The naming conventions differ with each manufacturer: wavesamples and hyperwaves are just two examples of names used for samples and multi-cycle waves, respectively. Some types are found in analogue/digital

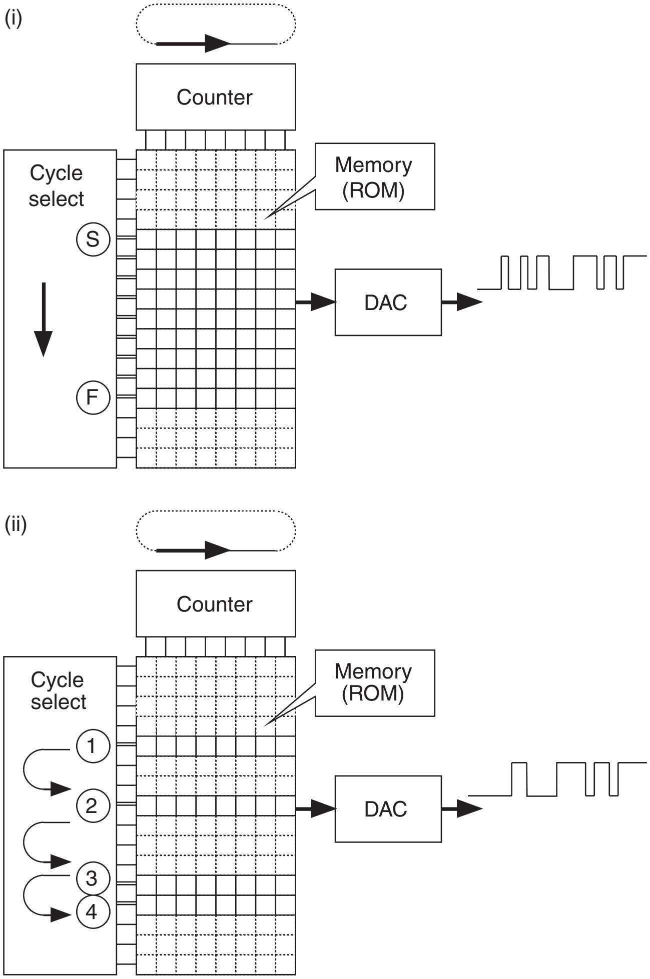


FIGURE 4.2.2
(i) A swept wavetable outputs each of the cycles between the start and finish points in the memory. (ii) A random-access wavetable only outputs the specific cycles which have been chosen.

hybrids as well as in digital instruments that emulate analogue hybrids, whilst the more complex types are only found in digital instruments. The major types of table storage are as follows:

- Single-cycle wavetable oscillators provide large numbers of single-cycle waveforms and can be implemented in hybrid or digital technologies. The technique of rapidly changing cycles can be used to provide results that are similar to granular synthesis.
- Multi-cycle wavetable oscillators contain waveforms with more than one cycle and can be implemented in hybrid or digital technologies, but are found mostly in digital instruments.
- Samples are just longer multi-cycles, although the implication is that the sample plays through once or only partially, whilst a multi-cycle waveform is usually short enough to be repeated several times in the course of a note being played. Some samples in wavetables are provided with multiple start points, which means that the sample can be played in its entirety, or that it can be started mid-way through. This can be used to provide a single sample that can be used as an attack transient sound with a sustain section following, or as just a sustained sample by playing the sample from the start of the sustain portion. The section on using S&S in this chapter contains more detail on these techniques.
- Sequence lists or wave sequences are the names usually given to the sequential set of pointers to cycles or samples in the wavetable. This list determines the order in which the cycles or samples will be replayed by the oscillator. Lists can automatically repeat when they reach the end, reverse the order when they reach the end or merely loop the last cycle or sample. Some sequences allow looping from the end of the list to an arbitrary point inside the sequence list, which allows a set of cycles or samples to be used for the attack portion of the sound, whilst a second set of cycles or samples is used for the sustain and the release portions of the sound. This interdependence of the oscillator and envelope is common in sample-based instruments, whereas in analogue synthesis the VCO and the EG are normally independent.

The Korg Wavestation is an example of a synthesizer that allows samples to be sequenced.

- *Mixed modes*: Some hybrid wavetable oscillators allow mixtures of single-cycle, multi-cycle and sample waveforms to be used in the same sequence list. Additional controls like repetitions of single- or multi-cycle waveforms, or even the length of time that a sample plays, may also be provided.

Multi-samples

The term 'multi-samples' can be applied to the result of a sequence list that causes samples to be played back in a different order to that in which they were recorded. These samples may be looped, in which case some interaction with an EG is usually used to control the transitions between the individual looped samples. Roland's 'RS-PCM' and many other sound-cards and computer sound generators all often use loops of multi-cycle 'samples' to provide the sustain and the release portions of an enveloped sound, and sometimes even the attack portion. This technique is equivalent to changing the waveform of a multi-cycle oscillator dynamically, which is an advanced form of wavetable synthesis.

'Multi-samples' are also used in samplers to mean the use of several different samples of the same sound, but taken at different pitches.

Loop or wave sequences

A loop or wave sequence is the name for the sequence of samples that are used in a multi-sample. It provides the mapping between the envelope segments and the samples that are looped in that segment (Figure 4.2.3). Loop sequences are sometimes part of a complete definition which includes the multi-sample and

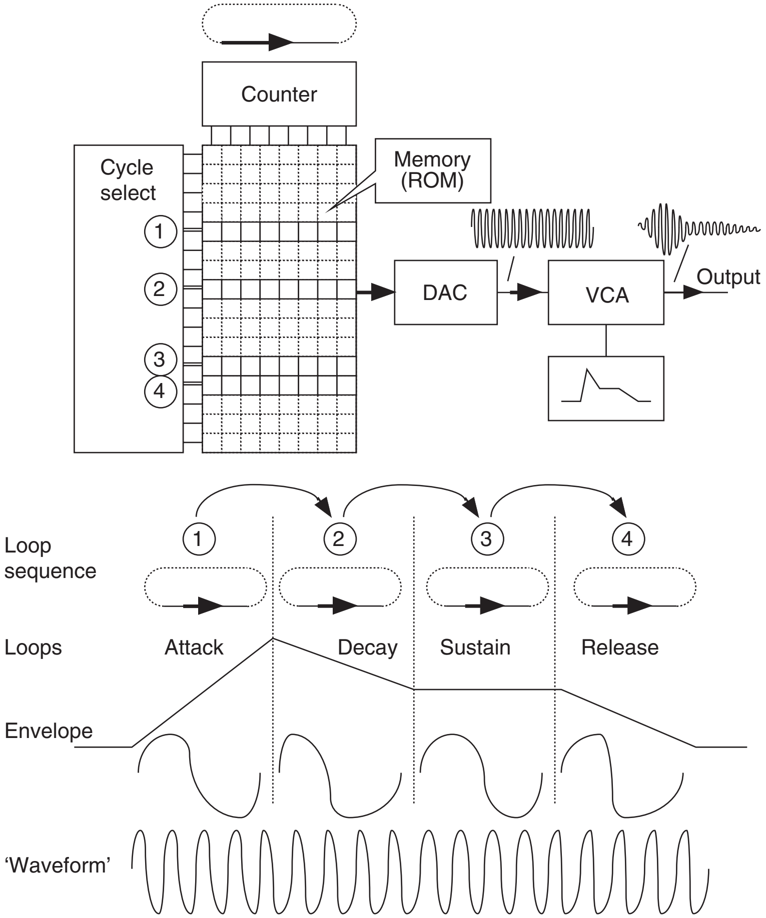


FIGURE 4.2.3

Loop sequences control the order in which looped wavecycles are replayed. In this example, the loop sequence is controlled by the envelope. Wavecycle 1 loops during the attack part of the envelope, followed by wavecycle 2 during the decay segment. The sustain segment is produced by looping wavecycle 3, and wavecycle 4 loops during the release part of the envelope.

loop-sequence information: for example, the musical instrument definitions in Apple's QuickTime and Roland's 'RS-PCM' are stored in this way.

When samples are looped as part of a sequence, then the playback time will normally be set by the sequence timing and can be made uniform across the keyboard span. So in a string sound consisting of a bowing attack sound and a sustain vibrato sound, the bow scrape attack might last the same time regardless of the key played on the keyboard, even though the pitch will change. This is the exact opposite to how most samplers work: in a sampler, the pitch and length of time which the sample plays for are normally directly related. Therefore higher pitched sounds tend to have shorter attacks and decays. The solution in a sampler is to provide several different samples taken at different pitches. These are also called 'multi-samples'. At the end of the twentieth century, sample-replay techniques that combined these two contrasting approaches were developed, and the link between pitch and time was largely removed.

Interpolation tables

Sequences of samples do not need to abruptly change from one sample to the next. By taking two differently shaped wavecycles or samples from a wavetable and gradually changing from one shape to another, the harmonic content can be dynamically changed, and the audible effect is like cross-fading from one sound to the other. The initial cycle will contain all of the values from one of the two cycles or samples, and the final cycle will contain only the values from the other cycle or sample (Figure 4.2.4). The process of changing from one set of values to another is called interpolation. Interpolation is mathematically intensive but requires only small amounts of memory to produce complex changing timbres.

Interpolating between two waveshapes does not always produce a musically useful transition because the changes in harmonic content may not be too great, and the result does not sound smooth and predictable. The relationship

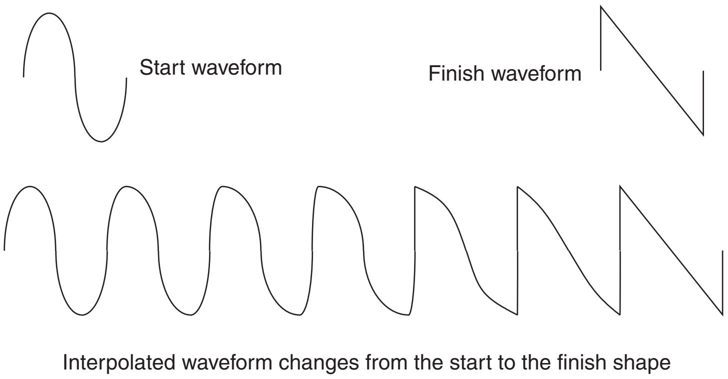

Start waveform
Finish waveform
Interpolated waveform changes from the start to the finish shape

FIGURE 4.2.4 Interpolation allows two waveforms to be defined as start and finish points, and the 'in-between' wavecycles are then calculated (or interpolated) to produce a smooth transition between the two waveforms.

between the shape of a waveform and its harmonic content is not a simple one, and minor changes in the shape of a waveform can produce large changes in the harmonic content. Interpolation can emphasize this effect by producing timbres which change from one sound to another, but that pass through many other timbres in the process, rather than the smooth 'evolution' that might be expected.

Rather than using interpolation as a mathematical transformation of information about a waveform, a much more satisfactory method is to use interpolation to produce the changes between one spectrum and the other. This method does produce smooth changes of timbre (or very unsmooth changes, depending on the wishes of the user!). This sort of spectral transformation is described in Section 5.7.

4.2.3 Additional notes

- Wavetable synthesis is a term used to cover a wide range of techniques, and as a result, there are as many definitions of wavetable synthesis as there are techniques.
- The differences between single-cycle wavetable synthesis and sampling are actually greater than the differences between multi-cycle wavecycle and sampling. Very few samplers have the facility to alter the order in which the cycles or other fragments of a sample are played back!
- Sound-card manufacturers tend to describe almost any hybrid technique as 'wavetable' synthesis.
- By loading a wavetable oscillator with a set of multi-cycle waveforms that have been generated from the addition of sine waves, an additive synthesis engine can be produced using hybrid digital and analogue techniques.

4.2.4 Sample sets

The samples that are provided in wavetable and wavesample-based hybrid synthesizers can have a great effect on the sound set that is possible to create. A determined programmer will see the exploitation of even the sparsest of sample sets as a challenge. Therefore one of the first tasks for a synthesist, who wishes to explore the sound-making possibilities of a hybrid instrument, is to become familiar with the supplied sound resources.

Typical wave and sample sets come in very standardized forms, mainly due to the twin influences of the general musical instrument digital interface specification (General MIDI (GM)) and history. GM has meant that most instruments need to contain sufficient samples to produce the 128 GM sounds, although the GS and XG extensions include more sounds and additional controllers. This means that otherwise serious professional instruments will also have sounds that are meant to be a bird tweet, a telephone ring, a helicopter and applause in their sample set. History dictates the inclusion of instruments like key-click organs, harpsichords, accordions and some now-unusual percussion sounds-all of which have become somewhat clichéd. Some manufacturers have used synthesis to create these sounds, but the results can be very different to the GM standard samples, which can be exploited by a synthesist, of course.

The provision of a good piano sound is also frequently obligatory, especially in a keyboard that is expected to have broad appeal, like a workstation. Only in a few 'pure' synthesizers has the piano sound ever been omitted. Unfortunately, a good piano sound requires large amounts of memory, often a significant proportion of the total sample set. It can also be very difficult to utilize piano sounds as the raw material for anything other than piano sounds.

There are three main types of samples that are found in sample sets:

1. *Pitched samples* are sounds that have a specific frequency component (the note that you would whistle).
2. *Residues* are the non-pitched parts of a sound: the hammer thud, fret-buzz, string-scrape, and soon. Often produced by processing the original sound to remove the pitched parts.
3. *Inharmonics* are unpitched, noisy, buzzy or clangorous sounds.

Piano samples will often start with several pitched multi-samples at different pitches. Piano residues will be one or more hammer thuds and harp buzz sounds. These can be useful for adding to other instrument sounds or for special effects when pitch shifted. Piano and electric piano residues can be good for sound effects; they sound like metal tapping and clunks.

'Classic' or 'analogue' samples will be the basic square, sawtooth and pulse waveforms, possibly a sine, often some samples of actual waveforms from real instruments, and sometimes some residues. These are intended to be used in emulations of analogue synthesizers.

Strings and vocal sounds will be looped sustained sounds, but some may have looping artifacts and cyclic variations in timbre: audition all of the samples and listen closely to the sustained sound as it loops. If a loop does exhibit a strong artifact, then consider using that as part of a rhythmic accompaniment.

Woodwind sounds can be used as additional waveforms for analogue emulations, as well as thickening string sounds.

Plucked and bass sounds can be pitch shifted up to provide percussive attacks or shifted down for special effects.

Percussive sounds can be used as attack segments or assembled into wave sequences to provide rhythmic accompaniment.

Digital waveforms come in either PWM wave sequences or sequences with varying harmonic content. These can be used in cross-faded wave sequences to provide movement in the sound. Samples that are large enough to contain complete cycles of PWM beating are rare, whereas single cycles of waveforms are small and therefore numerous.

Gunshots, sci-fi sirens, laughter and raindrops can be pitch shifted downwards to provide atmospheric backdrops.

4.3 DCOs

DCOs are the digital equivalent of the analogue VCO. DCOs have much in common with wavecycle synthesis. In fact, they can be considered to be a special case of the most basic wavecycle oscillator: one which produces only the 'classic' synthesizer waveforms (sine, square, pulse and sawtooth).

DCOs are combinations of analogue and digital circuitry and design philosophies; they are literally hybrids of the two technologies. They were originally developed in order to replace the VCOs used in analogue synthesizers with something that had better pitch stability. The simple exponential generator circuitry used in many early analogue VCOs was not compensated for changes in temperature, and therefore the VCOs were not very stable and would go out of tune. Replacing the VCOs with digitally controlled versions solved the tuning stability problems but changed some of the characteristics of the oscillators because of the new technology. The DCO is also notable because it marked the final entry of the era of three-character acronyms: VCO, VCF, LFO and VCA. Subsequent digital synthesizers moved away from acronyms towards more accessible terms such as oscillator, filter, function generator and amplifier.

It is worth noting that although early VCO designs did suffer from poor pitch stability, the designs of the late 1970s gradually improved the temperature compensation, and the custom chips developed in the early 1980s had excellent stability. But the damage to the reputation of analogue VCOs had been done, and DCOs replaced the VCO permanently for all but the most purists of analogue users. DCO-based synthesizers were also used by frequency modulation (FM) in the mid-1980s, and the DCO-based sample player is the basis of all S&S synthesizers.

In a curious looping of technology, the modeled analogue synthesizers of the 2000s use DCO-based sample-replay to replay samples that are based on mathematical models of analogue VCOs of the 1970s.

4.3.1 Digitally tuned VCOs

The simplest DCOs are merely digitally tuned VCOs. A microprocessor is used to monitor the tuning of the VCOs and retune it when necessary. This process was usually carried out when the instrument was initially powered up, although it could be manually started from the front panel. The technique isolates the VCO from the keyboard circuitry (see Chapter 7) and then uses a timer to measure the frequency by counting the number of pulses in a given time period. This measurement process is carried out near the upper and lower frequency limits of the VCO by switching in reference voltages, with additional measurement points as required to check the tracking of the VCOs (see Chapter 2). From this information, the two main adjustments can be calculated as follows:

1. An 'offset' voltage can be generated to bring the VCO back into 'tune'.
2. A 'tracking' voltage can be used to set the tracking.

The offset and tracking voltages for all the VCOs in a synthesizer would be stored in battery-backed memory. This technique is often known as 'auto-tune' (Figure 4.3.1).

Some analogue polysynths need to be left alone during auto-tuning. Moving the pitch-bend wheel on some 1970s' examples whilst auto-tuning was in progress could put the entire instrument out of tune.

A variation of this technique is currently used in some analogue-to-digital converter (ADC) chips, where the circuit monitors its own performance and recalibrates itself continuously for each sample. In early auto-tune synthesizers, the time required to measure the frequencies at the various points meant that it was not possible to continuously tune the VCOs – it could take several minutes to retune all of the VCOs in a polyphonic synthesizer. It would be possible to arrange the voice allocation scheme so that a VCO that was out of tune could be removed from the 'pool' of available voices, but this would effectively reduce the polyphony by one. Some synthesizers allowed voices to be disabled in exactly this way if they could not be tuned correctly by the auto-tune

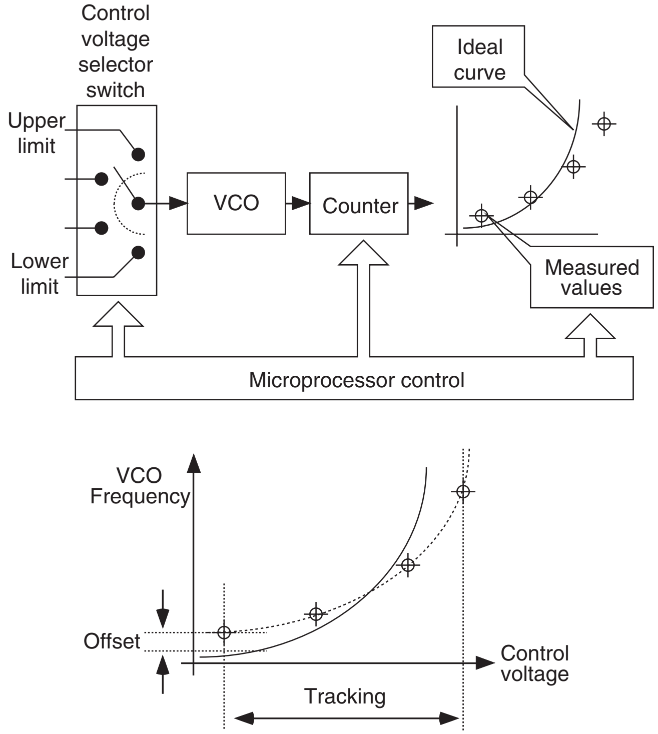


FIGURE 4.3.1 In an 'auto-tune' system, a microprocessor sends a series of control voltages to the VCO, and compares the output frequencies with the ideal values. These numbers are then used to provide offset and tracking adjustments to the VCO, so that its response matches the ideal curve.

circuitry, which explains the poor reliability reputation of the VCOs in some early polyphonic synthesizers.

Another combination of digital microprocessor technology with analogue VCOs occurs in synthesizers where the keyboard is scanned using a microprocessor, and the resulting key codes are turned into analogue voltages using a DAC and then connected to the VCOs. Although suited to polyphonic instruments, this technique has been used in some monophonic synthesizers, particularly where simple sequencer functions are also provided by the microprocessor. The Sequential Pro-One is one example of a monophonic instrument that uses 'digital' storage of note voltages, and the use of a microcontroller chip allows it to provide two short sequences with a total of up to 32 notes clocked by the LFO.

4.3.2 Master oscillator plus dividers

A nearer approach to an all-digital 'true' DCO uses ideas taken from master oscillator organ chips. A quartz crystal-controlled master oscillator provides a high-frequency clock, which is then divided down to provide lower rate clocks through a series of divider chips (Figure 4.3.2). By using a high rate of master

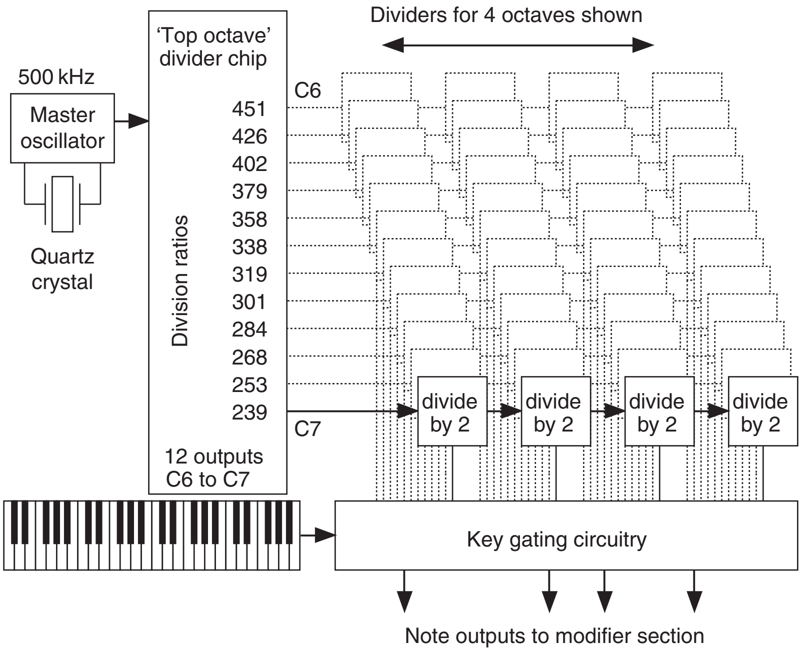


FIGURE 4.3.2 A 'top octave' divider systems uses a high-frequency master oscillator and dividers to provide all the required frequencies for all the notes on a keyboard. In this example, the master oscillator frequency is 500 kHz, and the division values required to produce the 12 top notes in an octave are shown. Each of these frequencies then needs to be further subdivided to produce the lower octave notes.

clock and correspondingly large divider ratios, it is possible to generate all the frequencies that are required for all the notes for a synthesizer from just a few chips. These notes can then be gated from the keyboard circuitry, with the end result being a polyphonic oscillator that is derived from a stable crystal-controlled master oscillator.

Rather than having separate stages of dividers for each note, an obvious design simplification is to have 12 dividers that produce the highest required frequencies, and then divide each of these outputs successively by 2 to produce the lower octaves (this presupposes that the scale used will be fixed, usually equal temperament, and that each octave has identical ratios between the notes). This is called a 'top octave' method, and with the right division values and a high enough clock, it gives very good results. For example, with a master clock frequency of 500 kHz, the division required to produce a C#6 at 1108.73 Hz is 450.96, which is almost exactly 451, whilst for the next note, the D6 at 1174.66 Hz, the division is 425.65, and therefore an integer value of 426 will produce an output frequency that is slightly too low. In a real-world design, you might expect that the clock frequency and division values would be chosen to minimize the errors by setting real division values that are as near to integers as possible. In practice, a real-world custom top octave synthesizer chip, the General Instruments (GI) AY1-0212A used exactly these division values, as shown in Table 4.3.1.

As seen in table 4.3.1, using integer dividers only makes a slight difference in the output frequency. Using individual separate division stages only improves the accuracy slightly. Taking the values for the C1, C5 and C7 division values given earlier, if the 239 value for the C7 'top octave' division is then divided down by successive dividers, this is equivalent to doubling the effective division value – which would thus have the values of 956 for the C5 frequency and 15,296 for the C1 frequency. The 956 division value is identical to the one used, but the 15,296 is slightly too large, which means that output frequency will be too low – by about 0.0149 Hz or 0.00045 cents. The difference in division values thus produces only very slight differences in the output frequencies. For comparison purposes, the human ear can detect pitch changes of a minimum of about 5 cents, whilst the E-mu Morpheus had fine tuning steps of 1.5625 cents, the Yamaha SY99 had micro-tuning steps of 1.171875 cents and the MIDI Tuning Standard has steps of 0.0061 cents.

Changes in the pitch of these types of 'master oscillator plus divider' DCO might be achieved by using a voltage-controlled crystal oscillator to make minor changes in pitch for pitch-bend or vibrato effects, but it is difficult to change the frequency of a crystal oscillator enough for satisfactory pitch control. A more satisfactory method uses a rate adapter, which is a counter-based circuit that removes just one clock occasionally from a continuous clock signal and therefore reduces the effective clock rate. This 'gapped' clock needs to be followed by the equivalent of a low-pass filter to remove the effects of the jitter in the clock pulses, but this type of DCO has just such filters in the form of divider circuits.

Table 4.3.1 DCO Dividers

Clock (Hz)	Note	Note Frequency (Hz)	True Divider	Integer Divider	Actual Note Frequency (Hz)	Frequency Error (%)	Frequency Difference (Hz)
500,000	C#0	17.3239	28861.84	28862	17.3238	−0.0006	9.8E-05
500,000	D0	18.354	27241.95	27242	18.354	−0.0002	3.6E-05
500,000	D#0	19.4454	25712.97	25713	19.4454	−0.0001	2E-05
500,000	E0	20.6017	24269.82	24270	20.6016	−0.0008	0.00016
500,000	F0	21.8268	22907.66	22908	21.8264	−0.0015	0.00033
500,000	F#0	23.1247	21621.95	21622	23.1246	−0.0002	5.6E-05
500,000	G0	24.4997	20408.4	20408	24.5002	0.00196	−0.0005
500,000	G#0	25.9565	19262.97	19263	25.9565	−0.0002	4.7E-05
500,000	A0	27.5	18181.82	18182	27.4997	−0.001	0.00027
500,000	A#0	29.1352	17161.35	17161	29.1358	0.00205	−0.0006
500,000	B0	30.8677	16198.16	16198	30.868	0.00098	−0.0003
500,000	C1	32.7032	15289.03	15289	32.7033	0.00017	−6E-05
500,000	C#4	277.183	1803.865	1804	277.162	−0.0075	2.07686
500,000	D4	293.665	1702.622	1703	293.6	−0.0222	0.06524
500,000	D#4	311.127	1607.061	1607	311.139	0.00379	−0.0118
500,000	E4	329.628	1516.863	1517	329.598	−0.009	0.02967
500,000	F4	349.228	1431.728	1432	349.162	−0.019	0.06622
500,000	F#4	369.994	1351.372	1351	370.096	0.02751	−0.1018
500,000	G4	391.995	1275.525	1276	391.85	−0.0372	0.14591
500,000	G#4	415.305	1203.935	1204	415.282	−0.0054	0.02231
500,000	A4	440	1136.364	1136	440.141	0.032	−0.1408
500,000	A#4	466.16	1072.593	1073	465.983	−0.0379	0.17678
500,000	B4	493.88	1012.392	1012	494.071	0.03869	−0.1911
500,000	C5	523.25	955.5662	956	523.013	−0.0454	0.23745
500,000	C#6	1108.73	450.9664	451	1108.65	−0.0074	0.08255
500,000	D6	1174.66	425.6551	426	1173.71	−0.081	0.95108
500,000	D#6	1244.51	401.7645	402	1243.78	−0.0586	0.72891
500,000	E6	1318.51	379.2159	379	1319.26	0.05694	−0.7512
500,000	F6	1396.91	357.9329	358	1396.65	−0.0188	0.26196
500,000	F#6	1479.98	337.8424	338	1479.29	−0.0466	0.69006
500,000	G6	1567.98	318.8816	319	1567.4	−0.0371	0.58188
500,000	G#6	1661.22	300.9836	301	1661.13	−0.0054	0.09043
500,000	A6	1760	284.0909	284	1760.56	0.032	−0.5634
500,000	A#6	1864.66	268.1454	268	1865.67	0.05422	−1.0116
500,000	B6	1975.53	253.0966	253	1976.28	0.03818	−0.7546
500,000	C7	2093	238.8915	239	2092.05	−0.0454	0.94979

Based on a GI AY1-0212A TOS chip.

For a 4.096-MHz clock, removing just one clock pulse with a rate adapter can be thought of as changing the effective frequency to 2.048 MHz whilst the clock pulse is missing, but then the next clock pulse restores the frequency to 4.096 MHz again (Figure 4.3.3). The actual number of pulses per second (which is what frequency measures) varies depending on when and how the measurement is taken. If the frequency is measured by timing from the start of one clock pulse to the start of the next, then the frequencies of 4.096 and 2.048 MHz are correct. The brief change in frequency is large when measured in this way, but by measuring more than one clock pulse, the change in frequency reduces as the number of clock pulses used for the measurement increases. This process of averaging the frequency over several clock cycles is just what happens when a divider circuit is used to divide down the output of a rate adapter.

The missing clocks are an extreme form of the random variation in the time between successive clock edges in a digital system called jitter. Caused by unstable clock sources, or noise affecting the switching point of gates, jitter usually varies randomly around the true edge position: some clocks are slightly closer together, whilst others are slightly further apart. More complicated circuits like accumulator/divider circuits can provide very small frequency changes without the need for large numbers of divider stages to filter out the jitter.

The main indicator of the use of this type of 'master oscillator plus dividers' DCO is the presence of global pitch control. If you change the master oscillator frequency, then all of the derived notes change too. Because of this 'global' pitch change, synthesizers that have this type of DCO do not usually provide

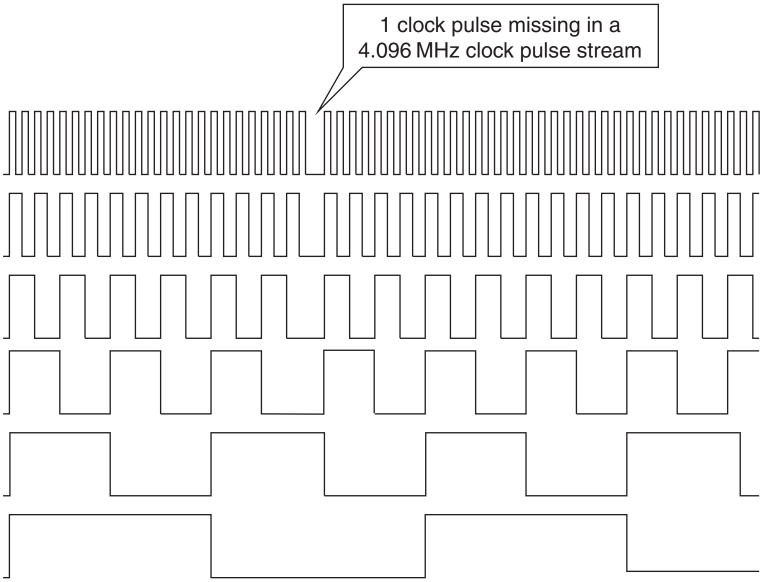


FIGURE 4.3.3
Dividers can be used as filters. In this example, a single pulse is missing from the 4.096-MHz clock. Subsequent divide-by-2 stages reduce the effect of the missing clock by 'averaging' out the frequency.

pitch envelopes that change the pitch of a note, since any new notes that are played would pitch bend any pre-existing held notes, which is not very useful musically. The pitch bend and vibrato normally affect all notes that are being played, and individual pitch control for pitch bend or vibrato on a 'per voice' basis is more unusual.

The upper frequency limit of many top octave synthesizer chips was limited; for a 500-kHz input, the chip mentioned earlier can only produce a C7 at 2093 Hz, which is an octave below the top note of an 88-note piano keyboard. In addition, having lots of simultaneous frequencies produced by a large number of divider chips can induce a characteristic buzzing sound in the audio output if care is not taken with wiring layout and circuit board design – the commonly used onomatopoeic term for this problem is 'beehive' noise. 'Electronic pianos' and 'string machines' of the mid-1970s that used top octave synthesis were notorious for this extraneous noise.

Another useful hint that this type of DCO is being used is the lack of any 'detune' facility if there is more than one DCO provided in the voice. Since the only way to provide fine resolution pitch changes is by using the rate adapter (of which there is usually only one), which produces global pitch changes, then it is not possible to achieve the slight 'detuning' effects of two VCOs. By using two rate adapters and two sets of divider circuits, it is possible to produce detune, but this almost doubles the required circuitry. Many 'master oscillator plus divider' synthesizers provide 'sub-oscillators' that are merely the output of the gated notes divided by 2 or 4 to give extra outputs that are one or two octaves down in pitch from the main output. Chorus is often also provided to try and reproduce the effect of detuned VCOs (see Chapter 6).

4.3.3 Waveshaping

The basic output of most simple DCOs is a pulse or square wave at the required frequency. In order to emulate a conventional VCO, this needs to be converted into the 'classic' analogue waveforms: sine, square, sawtooth, triangle and pulse. This can be done using analogue electronics, but a much more flexible system can be achieved by using wavecycle/wavetable techniques. By setting the DCO to produce an output that is much higher in frequency, a look-up table can be used to store the values for each point in the waveform and a DAC can be used to produce the required waveforms. The purity of the waveforms produced is limited only by the number of bits used to represent each point on the waveform, and the highest output frequency of the DCO, which sets the number of points that can be used (Figure 4.3.4).

Since several of the 'classic' analogue waveforms have lots of symmetry, the number of points that need to be stored in order to produce a single cycle can be minimized. For a square wave, it can be argued that only two values need to be stored, but this is inadequate for the remaining waveforms. Whilst the sine and triangle waveforms can be perfectly described with only a quarter of

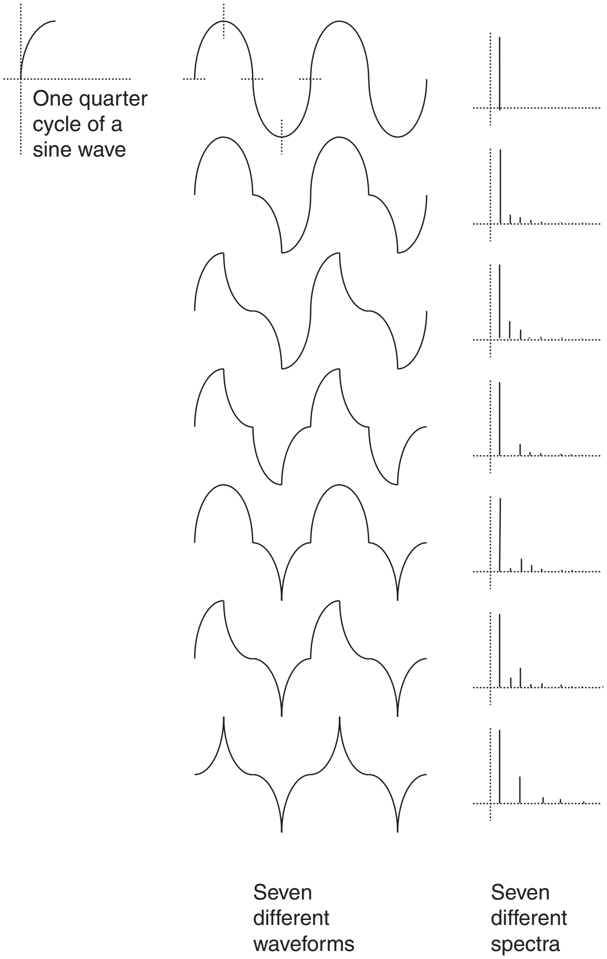


FIGURE 4.3.4 Symmetry can be used in a wavetable synthesizer to produce many waveforms from a small segment of a complete waveform. In this example, a quarter cycle of a sine wave is used to generate a sine waveform plus six other waveshapes.

a cycle, the sawtooth and pulse waveforms require at least half a cycle. Using 256 points to define a half cycle of the waveform, is thus the equivalent of having 512 separate stored points, which means that the DCO needs to run at 512 times the cycle frequency. By exploiting the symmetry or asymmetry of waveforms, a number of waveforms can be produced by using the same set of points.

Sometimes, 8-bit values are used to store the waveform point values, but for only a doubling of the memory requirements, 16-bit values give a huge increase in the perceived quality (the doubling of the number of bits produces a disproportionately large increase in the audio quality, see Chapter 4). For high-pitched notes, the whole of the wavetable need not be used, since only one or two harmonics will be audible, and therefore less points are required in the table; this can be achieved by only using every other value, or perhaps even missing out three points, and only using every fourth value.

4.3.4 High-resolution DCOs

By the 1990s, DCOs were using higher frequency oscillators and similar division techniques (now using programmable divider chips) to those of the mid-1970s, but with much finer resolution, sufficient to provide frequency steps so small that they were almost inaudible. They also had multiple dividers so that each voice can have an effectively independent DCO. Higher clock speeds, often higher than the CD sample rate of 44.1 kHz – approximately 48 or 62.5 kHz are frequently used for the master clock. These enhancements removed all the problems described earlier for the 'master oscillator plus divider' type of DCO and gave a tone generation source that has almost ideal performance – limited only by the master clock rate and the precision of the dividers. Most of these improvements were due to the availability of faster chips and bigger division ratios rather than any major changes in design.

The frequency steps from realizable oscillators depend on the number of bits that are used to control the frequency changes through the programmable dividers. With 20 bits of divider resolution, it is possible to have frequency steps of 0.3% at 20 Hz and 0.005% at 1 kHz, using a basic clock rate of 62.5 kHz. For comparison with the 500-kHz clocks of the 1970s, here are some mid-1990s' figures: the Roland D50 uses 32.768 MHz for its tone generator Application-Specific Integrated Circuits (ASICs), the Yamaha FB01 uses a 4-MHz clock and a 62.5-kHz sample rate, whilst the Yamaha SY99 uses 6.144-MHz clocks for its tone generator chips and a 48-kHz sample rate. In the 2000s, digital signal processing (DSP) chips and even general-purpose microprocessors were used as tone generators, with clock speeds of hundreds of megahertz. The sample rate has remained at 48 kHz, with some examples using 96 kHz.

4.3.5 Minimum frequency steps

The most important pointer to a good DCO design (apart from temperature stability) is the minimum step in frequency that can be made. This is most apparent when the pitch-bend control is used. Some DCOs have audible jumps or steps in pitch, which shows that insufficient frequency resolution is available. A more rigorous method of verifying the size of the frequency steps can be achieved by detuning two DCOs so that they beat together.

The pitch differences required for slow beating are quite small. For example, if two frequencies are 1 Hz apart, then they will beat once every second. If they are 0.1 Hz apart, then the beat will cycle once in every 10 seconds. For a pitch difference of 0.01 Hz, the beat will take 100 seconds to complete 1 cycle.

Therefore, to measure the minimum frequency step, you leave one DCO unchanged, and apply the smallest pitch change that you can produce to the other; this is probably not going to be audible, but by listening to the beats you can hear when the two DCOs go from the same frequency (no beats) to slightly different frequency, when the beating starts. By timing the length of one cycle of the beat, you can work out the difference in frequency.

4.4 DCFs

Digital control and tuning stability is not as important to a VCF as a VCO for most applications (except for FM, and playing the VCF as a sine wave VCO), but the launch of digital instruments like the Yamaha DX7 meant that 'digital' became an essential buzz-word, and 'analogue' acquired an association with 'previous generation' and 'poor stability', and therefore 'digitally controlled' was incorporated into marketing speak, and DCOs and digitally controlled filters (DCFs) quickly replaced VCOs and VCFs on specification sheets. A DCF is an analogue filter (often a VCF) where the cut-off frequency and the *Q* (or resonance) can both be digitally controlled. A DAC is used to convert the digital number, representing the cut-off or *Q* value, into a control voltage, and this then controls the VCF. The minimum frequency step (the smallest control voltage change) produced by the DAC is important in a DCF because filter sweeps, particularly the resonant ones, can make jumps in cut-off frequency audible (although it can be used as a special effect). Because sweeping of frequency is common in VCFs (DCFs are almost the opposite: notes need to be steady whilst being played) if the DAC output is not filtered sufficiently, then the onomatopoeically named 'zipper' noise may be heard. DCFs and DCOs thus have different design criteria.

Hybrid mixtures of analogue and digital circuitry can also be used in filters. Some designs from the 1970s used an interesting method to produce variable resistors in analogue filters. The design used chips that allowed digital control of switches, and by turning these switches on and off at an ultrasonic frequency and changing the duty cycle, the effective resistance was changed.

The twenty-first century 'retro' instruments with 'analogue audio paths' typically use analogue VCF chips normally with digital control of parameters, just as in the DCFs of old, but here the term 'analogue' is a positive marketing term once again.

The Alesis Andromeda and Bob Moog's Little Phatty are just two examples of hybrid synthesizers with 'analogue signal paths'.

4.5 S&S

S&S is a generic term for many of the methods of sound synthesis, which use variations on a sample playback oscillator as the raw sound source for a VCF/VCA synthesis modifier section. The samples are normally stored in ROM using pulse code modulation (PCM). This is just a technical term for the conversion of analogue values to digital form by converting each sample into a number, but the acronym has become widely used in manufacturer's advertising literature. The source sample playback is much the same as for a DCO driving a large wavetable, whilst the modifier sections are usually based on the VCF/VCA structure of analogue synthesizers.

Although the use of the term 'S&S' has been introduced for instruments where the modifiers are digital emulations of the VCF and VCA section of an analogue synthesizer, S&S is not necessarily restricted to digital instruments. It can also be produced with analogue equipment, and in fact, instruments such as the Mellotron, Chamberlin and Birotron could be considered to be S&S synthesizers, which use magnetic tape instead of solid-state memory. Many of the early wavetable instruments and samplers replayed digital samples and then processed them through analogue modifiers.

The availability of low-cost, high-capacity ROM is one of the major factors in the change from simple wavecycle DCOs to sample-replay instruments with hundreds of sampled sounds. In the same way, advances in digital technology have allowed a gradual changeover from analogue modifiers to digital emulations. So S&S synthesizers start out as hybrid instruments with a DCO driving a sample replay, processed by analogue filters, but end up as completely digital instruments. A typical S&S synthesizer of the early 2000s mixes many of the features of a sampler with an emulated modifier section which has the processing capability of an analogue synthesizer of the 1970s – complete with detailed emulations of resonant VCFs.

4.5.1 S&S samples

Unlike dynamic wavetable synthesizers, the samples that are provided with S&S instruments are normally replayed singly rather than being sequenced into an order. The only available source of the raw sound material for subsequent modification is thus a collection of preset sounds or timbres. S&S instruments then allow the processing of this raw sample 'source' of sound through one or more 'modifiers', and therefore allow different sounds to be synthesized. The modifiers are usually just some sort of filtering and enveloping control. The complexity of the processing varies a great deal – some have just low-pass filtering and simple envelopes, whilst others have complicated filtering that can be changed in real time, and loopable or programmable function generators instead of envelopes. In general, the most creative possibilities for making interesting sounds are provided by a combination of powerful sample replays options with elaborate processing functions.

Most S&S instruments have their sample sets held in ROM, which means that there is a fixed and limited set of available source sounds. Many GM and low-cost 'home' keyboards use S&S technology to produce their sounds – replaying the sounds is relatively straightforward and can provide high-quality sounds. In a typical GM instrument, a large proportion of the memory is taken up with a multi-sampled piano sound, and the rest is almost entirely devoted to other orchestral or band instruments. These instrument samples are chosen because they have the correct characteristics for the instruments that they are intended to sound like; if they do not sound correct, then they fail to sound convincing. Unfortunately, because these audio fingerprints are so effective at identifying a sound as being of a particular type, it is not easy to make any meaningful modifications to the sample – a violin sample still tends to sound like a violin, regardless of most changes to the envelope and the filtering. The sample sets in most S&S instruments thus represent a pre-prepared set of clichés: all readily identifiable, and all very difficult to disguise. Rather like the audio equivalent of a 'fingerprint' in fact (Figure 4.5.1).

This fingerprint analogy can also be extended to the modifiers of the source sounds as well. If the only filtering available is a low-pass filter, then there will be a characteristic change of harmonics as the filter frequency is changed, and this can be just as distinctive as a specific sample. Filters with alternative 'shapes' like high-pass, notch, band-pass and comb filters can help to give extra creative opportunities for sound-making. Again, the creative potential is reflected in the complexity of the available processing.

In order to avoid this fingerprinting problem, the programmer of an S&S synthesizer needs to have more control over the samples than just simple sample replay. Multi-sample wavetable hybrid instruments – like the Korg Wavestation – provide sample sequencing, cross-fading and wave-mixing facilities that enable samples to be manipulated in ways that can remove some of the more identifiable characteristics. Pitch shifting a violin residue and then using just part of it as the attack for another sound can produce some powerful synthesis capabilities in just the sound source.

There are many synthesizers and expander modules that use the S&S technique to produce sounds, and it has been very successful commercially for a number of reasons. It is comparatively easy to design an S&S instrument that incorporates sounds like the GM set, and it will have a broad range of applications, from professional through to home use. Because S&S instruments use

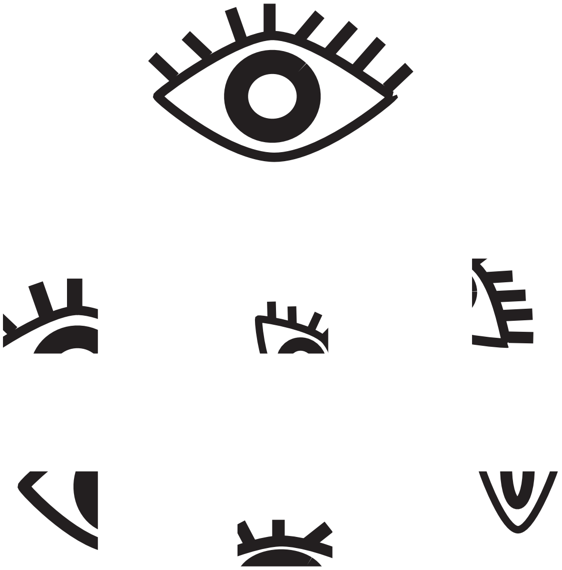


FIGURE 4.5.1
Sample 'fingerprints' are characteristic features of sounds that resist changes aimed at obscuring them. Just as you only need to see part of a well-known logo or symbol, so the distinctive elements of some sounds can be hard to hide.

pre-defined and fixed samples in ROM form, there is also considerable scope for selling add-ons like extra sample ROMs. Despite this, because many samplers also have the same sort of synthesizer processing and modification stages but their samples are held in RAM instead of ROM, the creative possibilities of a sampler are much wider!

4.5.2 Counters and memory

The basic process for reproducing a sample from memory involves using a digital counter to sequentially access each sample value in the sample memory. The first sample is pointed to by the counter, which then increments to point to the next value, and this repeats until the entire sample has been read out. In practice, these retrieved values may be used as the input to an interpolation process, but the counter and memory structure remains the foundation of the replay technique. Samples are normally organized serially throughout the memory device – the end of a sample is followed by the start of the next sample. Some manufacturers deliberately order their samples so that successive samples are related in their harmonic content, which allows the sample memory to be used as a form of dynamic wavetable. But this is often complicated by the provision of multi-samples where the same instrument is represented by samples taken at different pitches (Figure 4.5.2).

In order to hold several different samples in one block of memory, pointers to the individual samples are required. There are many approaches to providing these pointers to the locations or addresses of the sample values. The simplest method specifies the start and the stop addresses for each of the samples, where the start address can be used to pre-load the counter, and the stop address can be used to stop the counter when the end of the sample is reached. Alternatively, start and length parameters can be used when the counter merely adds an offset to the start address, since then the length parameter stops the counter when the count equals the length. By changing the length parameter, the playback time of the sample can be controlled.

If it is required to commence playback of the sample after the true start of the sample, then an offset parameter may be used to add an offset value to the start address that is loaded into the counter. Some instruments allow the length parameter to be set too longer than the sample, in which case the playback will continue into the following sample. Offsets can sometimes be used to provide similar control over the start address and can cause the replay to commence from a different sample altogether. E-mu's Proteus series is one example of S&S instruments, which implements start, offset and length parameters, as well as a partially ordered sample ROM to allow wavetable-like usage.

When offsets are applied to the start and the end of sample replay, then the sample values at those points may be numerically large, which can produce clicks in the audio signal, especially if the sample is looped or concatenated with another sample. Some S&S instruments only allow start and stop addresses to be selected when the sample values at those addresses are close to

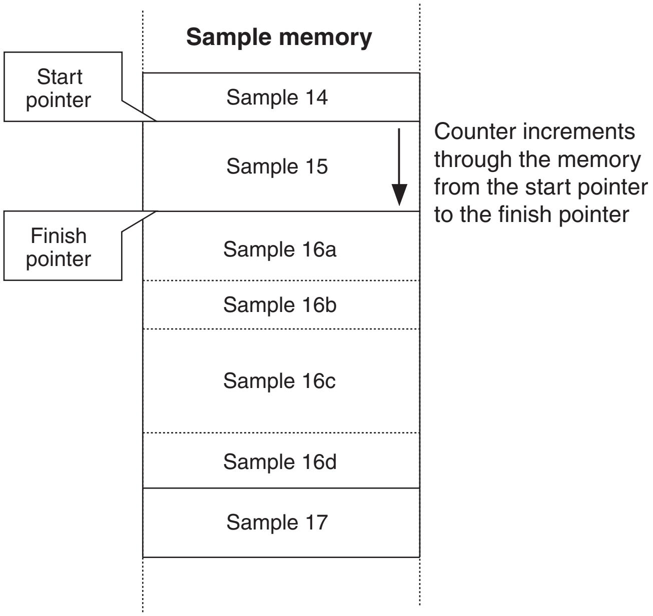


FIGURE 4.5.2 Sample memory is often arranged as a single contiguous block of ROM or RAM (or a mixture of the two). Sample replay consists of setting pointers to the beginning and end of the required sample, and then loading a counter with the start location and incrementing a count until the finish location is reached. Often it is possible to set the start and finish pointers to encompass several samples. In the example shown, only sample 15 will be replayed, but by moving the finish pointer then the multi-samples of sample 16 could also be included.

zero; although this is useful to prevent clicks in the output, it can be a problem when trying to loop the sample. Most samples normally start and finish with values that are close to zero (Figure 4.5.3).

Looping samples can involve either the entire sample between the start and the stop addresses or any portion in between. Some instruments allow loops to extend beyond a single sample – sometimes even through the whole of the sample memory. The obvious loop is to play the sample through to the end of the loop, then to return to the start of the loop, and then repeat the section between the start and end of the loop. Loops can be set to occur a number of times, or for a specified time period, or they may be controlled by the EG. Loops can be forwards, where the end of the loop is immediately followed by the start of the loop, or can alternately move forwards and backwards through the looped section of the sample. Alternate sample looping can help to prevent audible clicks when the sample values at the loop addresses are not at zero-crossing points. Another possibility is to invert the sample playback for each alternate repetition again so that clicks are minimized.

The predominant use of the loop is to provide a continuous sound when the EG is in the sustain portion of the envelope. These are called sustain loops. But

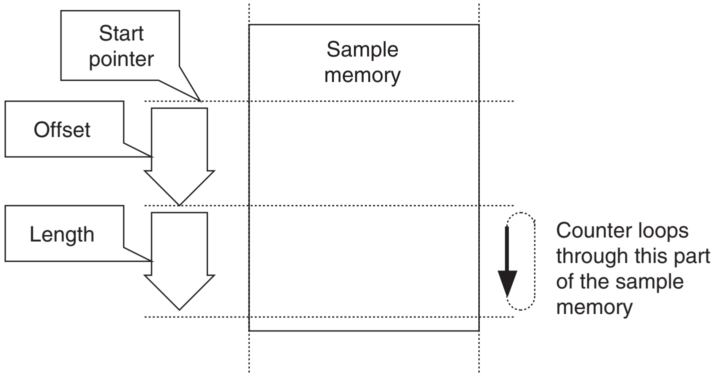


FIGURE 4.5.3 Sample-replay parameters provide additional control over how the counter starts and loops whilst replaying a sample. An offset parameter allows the start of the sample to be later (or earlier) in the sample memory, whilst a length parameter allows the offset or start to be changed dynamically without altering the replay time of the sample.

it is also possible to have attack or release loops, where the start or end of notes can be extended without requiring long samples. Loops are a way of minimizing the storage requirements for sounds that are required to have long attack, sustain or release envelopes. S&S techniques where each sample is closely connected to the EG, and therefore has separate attack, decay, sustain and release loops, are becoming rarer as the cost of memory reduces.

Storing the parameters required for playing back a sample requires two separate storage areas. A look-up table is required to map the samples to their addresses in the sample memory, and this can also contain details of the length of the sample, zero-crossing points for potential start and offset addresses, as well as default loop addresses. These default control parameters can often be replaced by values held as part of the complete definition of a sound.

4.5.3 Sample replay

Replaying a sample involves reading the individual sample values from a storage device, and then converting these numbers into an analogue signal. The conversion from digital to analogue is carried out by a DAC chip. There are two methods of replaying samples: variable frequency and fixed frequency.

Variable frequency playback

The easiest method of replaying a wavecycle or wavetable would be to output the sample values at a rate controlled by an oscillator or DCO. This is called variable frequency playback. The oscillator steps through the values that specify the waveform, and this is converted into an audio signal by a DAC.

Although simple to understand conceptually, this technique has several major limitations:

- The same number of sample values are replayed regardless of frequency. Because the oscillator is merely stepping through a fixed series of values, the detail contained within the waveform is constant, but the same is not true for the spectrum: at low pitches all of the harmonics may be below the half-sampling frequency, whilst for high pitches, only one or two harmonics may be below the half-sampling frequency. The sample should thus ideally have more detail when it is used to produce low-pitched sounds, because this is where the harmonic content is most important, whilst for higher pitched sounds less, detail is required because less harmonics will be heard. One technique that can be used to provide the required detail in samples is to use different sample rates for different pitches (Figure 4.5.4).
- The half-sampling frequency changes as the pitch changes. Because using a DCO to control the replay rate means that the half-sampling

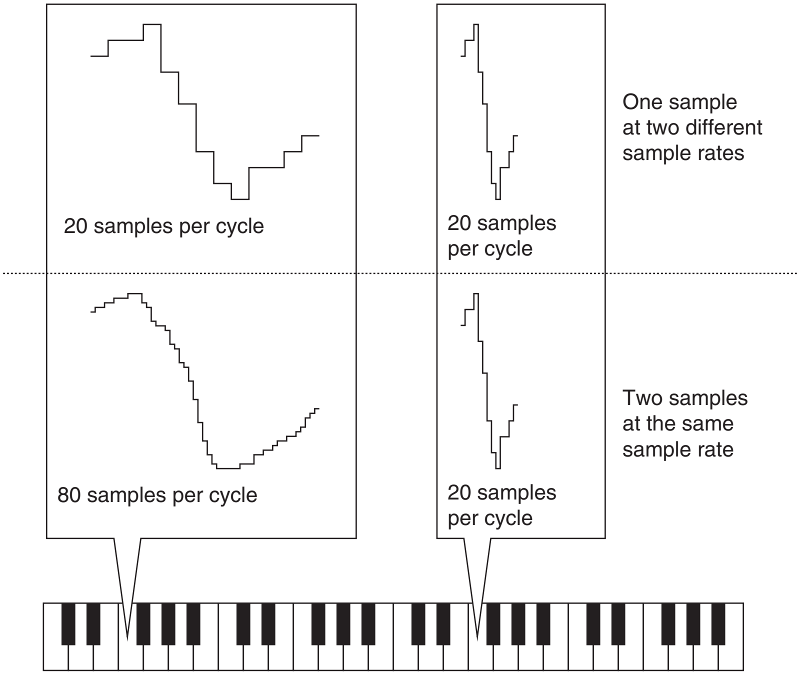


FIGURE 4.5.4 Multi-sampling is often used to provide several different samples mapped onto a keyboard, but it can also be used to provide different degrees of detail in a given sample. This diagram compares two methods of shifting down by two octaves in pitch: using one sample and slowing down the replay rate provide the same number of sample points regardless of the output pitch, whilst the use of two multi-samples enables the same sample rate to be used for each sample, thus increasing the amount of detail which is available for the lower pitched sample compared to the single sample method.

rate tracks the playback pitch, then the reconstruction filter also needs to track the half-sampling rate. (Note that early sample playback devices did not always do this. Instead they set the reconstruction filter so that it filtered correctly for the highest playback pitch, which meant that for lower pitches aliasing was present in the output signal.) Tracking means that a low-pass VCF is required to follow the changes in playback pitch, so that frequencies above the half-sampling rate are not heard in the output audio signal. Such VCFs have much more stringent design criteria than the VCFs found in analogue synthesizers: the 24 dB/octave roll-off slope of a typical analogue synthesizer VCF is not adequate for preventing aliasing, and slopes of 90 dB/octave or more are often required, with 90 dB or more of stop-band attenuation (Figure 4.5.5).

- The playback is monophonic. Because the sample-replay rate is set by an oscillator or DCO, variable frequency playback requires a separate sample-replay circuit for each individual pitch that is playing. Each DCO in a polyphonic synthesizer can produce a differently pitched monophonic audio sample, and these analogue outputs are then processed by analogue filters in the modifier section.

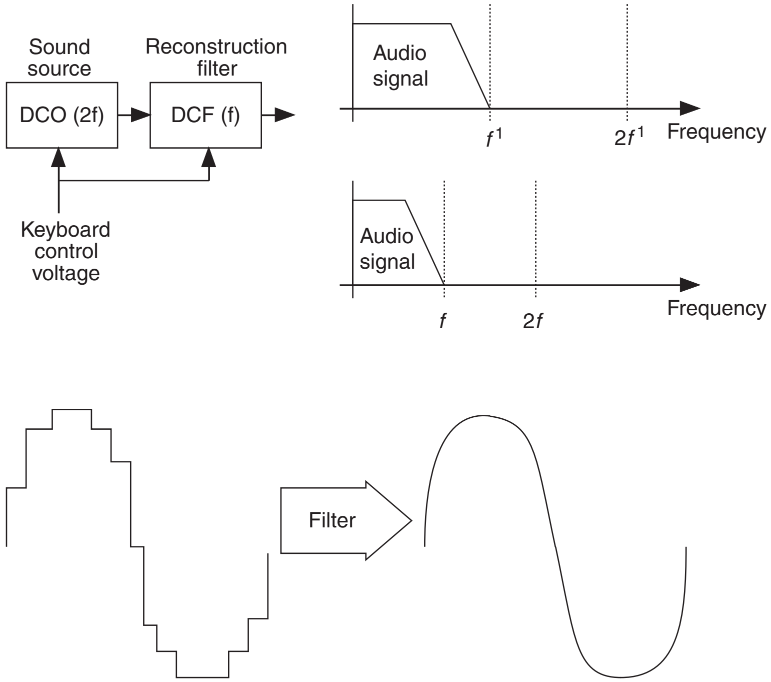


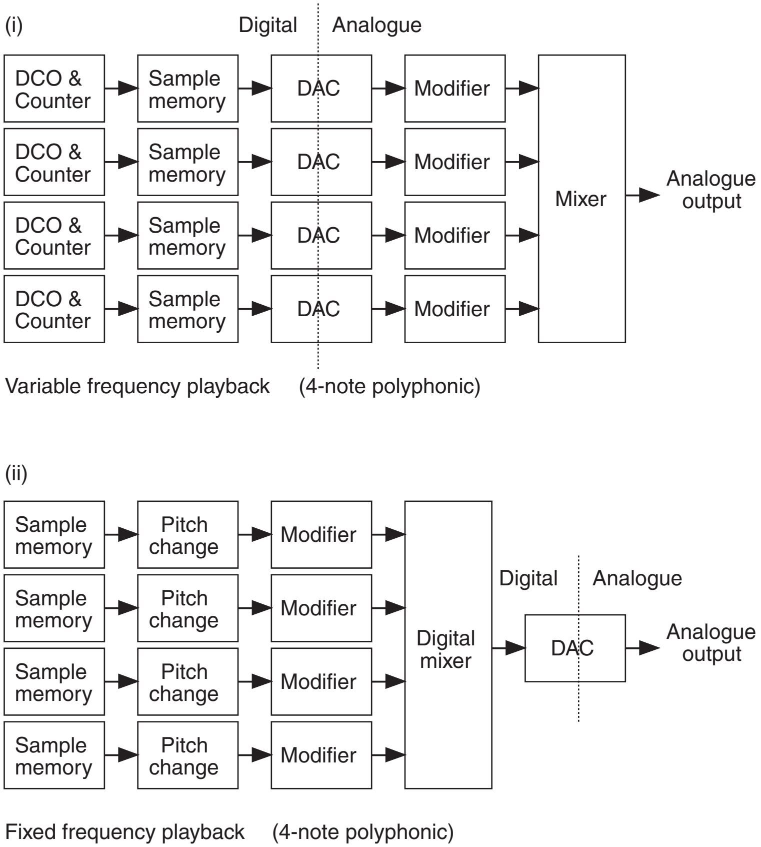
FIGURE 4.5.5 Reconstruction filters are normally thought of as being used in the output stage of digital audio systems, but they can be required to process the output of a DCO if it uses the variable frequency method to provide different pitches. In this example, a DCF is used to track the DCO frequency so that any aliasing components are removed before any post-DCO modifiers process the audio. The DCF 'smooths' the DCO waveform so that no aliasing components are present.

Fixed frequency playback

Fixed frequency playback uses just one frequency for the sample rate, but changes the effective number of sample values that are used to represent the different pitches by calculating the missing values. It has the advantage that only one sample rate is used, and therefore a fixed frequency sample playback circuit can be polyphonic and can be connected to a digital modifier section, which allows a completely digital synthesizer to be produced where the digital-to-analogue conversion happens after all the digital processing. Fixed frequency playback is now almost exclusively used in digital synthesizers and samplers (Figure 4.5.6).

4.5.4 Interpolation and pitch shifting

Changing the playback sample rate is not the only way of changing the playback pitch of a sampled sound. Consider a sample of a sound: a single cycle



Variable frequency playback (4-note polyphonic)

Fixed frequency playback (4-note polyphonic)

FIGURE 4.5.6 (i) Variable frequency playback (4-note polyphonic) requires a separate DCO and counter to access the sample memory, followed by a DAC to convert the sample into an analogue signal for processing by the modifier section. (ii) Fixed frequency playback (4-note polyphonic) changes the pitch of the sample and allows the use of a digital modifier section, with a single DAC to convert the output to analogue.

of a given pitch will contain a number of sample points, where the number is related to the cycle time for the waveform, and the sample rate. Therefore if 256 sample values represent a single cycle of a waveform at one pitch, then for a lower playback pitch more sample values would be required, whilst for a higher playback pitch less sample values are required. But it is possible to take the existing sample values and work out what the missing values are by a process called interpolation. This is used in fixed frequency sample playback.

Interpolation attempts to represent the waveform by a mathematical formula. If the sample values are thought of as points on a graph, then interpolation tries to join up those points. Once the points are joined up, then any sample values in between the available points can be calculated. The simplest method of interpolating merely joins the sample points with straight lines; this is strictly called linear interpolation, although it is often erroneously shortened to just interpolation. Although this is easy to do, real-world waveforms that consist of lots of straight lines joined together are rare! A better approach is to try and produce a curve that passes through the sample points (Figure 4.5.7).

One method that can achieve this uses polynomials: general-purpose algebraic equations that can be used to represent almost any curve shape. Polynomials are categorized by their degree, and in general, n points can be matched by an $(n–1)$th degree polynomial. Therefore for two sample values, a first-degree polynomial is, used which turns out to be the formula for a straight line.

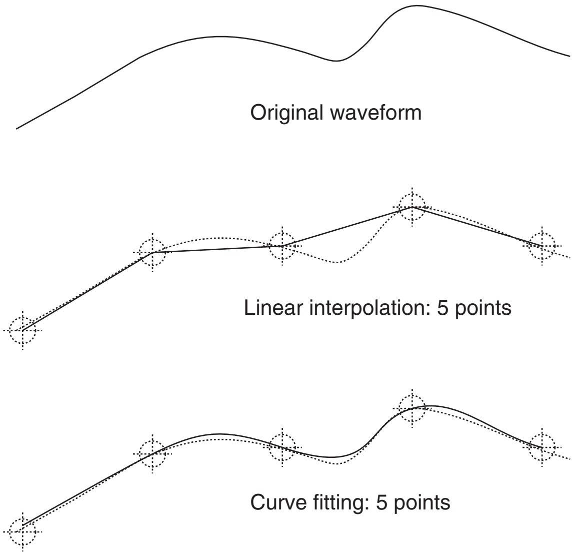


FIGURE 4.5.7 Interpolation is used to calculate missing or intermediate values in a sample. Linear interpolation draws straight lines between the sample points, whilst polynomial curve fitting attempts to match a curve to the sample points. In this example, a sample curve is shown, together with a linear interpolation based on 5 sample points, and a curve-fitted interpolation. The linear interpolation misses some of the major features, whilst the curve fitting produces a much better fit.

If more sample values are used to work out the shape of the curve, then higher order polynomials can be used: three points can be fitted by a quadratic (or second-degree) equation, whilst four points require a cubic equation. Manufacturers rarely reveal how they do their interpolation; in general, the lower the cost of implementation, the lower the degree of polynomial that is used, and the poorer the resultant audio quality.

Interpolation using polynomials with degrees higher than 1 is sometimes called differential interpolation to distinguish it from linear interpolation. It is also possible to design filters that can interpolate, and these are used in many digital systems.

An alternative technique that can reduce the number of sample values at high frequencies is literally to remove samples, or conversely to add in extra sample values at low frequencies. The simplest way to do this for an octave shift up or down is to either miss every other sample or repeat each sample. This is called decimation, and it is crude but effective. Since there are no calculations involved, it is easier to implement than interpolation, but it can produce distortion in the output. Because of the relatively low cost of ROM, sampling at a higher rate than is required can be used. Known as 'over-sampling', the idea is to provide more points for the interpolation processing. Since the points are closer together and are more available for the calculations, the interpolation quality improves. Over-sampling can be at any rate twice from twice the required rate to 64 times or more. The performance of the memory and the interpolation processing requirements limit the over-sampling rate.

4.5.5 Quality

Sample reproduction quality is determined by:

- sample rate in kilohertz (affects the bandwidth)
- sample size in bits (affects the signal-to-noise ratio (SNR))
- interpolation technique (linear or polynomial: affects the distortion when pitch transposing)
- the anti-aliasing and reconstruction filters (affect the distortion and SNR).

The CD sample rate of 44.1 kHz and the digital audio tape (DAT) sample rate of 48 kHz have become widely used in electronic musical instrument, with some instruments using even higher rates of 96 kHz or more. Samplers often have a range of available sampling frequencies, so that their memory usage can be maximized – sampling at 32 or 22.05 kHz can reduce the amount of storage that is required for sounds that have restricted bandwidths.

Sixteen-bit sample size has become the norm. Internal processing is often higher, but conversion chips designed for CD players (which are fundamentally based on 16-bit sample storage) are widely used in synthesizers and samplers. As higher resolution converters have become affordable, they have been incorporated, in some sample replay devices.

Interpolation techniques depend on the processing power that is available. Microprocessors and DSP chips continue to increase their performance, and therefore more sophisticated interpolation techniques will become possible, which should improve the quality of sample replay and transposition.

Analogue filter technology is almost at the theoretical limits, and therefore any improvements are likely to take place by adding in digital filtering. By increasing the sample rate inside the conversion chips, it is possible to augment the anti-aliasing and reconstruction filters with additional digital filtering using DSP chips. This allows enhanced performance, and yet outside the conversion chips, the samples can still be at a sample rate of 44.1 or 48 kHz.

Synthesizers and samplers will continue to follow development in audio technology. Future developments are likely to include more digital processing and less analogue electronics.

4.6 Topology

In general, the component parts of a hybrid synthesizer can be connected together in much the same way as an analogue synthesizer (see Section 3.6).

Because wavecycle, wavetable and S&S instruments have a 'pre-packaged' set of samples, they are sometimes described as merely sample-replay instruments, and not true synthesizers. But unlike many analogue synthesizers with a fixed signal path, hybrid instruments often have more flexibility in how the parts can be connected.

For the case of a single sample being replayed by an S&S instrument, the only changes that can be made to the sample are restricted to the modifier section, which allows changes to the filtering and envelope of the sound. But almost all S&S instruments provide rather more than this 'basic' mode of replay: normally either two independent sets of 'sound source and modifier' or two separate sound sources processed by a single modifier. In addition, some instruments also allow more than one sound to be triggered from the same note event, and therefore several samples can be combined (Figure 4.6.1). This variable topology, particularly the paralleling of complete source and modifier sections, allows a lot of control over two separate parts of the sound that is being produced.

It should be noted that polyphony is almost always traded against the complexity of the topology. Polyphony decreases as the number of sets of sound source and modifiers increases. For example, the polyphony would halve if the sound source and modifier resources required are doubled. Because of this, polyphony has tended to increase with time. A typical S&S synthesizer of the early 2000s may have 128-note polyphony or more, although the demands of typical sounds will reduce this to 32 or 16 notes.

The ability to trigger the playback of several different samples from one event opens up considerably more synthesis possibilities. Some early S&S instruments used an 'attack and sustain' model, where one sample was used

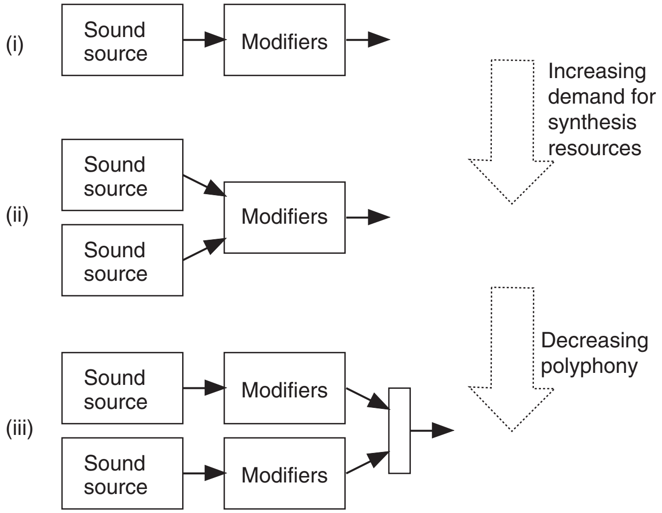


FIGURE 4.6.1 The basic S&S topology is (i) a single sound source followed by one modifier section. But most S&S synthesizers have (ii) either two sound sources which share a single modifier section or (iii) two separate sets of sound source and modifier.

to produce the attack portion of a sound, whilst a simplified 'subtractive synthesizer'-type section was used to produce the sustained portion, with a cross-fade between the two portions of the sound. As the technology developed, two sample-replay sound sources were used to produce the sound, and this allowed a more flexible division of their roles. Chapter 7 describes some of the ways in which two or more separate sound sources can be combined to produce composite sounds.

4.7 Implementations over time

Section 4.3 discussed the technology of DCOs and mentioned the difference between the design of instruments in the mid-1970s and those of the mid-1990s. This section summarizes the differences between hybrid synthesizers since the 1970s.

The 1970s

In the 1970s, hybrid instruments were just developing. VCOs were gradually being enhanced by the addition of auto-tune and digital control features, as well as programmability of the complete synthesizer with the change in emphasis from 'live' user programming to instant access through large numbers of memories. There were two distinct types of keyboard synthesizers: versatile monophonic or polyphonic synthesizers with rather more limited functionality – all based on mixtures of analogue and digital circuitry, and fully polyphonic 'electronic pianos'/'string machines' and multi-instruments (string and brass) – all based on top octave chips plus dividers followed by simple filtering and enveloping circuits. The second category was already in decline the polyphonic digital instruments of the 1980s would cause their complete disappearance.

The synthesizers had limited wavecycle waveforms, and if any controls were provided for the levels in the wavecycle, they would be on a 'one control per function' basis. The display would use light emitting diodes (LEDs), or perhaps a discharge tube/fluorescent display. Waveform samples would be in 8 bits, and the sample rate would be between 20 and 30 kHz, giving an upper limit for frequency output of between 10 and 15 kHz. Control would be via 4- or 8-bit microcontrollers, adapted from chips intended for simple industrial control applications. The interfacing would be via analogue control voltages, gates and trigger pulses, or perhaps from a proprietary digital bus format.

The 1980s

The release of the Yamaha all-digital FM synthesizers in the early 1980s saw all the other manufacturers trying to catch up and releasing hybrids whilst their development teams worked on the digital instruments that would begin to appear in the late 1980s.

These hybrids used digital enhancements to make most of the analogue oscillators, and eventually replaced the VCO completely with a digital equivalent. Portamento was the first casualty of this conversion, but by the end of the decade it had reappeared as the clock speed of chips made more sophisticated DCOs possible. Early designs used medium- and large-scale integrated circuits (ICs) containing tens, hundreds or thousands of digital gates. Wavecycle was joined by wavetable, usually with either 8- or 12-bit waveform samples.

The display gradually replaced the front panel knobs as the center of attention during the programming process, although a 2-row by 16-character liquid crystal distal (LCD) display (which might be backlit) was not ideal. Individual controls were replaced by 'parameter access', where a single slider or knob was used to change the value of a parameter that was selected by individual buttons. The 8- and 16-bit microprocessors were used to control the increasingly complicated functionality, especially once MIDI became established. Interfacing polarized rapidly from proprietary interface busses to MIDI within a couple of years of the launch of MIDI in 1983.

The 1990s

The 1990s opened with a preponderance of all-digital instruments and a consolidation of sampling. But this was quickly followed by a resurgence of interest in analogue technology, and some manufacturers began to rework older designs or even design completely new instrument from scratch. Although often labeled 'analogue', many of these instruments were actually hybrids; most often they use DCOs rather than VCOs. Even the 'pure' analogue instruments had considerable amounts of digital circuitry used for control and programming purposes.

DCOs used multiplexed circuitry to provide independent 'oscillators', and these used sophisticated accumulator/divider-type techniques to provide very fine resolution frequency control – typically on custom chips made for the individual manufacturer (ASICs). With plenty of processing power available, the wavecycle and wavetable generation techniques were combined by sampling, normally with 16-bit waveform samples and better than CD sampling rates (>44.1-kHz sample rate) and by wave sequencing. Displays increased in size, with 4-row by 40-character backlit LCDs (and larger) in common usage: some with dot-addressable graphics modes instead of just character-based displays. Allied to this was the increasing importance of a graphical user interface (GUI), sometimes with a mouse used as a pointing device, but almost certainly with soft-keys or assignable buttons. Computer-based editing software helped to make the front panel display almost superfluous on some rack-mounting instruments.

Control functions were provided by 16- or 32-bit microprocessors, perhaps with a DSP for handling the more complex signal processing functions. Interfacing was through MIDI. The conversion from analogue to digital was almost complete – often only the VCFs and enveloping was analogue.

The mid-1990s saw the release of all-digital instruments that replaced even the VCFs with digital 'software-based' equivalents, and the era of 'emulation' began. With software now capable of producing complex imitations of entire analogue instruments and even models of real instruments on DSPs, the mid-1990s hybrid designs were the last: software emulations priced analogue designs out of the market by the end of the decade.

The 2000s

The twenty-first century has seen the hybrid synthesizer more or less squeezed out of existence by the two opposing forces of analogue modeling and retro analogue. The emulation of analogue synthesis in mathematical models has become widely accepted, and specialist modern recreations of analogue synthesizers are now available for the wealthy 'retro analogue' purist. Some wavetable sound generation techniques have survived and are now incorporated into many all-digital S&S instruments.

Table 4.7.1 summarizes these points in a table format.

4.8 Hybrid mixers (automation)

Synthesizers were not the only audio electronic devices to have digital functions added to them during this period of hybridization. Mixers that had a number of variants of digital control were also produced.

The simplest were MIDI-controlled mixers, and these were typically line level submixers intended for use with synthesizers and other keyboard instruments. The Simmons SPM8:2 MIDI Mixer is one example that is often noted

Table 4.7.1 Comparisons

	Early designs			**Current designs**
	1970s	**1980s**	**1990s**	**2000s**
DCO	Digitally controlled VCOs, Top octave synthesizers	Master oscillators, rate adapters and dividers	Multiplex, accumulator/dividers	Multiplex, accumulator/ dividers
Technology	analogue/digital	MSI/LSI digital logic	ASICs & DSPs	Microprocessors & DSPs
Waveform	Wavecycle	Wavecycle, wavetable	Wavecycle, wavetable, sampling	Wavecycle, wavetable, sampling, modeling
Display	LEDs	16 × 2 LCD	dot-matrix LCD (4 × 40)	dot-matrix LCD (40 × 40)
Parameter Entry	Individual sliders	Slider and button selector	GUI, Softkeys	GUI, Softkeys, Softknobs, touch screen
Sample Bits	8	12	16	16–20
Control	4- and 8-bit microcontrollers	8- and16-bit microprocessors	16- and 32-bit microprocessors	16- and 32-bit microprocessors
Interfacing	Analogue: CVs, Gates	MIDI	MIDI	MIDI, mLAN, AES/SPDIF

as having audible zipper noise from poor MIDI-to-control-voltage conversion plus a difficult-to-use user interface.

Motorized faders enabled automation features to be added to analogue mixers. As with many mixtures of analogue and digital control, there is a basic physical problem with adding automation: how do you move the physical control to match the stored value? Rotary controls and linear faders require motors to do this, and the alternatives are awkward and time consuming – often the user moves the control until a flashing LED stops flashing.

Full store and recall of the positions of all the controls on an analogue mixer requires lots of additional circuitry, and this was much easier to achieve once mixers had gone either all digital or replaced the user interface with digital controls (see Section 5.17).

4.9 Sequencing

Because hybrid synthesizers have digital control, they tend to provide MIDI inputs and outputs, and therefore require either MIDI sequencer in either hardware or software form, or a CV/Gate hardware sequencer with a MIDI converter. Early MIDI hardware sequencers were often not very sophisticated but could be very expensive. CV/Gate and MIDI are not the only connections that might be required: Roland's digital communication bus (DCB) pre-MIDI digital interface is found on some hybrid instruments like Roland's Juno 106 (which has neither MIDI nor CV/Gate connections). MIDI also has a different sync system to DIN-Sync 24, which may require converter boxes. Roland's DCB and Oberheim's Parallel Bus Interface are just two contemporary additional proprietary interconnection methods that might be used.

Wiring

The transition from CV/Gate to MIDI means that 'hybrid era' wiring required a variety of different cables plus converter boxes and is arguably more complex than the CV/Gate connections of the 'pure analogue' era.

4.10 Recording

Hybrid synthesizers, particularly wavecycle and wavetable instruments, can produce lots of high-frequency content in their audio output. But tuning stability with temperature is better, and therefore less time needs to be spent on tuning. Hybrid synthesizers also tend to have some polyphony, and therefore polyphonic parts can be recorded.

Recording using MIDI has one hidden benefit that is not immediately apparent unless you have recorded using an analogue multi-track tape recorder. If you slow down the tape to either hear the detail of a track or to play a difficult part more easily, then the pitch changes. With MIDI, you can record and play at any speed, and the pitch stays the same. Partially because of this simple advantage, MIDI became widely adopted, and the scene was set for the late 1980s where MIDI arrangements would be prepared in a home recording studio using simple synthesis modules, and then be taken to a recording studio to be played back on synthesizers and samplers, plus have vocals and other instruments recorded using a multi-track tape recorder.

4.11 Performing

Hybrid synthesizers tend to be polyphonic rather than monophonic, and therefore they tend to be lower in a stack, perhaps replacing a string synthesizer, or even the electric piano or organ. Because of the wide range of possible sounds, hybrid synthesizers can also be used as solo instruments, thus reducing the need for a monosynth for lead lines.

Sounds

Hybrid instruments have a broad range of timbres, plus subtractive-style modifier sections in some cases. Thus they provide lots of flexibility in a single instrument and can replace several single-sound instruments such as string machines or electric pianos. Hybrid instruments also have memories, which means that changing from one sound to another can be rapid, and does not require the performer to make lots of changes to parameters in between songs.

4.12 Example instruments

Fairlight CMI Series I (1979)

The Fairlight CMI came from Australia and combined computer technology with sampling technology using voice cards that were a hybrid mix of analogue and digital technology on the earlier models. The first models offered plug-in 8-bit wavecycle and wavetable synthesis cards that had evolved into 16-bit sample-replay cards by the time that the Series III model came out in 1985. Additive synthesis, 'draw your own waveform', step-time rhythm programming and many other innovations made this a very popular instrument with those who could afford the high purchase price.

PPG Wave 2.2 and Waveterm (1982)

The PPG Wave 2.2 combines wavetable oscillators with analogue filtering and enveloping, whilst the Waveterm added sampling capability and sequencing facilities. The wavetable memory offered 1800 basic waveforms, whilst the samples were only 8 bits. Later models such as the Wave 2.3 and EVU were 12 bits.

Roland Juno-60 (1982)

The Roland Juno-60 (Figure 4.12.1) and its memory-less version the Juno-6 both had DCOs and provided low-cost polyphonic synthesis (albeit with no velocity sensing on the keyboard, arpeggios instead of portamento, Roland's proprietary DCB instead of MIDI, and only one DCO per voice).

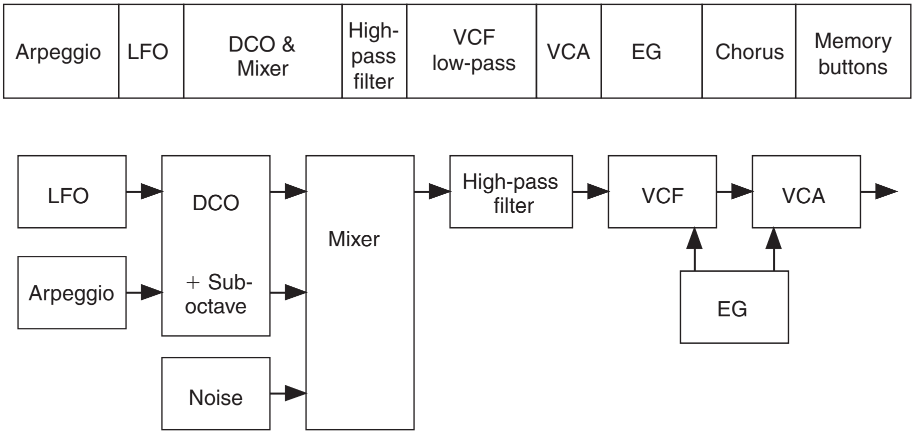


FIGURE 4.12.1
Juno-60.

Roland D50 (1987)

The Roland D50 was arguably the first commercial synthesizer to use S&S, although it uses the confusing term 'linear arithmetic (LA)' to describe the technique, and the implementation is only partial in comparison with later instruments. (The first full S&S implementation was probably the Korg M1, although it did not have resonant filters.) The D50 provides a combination of an analogue synthesizer emulation and a simplified S&S. The analogue synthesizer provides the classic synthesizer waveforms as the source material for a resonant filter and digital VCA modifier section. The sample replay is more primitive, with just a digital VCA and no filtering. The normal mode of operation is to use the sample part to provide the attack for a sound, whilst the sustained sound is provided by the synthesizer part. When it was released, this combination of sample realism and analogue familiarity proved to be a strong contender against the ubiquitous FM of the time (Figure 4.12.2).

The D50 was one of the first commercial polyphonic synthesizers to incorporate comprehensive built-in effects: EQ, chorus and reverb. It also marks the end of the front panel as a guide to the operation of the synthesis method and a change to mental models instead. The front panel clearly shows the influences at the time: diagrams influenced by FM synthesizers, joystick from vector synthesizers and a large soft-key-driven display to simplify the editing.

The D50 was the last of the 'first generation' of hybrids, although very little analogue is present. Instead, it was designed to appeal as an alternative to the

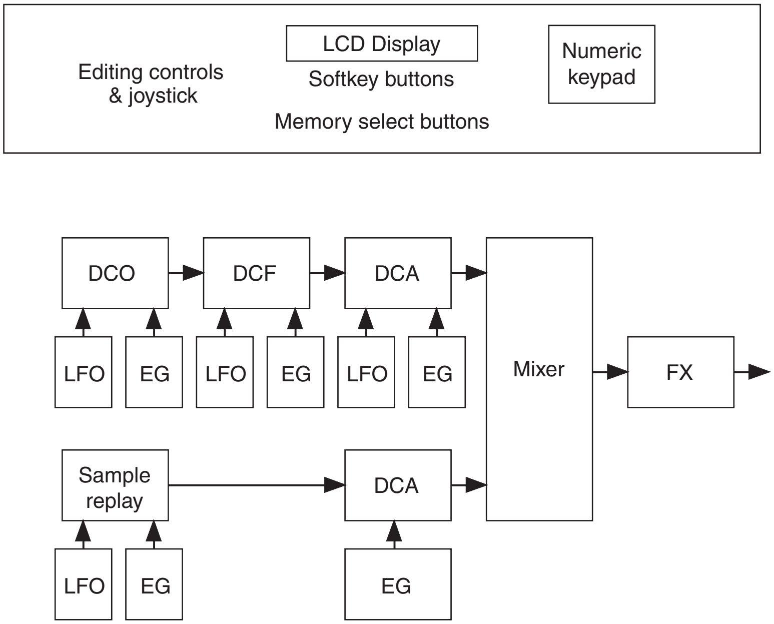


FIGURE 4.12.2 The Roland D50 mixes simple sample replay technology with a basic DCO/DCF analogue synthesis emulation.

all-digital FM synthesizers, whilst appearing as analogue as possible. But the Korg M1 changed the rules and ended the hybrids for a while.

Waldorf MicroWave (1989)

The Waldorf MicroWave is essentially a 'PPG Wave'-type of wavetable synthesizer, but redesigned to take advantage of the available electronics of the late 1980s. The minimalist front panel design relied on a large data entry wheel and a few buttons (Figure 4.12.3).

4.13 Questions

1. What is a hybrid synthesis technique?
2. What are the differences between cycles, multi-cycles and samples?
3. Give four examples of single-cycle waveshapes.
4. What is the difference between multi-cycle, wavecycle and wavetable synthesis?
5. How does a 'top octave' oscillator produce audio frequency outputs?
6. How does the frequency resolution of a DCO affect detuning and pitch bending?
7. Why is an early 1990s 'analogue' synthesizer really a hybrid?
8. How can dividers reduce the effect of jitter?
9. How does 'auto-tuning' work?
10. How are the contents of a wavetable converted into an audio signal?

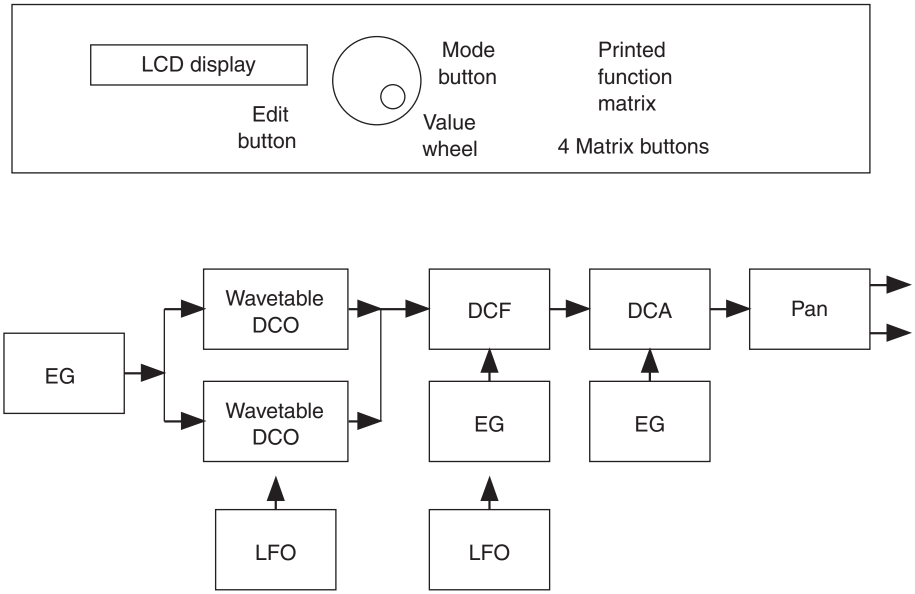


FIGURE 4.14.3
Waldorf MicroWave.

4.14 Timeline

Date	Name	Event	Notes
1969	Philips	Digital master oscillator and divider system.	
1970s	Ralph Deutsch	Digital generators followed by Tone-forming circuits.	The popularization of the electronic organ and piano.
1975	Moog	Polymoog was released.	More like a 'master oscillator and divider' organ with added monophonic synthesizer.
1982	PPG	Wave 2.2, polyphonic hybrid synthesizer, was launched.	German hybrid of digital wavetables with analogue filtering.
1986	Ensoniq	ESQ-1.	Digital sample replay synth with analogue modifiers (VCF, VCA).
1986	Sequential	Sequential launched the Prophet VS, a 'Vector' synth that used a joystick to mix sounds in real time.	One of the last Sequential products before the demise of the company.
1989	Waldorf	MicroWave, a digital/analogue hybrid based on wavetable synthesis.	Effectively a PPG Wave 2.3 brought up to date.
1996	Waldorf	Pulse.	The Waldorf Pulse was a three VCO, VCF analogue monosynth.
2006	Bob Moog	Little Phatty, a monophonic analogue synthesizer that is like a MiniMoog, revisited for the twenty-first century.	Has 'analogue signal path' and digital memories. A revisit to the OB1 type of synthesizer.

CHAPTER 5

Making Sounds with Digital Electronics

CONTENTS

Digital Synthesis

Digital Sampling

Environment

Digital synthesis of sound is the name given to any method that uses predominantly digital techniques for creating, manipulating and reproducing the sounds. Often, the only 'analogue' part of a 'digital' instrument will be the audio signal that is produced by the digital-to-analogue converter (DAC) chip at the output of the instrument.

Most digital synthesis techniques are based very strongly on mathematics: even methods like digital samples and synthesis (S&S), which often attempt to mimic, in software, the analogue filters found in subtractive synthesizers. The precision with which digital synthesizers operate has both good and bad aspects. Repeatability and consistency might seem to be a major advantage over the uncertainty, which often occurs in analogue synthesizers, but this precision can also be a disadvantage. For example, frequency modulation (FM) synthesis in an analogue synthesizer is difficult to control adequately because of the slight non-linearities of the FM inputs of many oscillators, whilst in a digital synthesizer, the precision of the calculations can mean that 'unwanted' effects like the cancellation of harmonics in a spectrum can happen. In an analogue synthesizer, the minor variations in tuning and phase would prevent this from happening; in a digital system, these may need to be artificially introduced.

This illustrates a very important point about digital sound synthesis. The degree of control that is possible is often seen as an advantage. But it also requires a considerable investment of time in order to be able to take advantage of the possibilities offered by the depth of detail, which may be required especially when there are potential problems if one does not fully understand the way that the synthesis works. This is very important in techniques like Fonctions d'Onde Formantique (FOF), where forgetting to set some of the phase parameters can result in major changes to the sound that is produced.

In summary:

- Analogue synthesizers offer the rapid and often intuitive production of sounds, but they have intrinsic non-linearities, distortions and inconsistencies, which can contribute to their characteristic 'sound'. If the speed of use and the available sounds are suitable, then the limitations may not matter.
- Digital synthesizers can provide a wider range of techniques, some of which are very powerful at the cost of complexity and difficulty of understanding. But they do not suffer from the built-in imperfections of analogue circuitry, and therefore these may need to be simulated, which adds to the task of controlling the synthesis and makes them less intuitive. The creative possibilities offered by digital synthesis are obtained at the expense of the detail required in setting up and controlling them.

Digital sounds

It has been said that digital sounds 'clean', whilst analogue sounds 'natural'. As a vague generalization, this is almost acceptable. It is possible to make very crisp, clear timbres using digital technology, but this is by no means the only tone color that is available. In fact, digital technology often introduces its own distinctive 'dirt', 'grunge' and 'distortion' into the signal.

Two of the commonest artifacts of using digits instead of analogue signals are quantization noise and aliasing.

- Quantization noise is the grainy, roughness that is typically found on the decay and release of pianos or reverbs. It happens because of the limited resolution of the numbers that digital systems use to represent audio signals, as the numbers get too small then errors get introduced and this appears as an extra noise.
- Aliasing is a side effect of the process of sampling; it is caused by a combination of imperfect filtering and 'just good enough' sampling rates. Aliasing sounds like ring modulation and is often heard as harmonically unrelated frequencies towards the top of the frequency spectrum.

Notice that both of these 'distortions' are due to imperfections in the way the digital works, and as such, are very similar to the limitations that are found in 'real' instruments or even 'analogue' synthesizers. Therefore the gross distortion that can be produced by overloading a filter in an analogue synthesizer is fundamentally no different to the aliasing in a digital sampler or to the 'wolf' tones that can be obtained by careful blowing into wind instruments.

The important thing is that applying descriptions like 'natural' or 'clean' to a sound is a very personal and subjective thing. Some very 'natural-sounding' flutes and harps can be entirely synthetic in origin, whilst some 'clean-sounding' clavinets might have high levels of distortion. Digital does not have any better claim on 'clean' sounds than any other method, nor does 'analogue' have a special reason to sound 'natural'. For example, there is no way that an

As digital audio transmission formats such as S/PDIF/AES/EBU and mLAN become more widely adopted as the outputs of synthesizers and the inputs of mixers, fully digital instruments may eventually appear where there is no DAC at all. Software synthesizers that are used in computers are already purely digital.

The main synthesis techniques covered in this chapter are: FM waveshaping, modeling, granular, FOF, and resynthesis.

analogue resonant filter-based 'Moog bass' sounds like anything in nature, because most real instruments do not have resonances that change in frequency quite to that extent!

5.1 FM

FM is an acronym for frequency modulation, an old technique that although possible on analogue synthesizers was not really practical for anything other than special effect sounds. Analogue synthesizer voltage-controlled oscillators (VCOs) are subject to frequency drift, variation with temperature, non-linearities, high-frequency mistuning and other effects, which lead to unrepeatable results when you try to use FM for generating melodic timbres. In fact, analogue-based FM is very good for producing a variety of 'non-analogue' sounding special effects: sirens, bells, metallic chimes, ceramic sparkles and more. It was not until the advent of digital technology that FM really became possible as a way of producing playable sounds rather than special effects.

FM essentially means taking the output of an oscillator and using it to control (modulate) the frequency of another oscillator. If you try this with two VCOs in a modular synthesizer then you are almost guaranteed to get some bell-like timbres at the output of the second, the 'modulated' VCO, especially if the only control input to the VCO is the exponential control input (FM should really use the linear control input – often marked FM!). In synthesizers, FM is used as a synonym for 'audio FM', where both the oscillator frequencies are approximately in the audio frequency range – 20 Hz to 20 kHz. FM radio uses an audio signal to modulate a very much higher frequency that is then used to carry the audio waveform as radio waves. This use of the word 'carrier' persists even in audio FM, where radio transmission has nothing to do with the sound that is produced.

FM synthesis is not really like any of the major synthesis techniques described so far, although it was briefly described in the context of a modulation method in Section 3.5. It is not a subtractive or an additive method, and it does not fit easily into the 'source and modifier' model either. FM has its roots in mathematics and is concerned with producing waveforms with complicated spectra from much simpler waveforms by a process that can be likened to multiplying. The simplest waveforms are sine waves, and FM is easy to understand if sine waves are used for the initial explanations. In fact, unlike analogue synthesizers, where the waveshapes are often the main focus of the user controls, FM is much more concerned with harmonics, partials and the spectrum of the sound (Figure 5.1.1).

5.1.1 Vibrato

Therefore, if an oscillator is set to produce a 1-kHz sine wave, and another oscillator is used to change the frequency with a 20-Hz sine wave, then the 1-kHz tone will have a vibrato effect. The oscillator producing the 1-kHz tone is

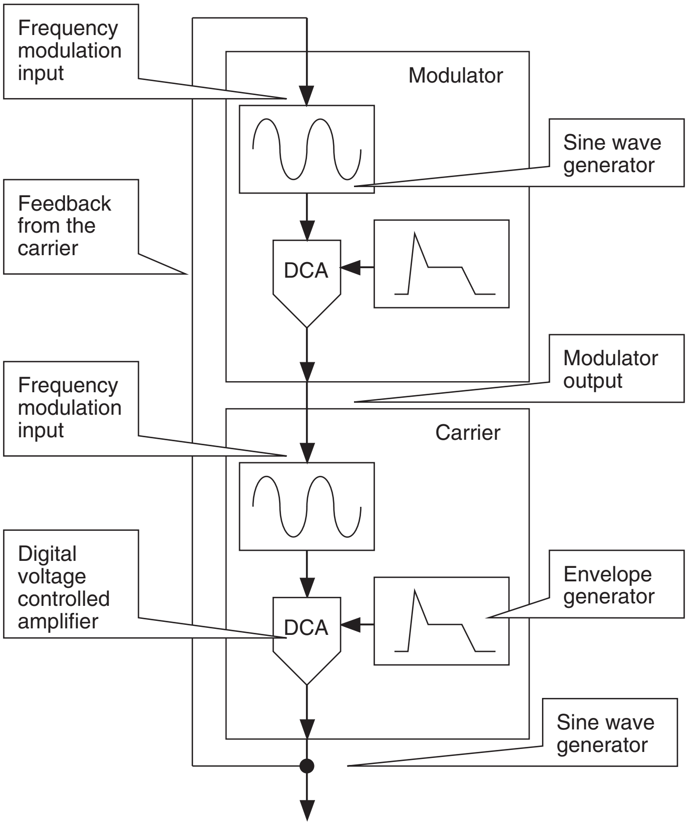

Frequency modulation input
Modulator
Sine wave generator
Feedback from the carrier
DCA
Modulator output
Frequency modulation input
Carrier
Digital voltage controlled amplifier
DCA
Envelope generator
Sine wave generator

FIGURE 5.1.1 The terminology of audio FM is different from analogue subtractive synthesizers, although many of the component parts are the same. In this example, the basic FM is produced by two identical 'modules' which can be thought of as consisting of a sine wave DCO, digital VCA and EG.

called the carrier, whilst the oscillator that is producing the modulation waveform is called the modulator. Although it is technically only correct to call the two oscillators the 'carrier' oscillator and the 'modulator' oscillator, it is more usual to call them the carrier and modulator for brevity.

If the modulator output is increased, then the depth of the vibrato will increase, which means that the carrier is sweeping through a wider range of frequencies. If the modulator output is decreased, then the carrier will be swept through a smaller range of frequencies around the original 1-kHz unmodulated frequency. The difference between the highest and the lowest frequency which the carrier reaches is called the deviation and therefore, an unmodulated carrier has no frequency deviation.

If the speed of the vibrato is increased above 30 Hz, it will stop sounding like vibrato, and if increased above 60 Hz, it will be perceived as several sine wave tones of different frequencies all mixed together, which is the 'characteristic 'bell-like', clangorous timbre often associated with FM.

5.1.2 Audio FM

Audio FM replaces the low-frequency modulator with another audio frequency. Suppose that the modulator level is initially zero. The output of the carrier oscillator will thus be a sine wave. As the level of the modulator is increased, the sine wave will gradually change shape as extra partials appear. Initially, two sidebands (partial frequencies on either side of the carrier) appear, but as the depth of modulation increases, so does the modulation index, and thus more partials will appear. The timbre becomes brighter as more partials appear, although unlike opening up a low-pass filter, partials appear at higher and lower frequencies. The lower frequencies can alter the perception of what the pitch of the sound is – if a 1-kHz sine wave acquires an additional sine wave (it is a partial, not a harmonic, because it need not be related to the fundamental by an integer frequency ratio) at 500 Hz, then it can sound like a 500-Hz tone with a partial at 1 kHz.

As described in Chapter 2, the output consists of the carrier frequency and the sidebands made from the sum and difference frequencies of the carrier and the multiples of the modulator frequency. The number of sidebands depends on the modulation index, and a rough approximation is that there are two more than the modulation index. The modulating frequency is not directly present in the output. The amplitudes of the sideband frequencies are determined by a set of functions called Bessel functions (Chowning and Bristow, 1986 is unfortunately now out of print). So, as the level of the modulator signal increases, the output gradually changes into a much more complex timbre. The transition is a smooth addition of frequencies much as you would expect with a low-pass filter gradually opening up, but with the added complication of extra frequencies appearing at lower frequencies too.

5.1.3 Bessel functions

Mention of Bessel functions normally means that mathematics takes over and the next few pages should be filled with formulas. FM is often presented as being inaccessible because of its complexity, so here I will attempt to try and describe how FM works in as simple and non-mathematical a way as possible.

We will start by taking the filter analogy a little further. Imagine an additive synthesizer (Section 2.3) that has individual envelopes for each of a number of frequencies. If we want to simulate a low-pass filter opening up, then we need some envelopes which allow first the low frequencies to appear, then the middle frequencies and finally the higher frequencies. The envelopes would look like a series of delayed attack and sustain segments, where the delay in the start of the attack was related to the frequency that was being controlled – the higher the frequency, the longer the delay time. Triggering the envelopes would cause the lower frequencies to appear, then the middle and finally the high frequencies. A similar set of envelopes could be used to produce frequencies that were lower than the fundamental (Figure 5.1.2).

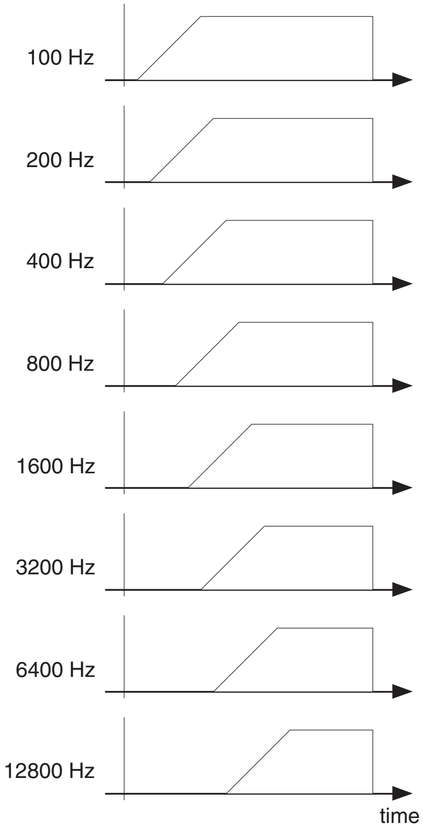


FIGURE 5.1.2 These envelopes can produce an output equivalent to a filter frequency being swept upwards. Each envelope processes one frequency component.

The shapes of these envelopes control the harmonic/partial structure of the sound produced by the synthesizer. By changing the shapes of the envelopes, we can change the way that the frequencies will be added as time passes. Actually we do not need to use envelopes; we can use any controller that can map one input to lots of outputs whose behavior can be controlled. It just happens that an envelope is one way of using time as the controller. If we used a control voltage (CV) and lots of voltage modifiers, it would be possible to control the frequencies from the additive synthesizer in just the same way, and the same envelope shapes could be used to describe what would happen; the only difference would be that the envelopes are now curves that show how the frequencies change with the input voltage instead of with time (Figure 5.1.3).

Bessel functions are the name given to the curves that relate how the frequencies are controlled in FM. Although they are smooth curves instead of the angular envelopes, the principle is exactly the same. In much the same way as the filter envelopes have time delays built into them, the Bessel functions vary

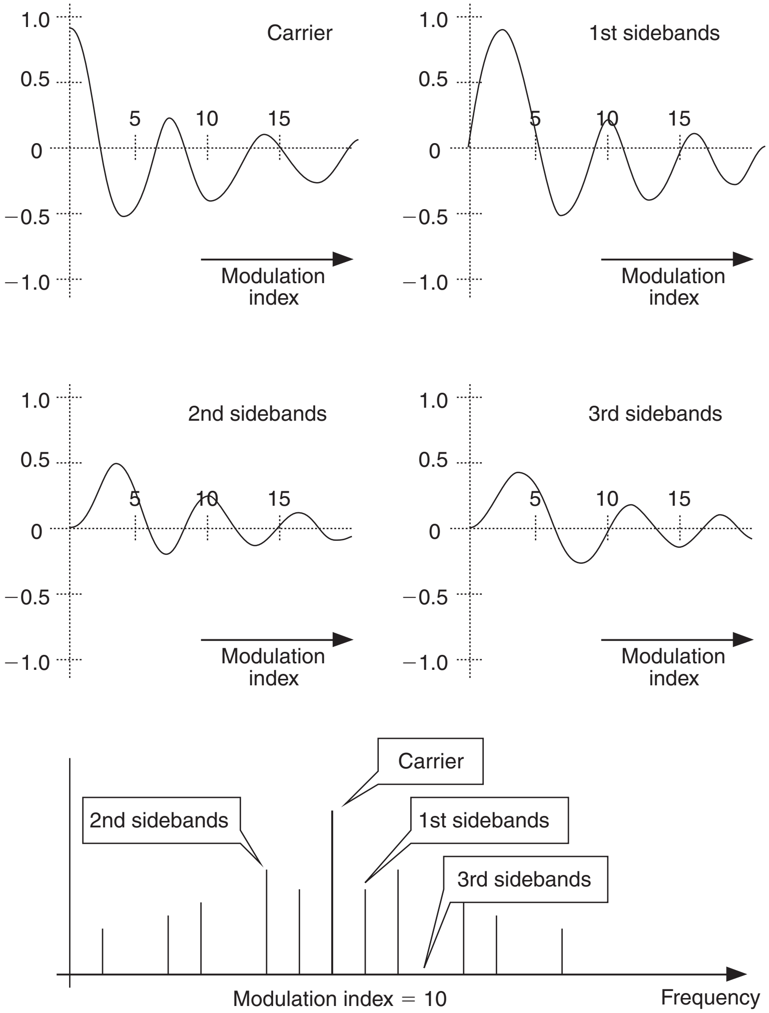


FIGURE 5.1.3 Bessel functions describe how the amplitude of the sidebands in FM vary with the modulation index. In this example, an FM output is produced using a modulation index of 10. Only the first four Bessel functions are shown.

in a similar way – the further away from the carrier frequency, the higher the value of the control needs to be to have an effect. Instead of time or a CV, the control is the modulation index. Therefore, as the modulation index increases (when the modulator level increases or the modulator frequency decreases), the number of partials increases. And that is really all there is to Bessel functions – they merely describe how the level of the partials changes with the modulation index. There are one or two complications in reality, and you should look at the references if you need more details.

The only other thing that needs to be considered for FM is how the frequencies are controlled. In FM, the spacing between the partials is related to the carrier and the modulator frequencies, but there are really only three basic relationships:

- *Integer*: Integer relationships between the carrier and the modulator frequencies produce timbres that have harmonic structures that are similar to those of the square, sawtooth and pulse waveforms – harmonics at multiples of the fundamental. The only complication is that the fundamental is not always at the carrier frequency because of the extra frequencies that can appear below the carrier.
- *Slightly detuned from integer*: Slightly detuned carrier and modulator frequencies produce the same sort of 'multiples of the carrier' harmonic structures, but with all the harmonics detuned from each other too. This can produce complex beating effects, although the amount of detuning needs to be carefully controlled to avoid too rapid beating effects.
- *Non-integer*: Non-integer relationships between the carrier and the modulator frequencies produce the bell-like, clangorous timbres for which FM is famous. If either the carrier or the modulator is fixed in frequency (i.e., it does not track the keyboard pitch), then the relationship will change with the pitch of the note being produced.

In all of these cases, the basic timbre produced is set by the relationship between the carrier and the modulator frequencies, whilst the number of harmonics and partials that are produced is controlled by the modulation index, using the values in the Bessel functions. There are some additional complexities caused when the modulation index is so large that the spreading out of the partials causes some frequencies to go below the zero-frequency point and get 'reflected' back, which can cause some additional cancellation effects.

That is really all there is to FM – you choose the timbre and then control it: usually dynamically. FM may be different from subtractive or additive synthesis, but the controls and the way that they work are relatively straightforward, once you understand what is happening. Almost all of the other functions, like low-frequency oscillators (LFOs), portamento, envelope shapes and effects, should be very similar to the same functions in other synthesizers.

Yamaha's TX81z was their first FM synthesizer to provide non-sine waveforms. These are in some ways equivalent to extra operators, because a single pair of non-sine wave operators can produce sounds that are like those from three or more sine wave operators.

FM is normally produced using oscillators that are made available as general-purpose building blocks called operators. These consist of an oscillator, an envelope generator (EG) and a voltage-controlled amplifier (VCA), all in digital form of course. The oscillator is a variable speed playback of a wavetable, whilst the VCA is a multiplier connected to the EG. In the first audio FM implementations, the wavetable held a sine wave, but later versions of FM had additional waveforms.

On a larger scale, FM may use more than one pair of operators (carrier and modulator), several modulators onto one carrier or even a stack of operators all modulating each other. It is also possible to take the output of a carrier operator and feed it back to the input of a modulator, which can be used to produce noise-like timbres, although it is still the modulation index and the frequency relationships that determine much of the timbre. Learning how to make the most of FM involves analysing the FM sounds produced by other people and programming sounds yourself, but the model described here should provide the basic concept of how FM works.

FM is good for producing sounds with complicated time evolutions and detailed harmonic/partial structures, but it can be difficult to program; the explanation aforementioned has been simplified, and there can be quite a lot of parameters to cope with. It is also possible to produce FM with non-sine wave oscillators, although all that happens is that each sine wave, which is present in the waveform, acts as its own FM system, and therefore you get lots of FM happening in parallel, which can lead to very noise-like timbres because of the large numbers of frequencies that are produced. FM is especially suited for 'metallic' sounds such as guitars, electric pianos and harpsichords (Figure 5.1.4).

FM really only requires the following parameters to define a timbre:

- carrier frequency
- modulator frequency
- modulator level.

But most FM sounds change the modulator level dynamically by using the modulator envelope, and the carrier also has an envelope, but even then the number of parameters required to specify a given sound is less than 20 parameters. In comparison to subtractive or additive synthesis, this is a much smaller number of parameters to deal with. For this reason, FM has been investigated as the synthesis part of an analysis–synthesis resynthesis system, but there are problems in extracting the FM parameters from sounds. In particular, it is not easy to take a specified waveshape or spectrum and calculate the required FM parameters, especially if there are any partials – non-harmonic frequency components (see Section 5.7 for more information on resynthesis).

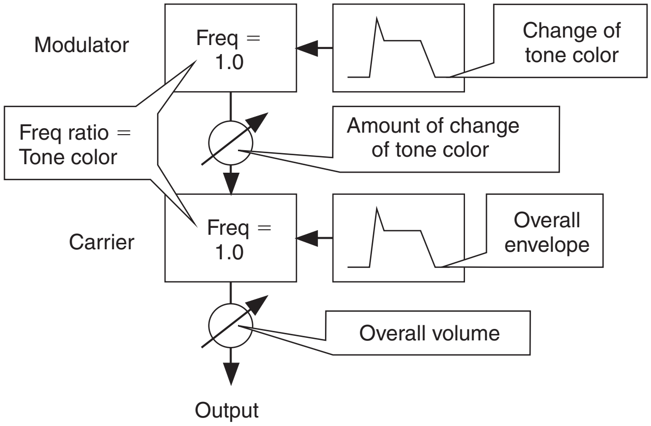


FIGURE 5.1.4
This overview of a simple FM synthesis system shows how the individual component parts contribute to the final sound output.

Filters did not appear until comparatively late in commercial audio FM history. The SY77 in 1990 had twin 12-dB/octave digital high- and low-pass resonant filters, whilst the DX200 in 2001 had a modeled voltage-controlled filter (VCF) that had high-pass, low-pass, band-pass and notch modes with up to 24-dB/octave cut-off slopes.

Having a small number of important parameters also enables FM to be a very powerful synthesis method when using real-time control. By changing the carrier or modulator frequency or the modulator depth with specialized musical instrument digital interface (MIDI) commands (or front panel controls), FM can be used to produce sounds that can change rapidly and radically. On an analogue subtractive synthesizer, the only comparable parameter change is the filter cut-off, and this is much more restricted in the timbral changes that it can produce.

5.1.4 Realization

The actual details of the way that FM is realized differ on different platforms. Computer-based software will probably use an approach different from the hardware-oriented custom digital signal processing (DSP) solutions used in synthesizers. But the basic elements are much the same in all cases, although the terminology may be very different. The descriptions that follow are mostly based on Yamaha's FM, mainly because it has been the most widely accepted and the most commercially successful of any of the digital FM implementations.

Oscillators

Initially, FM was produced using only sine waves. The mathematics behind FM are easy to understand if sine waves are used, and early FM work was at an academic level, where both the understanding of the sound production process and the esthetics of the resulting sounds are important. The first commercial FM synthesizers also used just sine waves, probably for reasons of cost: the Yamaha DX7 was introduced at a time when digital consumer electronics was virtually unknown – the compact disk (CD) player was not introduced for another year. The high ratio between the features and price of the DX7 was partially due to the use of digital technology but also due to a careful minimization of functionality; after all, Yamaha were testing the market with a very different type of synthesizer. The lack of front panel control knobs shows that they were prepared to take radical design decisions in both the synthesis and the user interface areas.

Implementing a digital FM sine wave generator has been covered in Section 3.3 on digitally controlled oscillators (DCOs). With only one waveform, the size of read-only memory (ROM) required can be quite small, especially if the symmetry of the sine wave is used to reduce the storage requirement – you can produce a complete sine wave cycle from just one quarter of a cycle of sine wave waveform points. The Yamaha design multiplexes the oscillator – it is used to provide the waveforms for all of the oscillators, with storage of the successive outputs used to give the equivalent of four or six separate oscillators. Further multiplexing is used to provide the DX7's 16-note polyphony, which was about twice the normally expected polyphony of polyphonic synthesizers in 1983.

Later, more advanced FM synthesizers like the TX81z used more complex waveforms. But this was often only achieved by deriving additional waveforms from the same sine wave ROM memory, by changing the way that the quarter cycle is reassembled to form a waveshape. Quite minor changes to the waveshape can have significant effects on the spectrum, and using waveforms that contain additional frequencies can produce FM sounds that are very rich in harmonic and partial content – even to the extreme of becoming noise-like. The second generation of commercial FM synthesizers from Yamaha (the SY77 and the SY99) also added the ability to use samples as part of the FM synthesis, but this did not prove to be very popular with users, and subsequent models in the 'SY' series concentrated on sample-replay technology rather than FM. Since then, Yamaha have released only two further devices that use FM synthesis: a rack-mount expander module, the FS1R, and a desktop synthesizer plus step sequencer, the DX200.

Most FM oscillators can be used in two modes. The usual mode is to allow the oscillator to track the keyboard pitch, although this need not be the normal keyboard scaling. The second mode is usually called 'fixed frequency', and here, the oscillator frequency does not change. Fixed oscillators can be used in several ways. At low frequencies, they can be used as carriers or modulators to produce vibrato-type cyclic timbral change effects, whilst at higher frequencies, they can be used to partially emulate a very resonant system. Fixed frequencies of a few hundred hertz are often used to produce vocal sounds, since the resulting sound has many of the qualities of the formants that determine human vocal sounds. A fixed oscillator within an FM algorithm produces an output spectrum with frequency components that are related to the fixed oscillator frequency or harmonics of it, and this can sound somewhat like a resonant tube. This was exploited and extended in the FS1R in 1998.

In each of these cases – sine wave, sine-derived waveforms and samples – the technology used to produce the FM waveform is very similar to the advanced DCO designs described in Chapter 3. The output of a period of time (not necessarily a single cycle) for each oscillator is stored and then used as the basis for the modulation of the next oscillator, and this iterative process is then repeated until it ends with the carrier oscillator. The high precision of the sine wave, the frequency resolution and the linearity of the FM all enable FM synthesis to be achieved in a precise, repeatable and controllable way – a big contrast to producing FM using analogue technology.

Envelopes and VCA

FM required digital control over the amplitude of the oscillator outputs, and for this Yamaha used the multi-segment rate/level type of envelope. Rate/level envelopes provide comprehensive control over the shape of an envelope by using function generator controls to set the characteristics of each segment. As the name implies, two parameters are used to control each segment: a rate and a level. The rate specifies how long the segment lasts, whilst the level sets the final level that the segment reaches. The initial level is normally the same as the final level, although some later instruments do not have this restriction.

Yamaha had identified that conventional attack decay sustain release (ADSR)-type envelopes were not suitable for envelopes that had complex attack stages – especially where the start of the sound did not rise at a constant rate or where the decay was complex. The multi-segment envelopes that they used in the DX7 had three segments to cover the 'key on' part of the envelope, plus one segment for the 'key off' or 'release' portion of the envelope. Because the final level of the third segment is held whilst the key is held down, it effectively produces a separate 'sustain' segment, which is the fixed level where the attack segments end at. This produces a categorization of the segments according to their function within the envelope. The first three segments are used to control the 'attack' portion of the envelope, the level of the last attack segment sets the sustain level, whilst the final segment controls the release behavior (Figure 5.1.5).

The envelope used in DX-series Yamaha six-operator FM synthesizers is a five-segment rate/level. There are three 'attack' segments, one 'sustain' segment

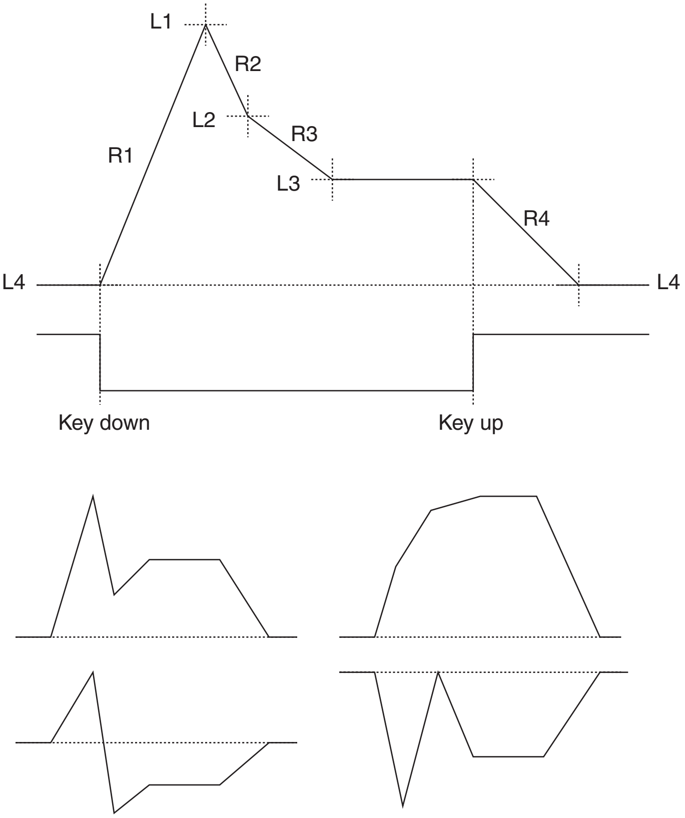


FIGURE 5.1.5 The EGs used by Yamaha in their FM synthesizers provide great flexibility because of their structure. The four levels can be anywhere in the permitted range, which allows a wide variety of envelope shapes, including inverted envelopes and pseudo-exponential attack segments.

and one 'release' segment. EG forced damping is found in the Mark II DX7 and forces the envelope to restart when a note is reassigned because of note stealing at the limits of the polyphony.

SY-series Yamaha six-operator FM synthesizers use eight-segment envelopes that add a separate initial level, delay time, two release segments and the ability to loop the envelope whilst the key is held down. Also, the final level is not necessarily the same as the first level. The two release segments enable additional control over the end of the note, whilst the delay time is used for special effects like arpeggiated operators or allocating operators to specific parts of the final sound – using one set of operators to generate the initial portion of a sound, whilst delayed envelopes produce the remainder of the sound. This effectively increases the apparent number of envelope segments at the expense of using operators for only part of the duration of the sound. The looping enables the sustain segment to be less static; simpler envelopes just reach a sustain level and stay there as long as the key is held down (Figure 5.1.6).

Low-cost FM implementations with four operators have simplified ADBDRR envelopes that had only six controls: five rates and two levels (breakpoints for the decay and release segments).

Not all multi-segment EGs use the word 'rate'. Time and slope are sometimes used as synonyms. There is also no standardization on how the parameters relate to the duration of the segment; some manufacturers use small numbers to mean short times, whilst others use the converse.

The digital VCA in FM is almost always treated as part of the EG.

Operators

The combination of an oscillator, envelope and VCA is such a fundamental building block of FM synthesis that it is often treated as a single module.

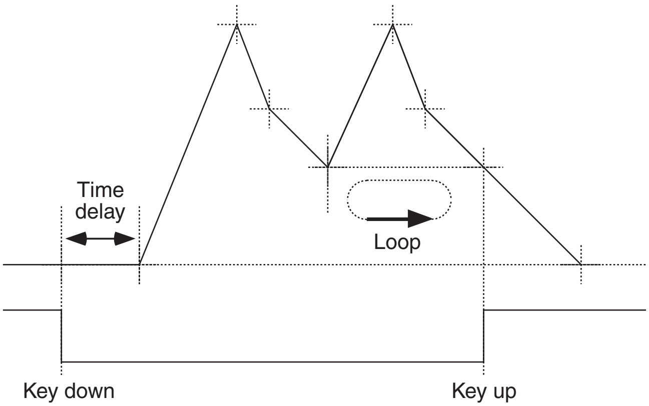


FIGURE 5.1.6 Loopable envelopes allow previous segments to be continuously looped whilst the envelope is in the sustain segment. In this example, the first three segments are looped. The use of delayed attack segments enables the production of echo-like and arpeggio effects.

Because the first major commercial success of utilising FM in a digital synthesizer was from Yamaha in the early 1980s, the terminology that they used has become widely adopted. Yamaha used the word 'operator' for the block formed by an oscillator, envelope and digital VCA (Figure 5.1.7).

The initial FM prototype instruments were the Yamaha GS1 with eight operators, and the GS2 with four operators, whilst the initial DX synthesizers had six operators. (The DX9 had two operators deliberately disabled in the software to give it reduced functionality.) Lower cost FM implementations followed using four operators with restricted frequency control and limited internal calculation precision and the chips for these were made available to other manufacturers – these became a 'de facto' standard for the basic implementation of a personal computer (PC) sound card. Prototype FM synthesizers like the V80 were produced by Yamaha with eight operators, although these never progressed beyond the development laboratory. Some of the Yamaha HX-series organs were released with eight operators, but these did not have the depth of user-programmability of the synthesizer products. Some FM implementations use multiple 'pairs' of operators, which do not provide the same flexibility as being able to arbitrarily connect more than six operators together.

In 1998, Yamaha released the FS1R rack-mount expander module, which had eight operators with built-in band-pass filtered 'formant' noise generators in place of the simple feedback or noise generators of previous implementations. The FS1R harked back to previous products based on speech synthesis concepts like the SFG-05 FM plug-in module for the CX-5M MSX computer, which had Japanese speech synthesis software. In the FS1R, the combination of the two types of operators, voiced and unvoiced, reflects speech synthesis

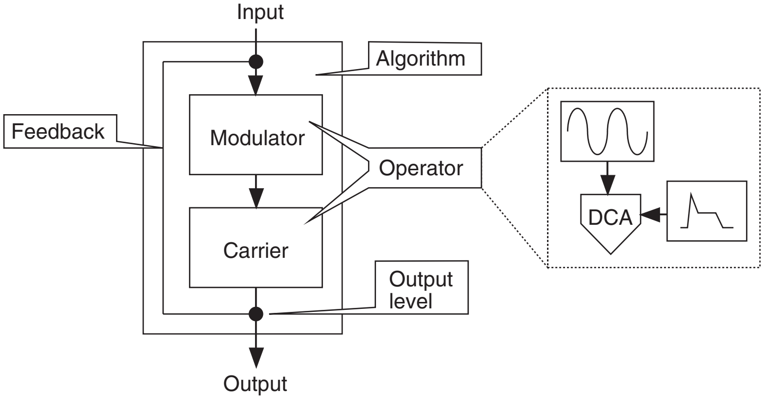


FIGURE 5.1.7 A Yamaha FM operator typically consists of a sine wave DCO, digital VCA and EG. Operators can be connected together in arrangements called algorithms, where the description of the operator as a carrier or a modulator is determined only by its position in the algorithm.

terminology. The 'voiced' operators were standard FM operators with pitched or fixed frequency modes: vowels in terms of speech, whilst the 'unvoiced' operators provide the 'f' and 's' sounds, and combinations of the two can produce consonants like 'b' and 't'. Of course, the FS1R could not only use these facilities for speaking, but also be used to produce singing (note the release of the 'Vocaloid' software a few years later) and instrumental and percussive sounds too. As with other implementations of FM, the FS1R requires an understanding of the principles behind the design in order to make the most use of the available facilities, and programming requires skill and time.

In 2001, Yamaha released the DX200, a tabletop synthesizer with a built-in 16-step sequencer. The DX200's FM had six operators and is DX7 voice compatible. But the DX200 introduced a new feature derived from the AN1X modeled analogue synthesizer – interpolation between two sounds, by using a front panel control, the parameters defining one sound can be smoothly changed to the parameters for another sound. The result is not always perfect: it can sound like a 'morph' between the two instrumental sounds or a blend from an instrumental sound to noise accompanied by metallic clanging and then to another instrumental sound. The morphing can be very effective, whilst the blend can be useful for adding just an edge to a sound by only moving slightly towards the noisy, metallic sound. The DX200 also attempts to provide an alternative user interface to FM, taking concepts from analogue synthesizers and the Korg DS-8 and 707, to give a set of front panel controls that are intended to provide live 'interactive' control over the sounds. For some algorithms, this approach is very effective.

Algorithms

Yamaha use the word 'algorithm' for the arrangement and interconnection of operators. Although there are many ways of arranging the topology of four or six operators, there are only a few important types:

- additive
- pairs
- stacks
- multiple carriers
- multiple modulators
- feedback
- combinations.

Additive

Although not actually FM, parallel operators can be used as a simple additive synthesizer producing several frequencies simultaneously. Unlike many additive synthesizers, the frequencies need not be harmonically related, and therefore slightly detuned oscillators can be used to provide chorused 'additive' sounds.

Each operator provides a single frequency component, or partial, or 'formant', with the EG controlling just that frequency.

Pairs

The simplest FM algorithm (apart from a single operator, which can only produce sine waves, of course) is a pair of operators: one carrier that is modulated by one modulator. The carrier EG and level control give control over the overall volume of the sound that is produced, whilst the level control and the envelope of the modulator control the modulation index of the FM. The timbral controls are thus the two operator frequencies, plus the modulator envelope and level controls.

Stacks

By taking a second modulator and connecting this to an FM pair, so that it modulates the modulator, a stack of three operators can be produced. Additional modulators can be added, although a stack of four operators is normally sufficient for most purposes. Since the pair formed by the two modulators produces an FM sound, the carrier which is modulated by this sound (and not by the sine wave that would be produced by a single modulator) is much more complex because each frequency in the modulating signal creates an FM with the carrier operator. Stacks are often used for pad sounds, where lots of slightly detuned harmonics and partials are used for producing rich, chorused sounds.

Multiple carriers

By connecting one modulator to more than one carrier, the same modulator can be used to control two carriers. By having different frequencies for the two carriers (or different envelopes), the output is two FM sounds that are related but separate. If the modulators have slightly detuned frequencies, then two similar but detuned sets of harmonics and partials are produced.

Multiple modulators

If several modulators are connected to one carrier, then each modulator can be used to produce part of the final sound, which can simplify the development of sounds. Having only one carrier operator means that controlling the output envelope is easier, but it also restricts the timbral possibilities because there is less flexibility in choosing the ratio between the carrier and the modulator frequencies (Figure 5.1.8).

Feedback

By connecting the output of an operator back to its frequency control input, the resulting feedback signal affects the output signal of the operator. In the simplest case, a single operator with a feedback loop will produce additional

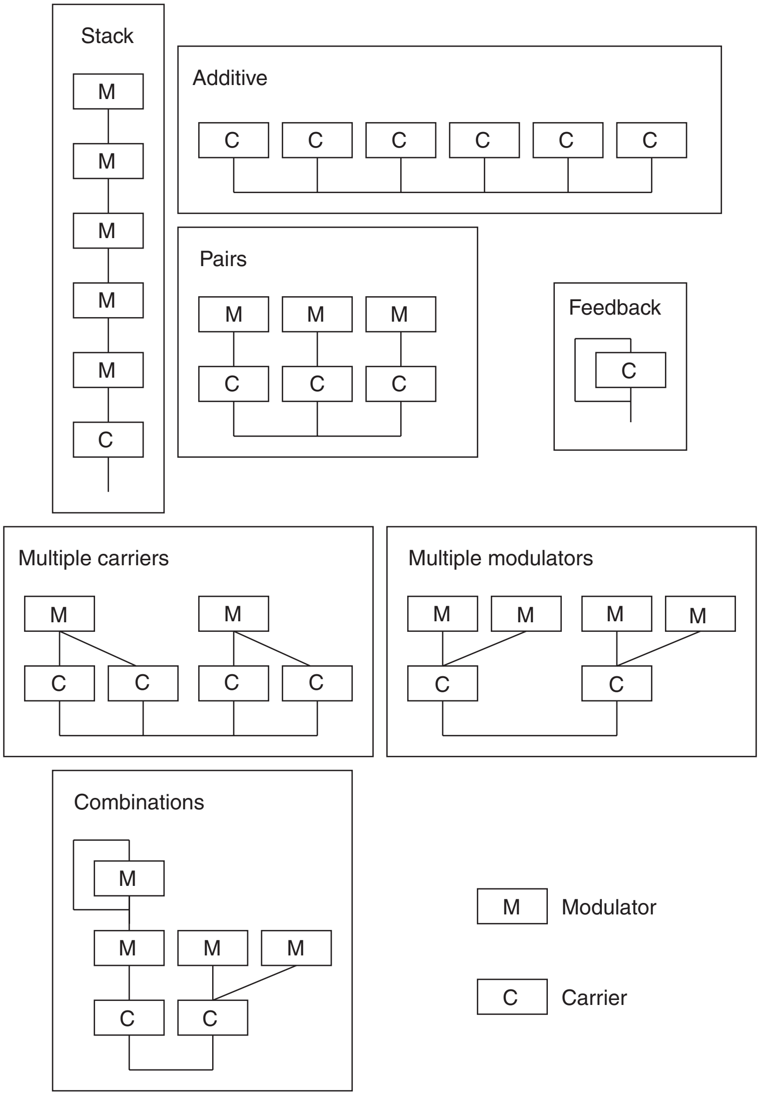


FIGURE 5.1.8

FM algorithms' summary. In these diagrams, C indicates a carrier operator, whilst M indicates a modulator operator. There are six basic arrangements of operators, plus a seventh consisting of combinations of parts of these. The examples shown here are for six operators, but the same topological arrangements apply to other numbers of available operators.

frequencies with a large amount of feedback, and this can sound similar to a sawtooth or a pulse type of waveform. Feedback around several operators can be used to produce very complex sounds. If too much feedback is applied, then noise-like sounds can be produced.

Feedback has always been one of the more interesting and less well-understood aspects of FM. The basic idea is that you take the output of the operator and connect it to the frequency control input (in some algorithms this is the same operator, in others you get a loop of two or three). On an SY-series FM synthesizer, you can patch several operators together and apply feedback between them. The 'feedback level' is a control over the level and therefore controls the modulation index: 0 is no feedback, whilst 7 is an index of about 13 on a DX7 or an SY. A modulation index of 13 produce, quite a lot of frequency deviation, and therefore the original sine wave is deviated well away from its basic frequency, but at a rate which is tied to itself. This produces lots of extra harmonics and perhaps even partials (the spectrum for a single operator on a DX7 with a feedback value of 7 has 23 harmonics) and a very contorted waveform. In fact with a feedback value of greater than 5, the underlying precision of the FM synthesis implementation used by Yamaha begins to become significant and the output begins to be noise-like, although the operator output level also affects the feedback, since the two level controls are in series! Below 5, the sound produced is merely richer in harmonics and partials.

Although the sounds produced by the feedback are described as 'noise-like', this does not usually mean that they are like the 'white' or 'pink' noise found in analogue synthesizers. With two operators, things rapidly get out of control once you start connecting a harmonic-rich waveform from an operator with feedback as the modulator of another operator. Aliasing and the finite precision of the FM synthesis 'engine' combine to produce a plethora of noise-like sounds, with not-so-flat spectra and lots of harmonics and partials – especially if you use non-integer ratios for the carrier and the modulator frequencies. Careful use of feedback level and operator output levels can keep things non-noise-like, and still in the realm of complex but interesting timbres. Because of the effects of aliasing, and the way that FM folds harmonics or partials when the modulation index is large, the resulting spectra may be rich in frequency content, but they are rarely flat – the noise is not white nor is it really colored, various shades of 'off-white', perhaps!

Because the output of an operator with feedback is a spectrum relatively full of harmonics and partials, changing the carrier or the modulator frequencies merely changes some of the harmonic and the partial amplitudes and the aliasing components. The only audible effect is often a change in the timbre or 'color' of the noise. Only with low-modulator indexes and low feedback values will the carrier and the modulator frequencies make any significant difference.

In the SY-series synthesizers, these problems with producing 'white' noise were solved by providing a noise generator. This produces white noise, and this sidesteps any need to use feedback to try and get a flat noise spectrum. Feedback noise sounds tend to be slightly too structured or grainy to fool the ear, whereas a simple maximal-length pseudo-random sequence is probably used by Yamaha's noise generator to provide white noise on tap. Using feedback creatively with all the flexibility offered by the SY-series synthesizers is worthy of further exploration.

In the FS1R, the noise generation is extended further by adding filtering, and the result is called an 'unvoiced' operator, referring to speech synthesis terminology. As a general rule, the pitched, harmonic or 'voiced' parts of FM have remained relatively the same throughout FM synthesis development, whilst the 'noise-like', inharmonic or 'unvoiced' part has seen the most development to try and extend and enhance the capabilities of FM synthesis. The FS1R might even be classified as being a combination of FM synthesis with formant synthesis. There is some basic information on DX-series FM feedback (Figure 5.1.9) in the book by Dave Bristow and John Chowning's now out-of-print book, FM Theory and Applications for Musicians (1986), pp. 133–136. (You may be able to find a PDF copy of this book on the Internet.)

Combinations

Most FM sounds are made from combinations of the simple algorithms. Two parallel stacks of three operators are often used because they enable two separate sounds to be combined, whilst each stack of three operators is a versatile FM sound source. Multiple modulators can produce complex sounds where each modulator contributes a distinct element to the sound, and they can each be controlled separately. The development process for FM sounds tends to be iterative, with operators being turned on and off to determine their effect on the sound each time as their parameters are changed. This technique is especially important where groups of operators are used to provide different parts of the sound. Unlike many methods of synthesis, FM allows the programmer to investigate the effect of minor changes both in isolation and in context.

DX-series FM synthesizers provide fixed algorithms where the topology can be selected from a number of presets. The presets provided typically include all of the possible arrangements of operators and many of the additional possibilities provided by adding one feedback loop. SY-series and subsequent FM synthesizers provide user control of the interconnections and multiple feedback connections, as well as preset topologies. Choosing a specific algorithm is largely a matter of experience. But in many cases, starting from a simple

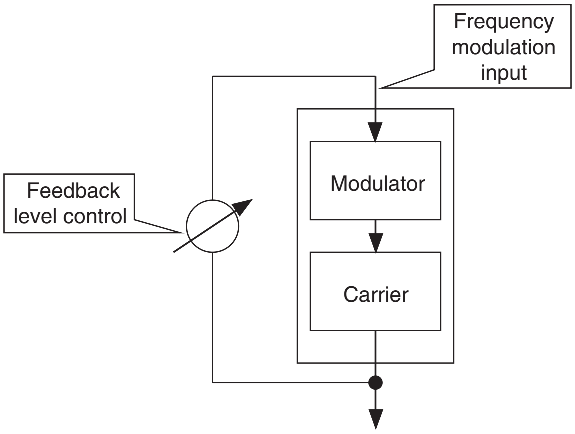


FIGURE 5.1.9 By feeding back the output of an FM algorithm to the FM input, it is possible to generate noise-like outputs. Some implementations have added specialized noise generators to supplement this method of generating noise-like sounds.

pair of parallel stacks is a good idea, because extra modulators (or carriers) can then be added as required. Familiarity with the timbral possibilities of a simple pair or stack of operators can be very useful in helping to produce FM sounds. Examining pre-programmed sounds can also help to reveal some useful techniques.

5.1.5 History

John M. Chowning's paper 'The synthesis of complex audio spectra by means of frequency modulation', in the *Journal of the Audio Engineering Society* in September 1973, was the first serious description of the practical use of digital technology to implement audio FM as a way of synthesizing timbres. This is very much a 'landmark' in digital synthesis; unlike additive synthesis, where the large number of required parameters made a digital realization unwieldy, FM showed that digital synthesis could be powerful and yet require only a relatively small number of controls.

5.1.6 Implementations

There are three strands of FM development from Yamaha: four, six and eight operators. Four-operator FM tends to be used in the lower cost and computer-oriented areas, whilst six- and eight-operator FM is used in 'professional' instruments. From 1982, Yamaha have continued to release FM instruments through to the early twenty-first century, although from the mid-1990s onwards their main focus has been towards S&S (Table 5.1.1).

Korg's DS-8 and 707 synthesizers from 1987 used FM technology as a result of a temporary pooling of research facilities by Korg and Yamaha in the middle 1980s. Many PC sound cards of the 1980s and 1990s used a Yamaha FM chip set to produce musical sounds.

Until the end of the 1980s, the FM chips and DACs used in FM implementations had limited resolution, and the resulting sounds had some background quantization noise. From the 1990s onwards, higher resolutions are used, and the FM has less of these digital artifacts. As with many 'retro' music fashions, the older sound has been subject to cyclic peaks of popularity, although adding in suitable emulated noise is also possible with more modern implementations.

The SY77 and the SY99 from the early 1990s added resonant filtering and sample replay to enhance FM synthesis. The FS1R from 1998 added additional filters in the form of formants (see Section 5.5.1) to eight-operator FM. The DX200 added interpolation between two sets of sound parameter settings in 2001.

Although Yamaha had acquired the patent rights to the commercial exploitation of FM in the early 1980s, there were several variants on FM that differ enough to be usable without actually infringing the patent. In fact, Yamaha's implementation actually uses phase modulation rather than frequency modulation. This has the effect of allowing an operator to be modulated by another without changing its pitch (in FM, the pitch would change if the modulating

Table 5.1.1 FM Implementations

	Sine waves													
						Sine variants								
									Samples					
4 Operator	GS2	DX9	CX5M	DX21	DX100 FB01	TX81z 707	DX11	V50 TQ5	SY22 TG33	EVS1	SY35	GS2	DX200	
6 Operator		DX7	DX1 TX7	DX5 TX816		DX7II DX7S			SY77	SY99				FM7
8 Operator	GS1					HX...		V80				FS1R		
	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1998	2001	2002

waveshape was asymmetric), which makes programming musical sounds considerably easier because the timbre changes rather than the pitch. Trying to create stable pitched FM sounds on an analogue modular synthesizer will quickly show the difference between modulation in phase and frequency.

Casio's CZ series of synthesizers used dynamic waveshaping, but their later VZ series of synthesizers used another FM-like phase modulation method, calling it phase distortion (PD). Eight operators were available with eight-stage envelopes.

Peavey has also used the term 'phase distortion' to describe synthesizers that appeared to be S&S instruments, and not a variant on FM.

FM produced on computers using software initially offered enhanced sophistication at the price of non-realtime operation, although faster processors and DSP technology now makes FM more accessible and immediate. In 2002, the FM7 plug-in from Native Instruments offered real-time six-operator FM that was DX7 sound compatible and added a number of enhancements like more sophisticated SY-series style noise generation and additional resonant filters. Some purists complained that the background quantization noise inherent in the early FM implementations was missing. By 2007, several other FM software implementations had been released.

5.2 Waveshaping

Waveshaping is a way of introducing controlled amounts of distortion onto a waveform. This differs from the 'fuzz box' type of distortion that is used by guitarists, because it is used on the 'monophonic' outputs of the oscillators, and therefore it merely changes the shape of the waveform without adding in all the intermodulation distortion that happens when more than 1 note is passed through a waveshaper or a fuzz box.

Waveshapers are non-linear amplifiers. This means that they provide control over the way that the amplifier processes incoming signals to produce an output. For an amplifier with a fixed gain of 2, you expect to get an output that is twice the input. If a graph of input against output for a linear amplifier is plotted, it would be a straight line through the origin (zero) of the graph. This line is called the 'transfer function' of the amplifier – it shows the way that the input is 'transferred' to the output. In fact, the straightness of this line can be used as a measure of the quality of an amplifier, since a perfect amplifier would have a perfectly straight-line transfer function. If the amplifier did not have a gain of 2 for high levels of input signal, then the transfer function graph would be curved at high input levels, which means that an audio waveform that is passed through the amplifier will change shape. Changing the shape of the transfer function changes the shape of the waveform.

It is the convention that transfer function graphs always have the input plotted horizontally and the output vertically. The scaling is also arranged so that the input and the output ranges are from 1 to –1, with the zero point of both axes being in the center of the graph. The input sine wave moves completely across the horizontal axis once per cycle: from a value of 0 to 1, then back through the zero position to 1 and then back to zero again. The output waveform is dependent on the transfer function, although the maximum and the minimum outputs are normally 1 and –1, respectively (Figure 5.2.1).

Although this sounds like an easy way to produce extra waveshapes, it actually does rather more than that. Distorting the shape of a waveform changes the harmonic content of the waveform; in fact, in most cases, it adds harmonics rather than subtracts them. If the transfer function is symmetrical about the horizontal axis or has a rotational symmetry, then the harmonics that are added will be the odd harmonics, whilst if the transfer function is symmetrical about the vertical axis or shows a mirror symmetry, then only even harmonics will be produced. So with a sine wave and a waveshaper, it is possible to use different shapes of transfer function curves to produce outputs that have a wide variety of harmonic contents. Now using a sine wave and producing extra harmonics from it sounds like FM, and in fact, with the right type of transfer function curve, waveshaping can produce sounds which are very FM-like in character. But it can also produce sounds which do not have 'FM-like' characteristics. When FM was at its peak of popularity in the mid-1980s, Casio used a waveshaping-based synthesis technique in their CZ-series of synthesizers, but called it phase distortion.

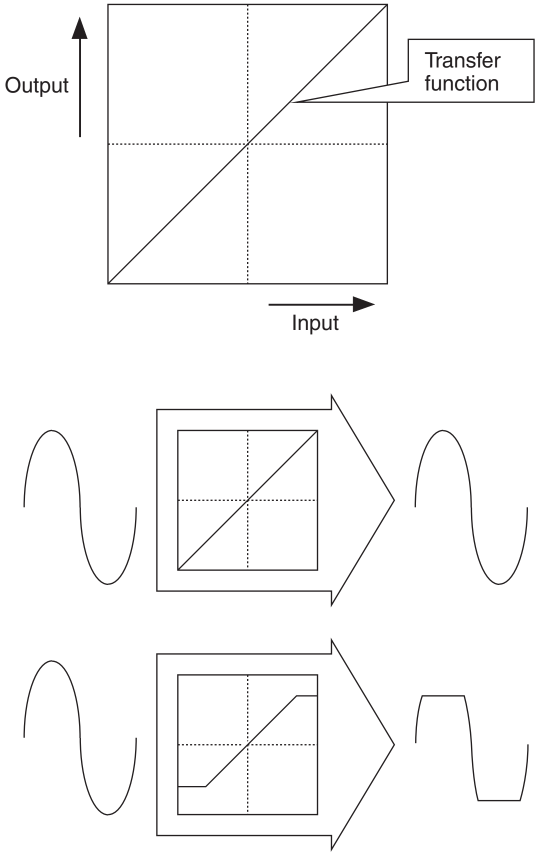


FIGURE 5.2.1
A transfer function is a graph which relates an input to an output. In this example, a straight-line transfer function allows a sine wave to pass unconverted, whilst a transfer function which has a steeper slope and two flat zones converts a sine wave into a trapezoidal waveform.

5.2.1 Phase distortion

The name ‘Phase distortion’ comes from an alternative way of looking at transfer functions. If a wavetable containing a sine waveform is being read out and the rate of reading changes, then this will cause the sine wave to be distorted. The result looks like the effect of a transfer function, but is really just the result of moving through the sine wavetable faster or slower. Since changing the rate of reading is equivalent to a change of phase in the sine wave output, then this is known as ‘Phase distortion’.

In general, transfer functions and phase distortion are just different ways of producing waveshaping.

Using a sine wave has advantages and disadvantages. It is possible to calculate a transfer function that will produce any given spectrum (but not waveshape) from a sine wave input. The technique involves the use of Chebyshev polynomials and enables a change in the frequency of input sine wave. The resulting output frequency is multiplied by the order of the Chebyshev function; therefore, a fourth-order function would produce an output sine wave which is four times the input frequency. By adding together several Chebyshev functions it is possible to produce a composite transfer function that will then produce any required spectrum. The calculations of the relevant Chebyshev polynomials are simplified if the input waveform is a sine wave. Because the sinewave shape has two different times when the same value occurs, then some waveshapes cannot be produced, but this restriction is often not a problem, since the harmonic content is normally more important for a specific timbre.

The input waveform to a waveshaper need not be a sine wave. If a sawtooth wave is used, then the waveshaper is little more than a look-up table for output values, and therefore resembles a wavetable oscillator. The positive and the negative half cycles of the sawtooth wave just map onto images of the output waveform and thus the two half cycles can be different. Effectively, the two half cycles, can be thought of as two separate transfer functions, although they normally share a common point at the origin. But for a sine wave, the symmetrical nature of the waveshape means that there is more redundancy, which in turn means that there is scope for more independence of transfer functions. A sine wave can only be converted into other waveforms of a particular class of shapes by using a single non-linear transfer function – basically only those where the first quarter of the cycle is the same as the next quarter of the cycle, but where the second cycle is time-reversed. As the sine wave input moves up and back down the transfer function horizontal axis, the symmetry is inevitable. The same applies to the third and the fourth quarters of the cycle.

But by providing different transfer functions for each of these quarter cycles, the waveshaper can be used to convert a sine wave into waveforms that do not have this first and second quarter-cycle mirroring. This means that a single transfer function graph can have two separate halves for each half cycle of the sine wave, and the symmetry produces waveforms that have a large content of even harmonics. In contrast, if there are two separate transfer functions, with each quarter cycle having its own graph, then any waveshape can be produced (Figure 5.2.2). This type of quarter-cycle waveshaping is used in the second generation of Yamaha FM synthesizers to produce additional waveforms from a wavetable ROM containing just a single high-precision sine wave (see also Figure 4.3.4).

Although waveshaping can be used as a general-purpose tool for changing the shape of a waveform, it can be arranged so that the audible behavior of the waveshaper is similar to the VCF found in an analogue synthesizer. Using digital technology to emulate familiar 'analogue' characteristics is a continuing theme of most digital synthesis methods. In the case of an analogue low-pass filter, harmonics are successively added as the filter cut-off frequency is increased; hence, the output waveform is initially a sine wave at the fundamental frequency. As harmonics are added, the shape of the output waveform will change until with the filter fully 'open', then all frequencies will pass through and the output waveform should have the same shape (and frequency spectrum) as the input.

For a basic waveshaper implementation, this 'filter emulation' behavior would seem to imply that the transfer function is changing dynamically, and it is possible to produce transfer functions that scale in size to produce this effect. However, by designing the transfer functions carefully, and by ensuring that the

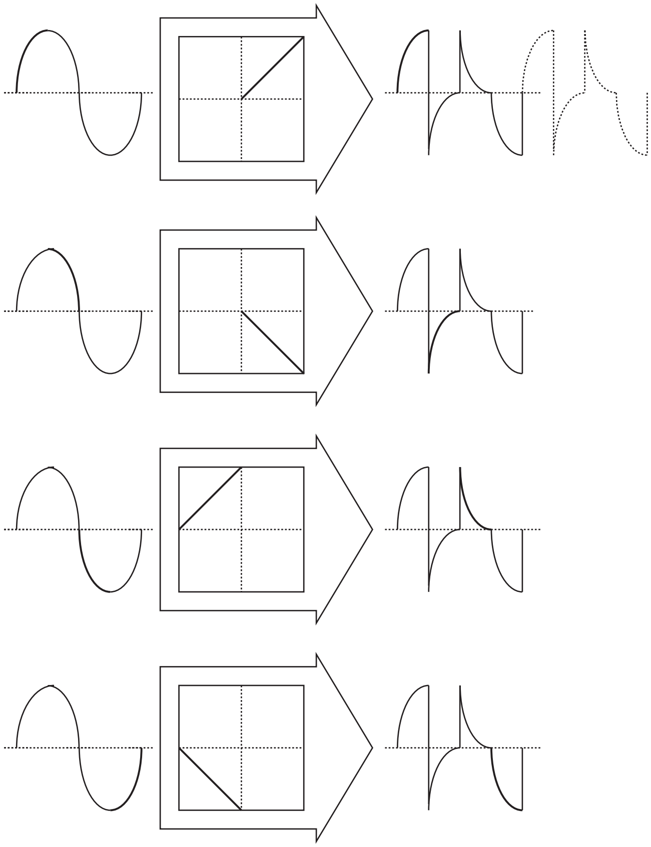


FIGURE 5.2.2 By using separate transfer functions for each half or quarter cycle of a waveform, it is possible to produce almost any required output waveform from an input sine wave. In this example, each quarter cycle of the input sine wave has its own transfer function. The output waveshape is the concatenation of the four output quarter cycles.

transfer function curve passes through the origin (zero) of the graph, it is possible to produce simple waveshapers with just one fixed transfer functions that can be used with inputs that are smaller than the 1 and –1 maximum and minimum levels, respectively.

As a simple example, consider a transfer function that is a straight line as it passes through the zero points of the input and the output axes, but which gradually curves away from a straight line as it moves away from the zero point. At low amplitudes of input sine wave, the output will also be a sine wave, because the linear portion of the transfer function will be used. But as the input level is increased, the non-linear parts of the transfer function will be used, and the waveform will be distorted. As the level increases to the maximum, then the largest waveform distortion will be produced. This tends to produce an output signal that starts out as a sine wave, but which gradually acquires additional harmonics as the amplitude increases, in much the same way as an analogue VCF. By arranging an amplifier to correct for the amplitude changes, it is possible to produce an output that does not change in level as the 'filtering' action takes place (Figure 5.2.3).

The audible result of this 'waveshaping' process is a smooth transition from a sine wave to a waveform containing a number of harmonics. But unlike an analogue VCF, the evolution of the waveform is dependent on the way that the transfer function changes with the input amplitude. This means that the harmonics do not need to be added in a progressive sequence comparable to a low-pass VCF, but can change in other ways which can be more interesting to the ear. Complex changes of harmonic content are also found in FM, although the evolution of FM waveforms is fixed by the Bessel functions. For a waveshaper-based synthesizer, the transfer function is not fixed and therefore can produce more sophisticated and varied harmonic changes, at the price of an increased need for mathematical understanding on the part of the designer of the transfer function. Unlike FM, the additional frequencies that are produced by waveshaping are always harmonically related to the input frequency, since the waveshaping is based on the shape of one cycle of the waveform.

Some manufacturers have used waveshaping in a much more limited sense. For example, the Korg 01 series S&S synthesizers implement waveshaping, but it is in a very limited form with only one non-linear transfer function. It is used to process the outputs of the oscillators and is really limited to just adding in a few extra harmonics to the raw samples. Casio-style dynamic waveshaping is a much more powerful technique; if Korg had moved the waveshaper after the VCF or the VCA, or made the transfer curve controllable or dynamic, then the possibilities for timbral change would have been much greater.

5.3 Physical modeling

Although other digital methods of sound synthesis tend to try and emulate the terminology of functions of analogue synthesis, mathematical modeling breaks away from these conventions. There are no samples, no function generators

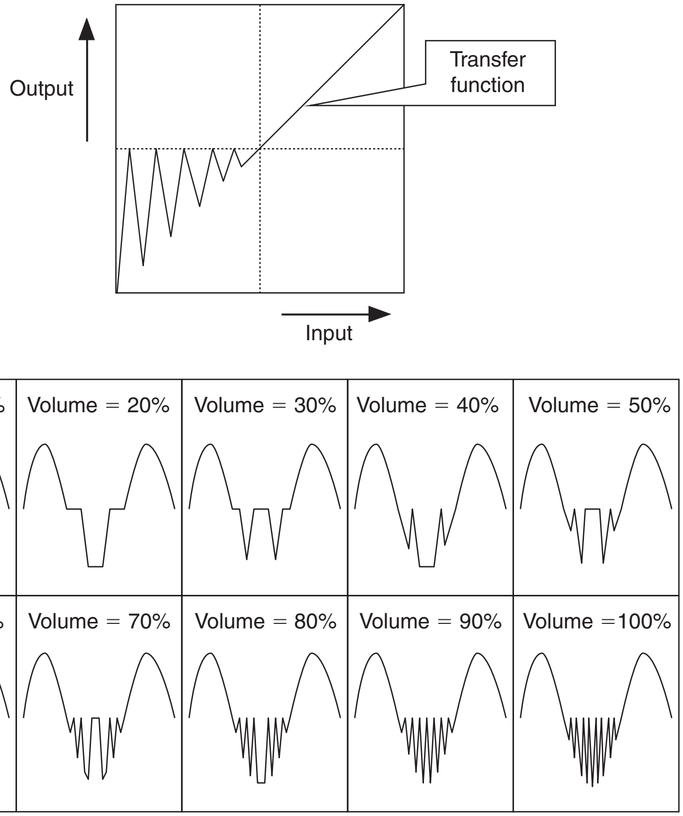


FIGURE 5.2.3 Dynamic waveshaping alters the input level and then scales the output to compensate. In this example a sine wave is passed through an asymmetric transfer function which is linear for positive inputs, but a complex function for negative inputs. The outputs for different levels are shown; it can be seen that the output waveform changes as the input level is increased in much the same way as opening a VCF does on an analogue subtractive synthesizer.

and much less use of envelopes and filtering, and yet despite throwing away almost everything with which the synthesizer user may be familiar, instruments that use modeling techniques can produce sounds that feel so much like real instruments that it is hard to think of them as electronically produced.

There are many variations in the basic idea of using mathematical models to produce sounds. In this section, two will be examined, and a third is covered separately in Section 5.4:

- 'Source-filter synthesis' is a simplified modeling technique that concentrates on the interactions between the two major component parts that produce an instrument's sound.
- 'Physical modeling' attempts to describe the complete instrument with a complex and sophisticated model.
- 'Analogue modeling' describes analogue synthesizer circuitry (see Section 5.4).

5.3.1 Source-filter synthesis

Instead of trying to describe how a complete instrument works in terms of equations, source-filter synthesis looks for a way that the important elements can be encapsulated in a form that provides control, but is easy to use. It turns out that there is a way, and it comes from research into speech. When you speak, your vocal cords are vibrated by the air that rushes past them, and this raw sound is then modified by the complex set of tubes and spaces formed by your throat, nose, mouth, teeth, lips and tongue. A physical model of this would need to consider the velocity of air, pressure, tension in the vocal cords, the space between them, their elasticity and soon; and trying to work out the exact mechanisms for how they vibrate could be difficult and time consuming. The more pragmatic approach of source-filter synthesis asks: what does the raw sound produced by the vocal cords sound like, what sort of filter do the throat, mouth and nose form and how do these two parts interact with each other?

Source-filter synthesis assumes that musical instruments can be split into the following three parts (Figure 5.3.1):

1. Drivers, which produce the raw sound. Examples are the hammer hitting a piano string, or the pick plucking a guitar string, or the reed vibrating in an oboe.
2. Resonators, which color the sound from the driver. Most musical instruments exhibit some sort of resonance, often the whole of the instrument vibrates along with the sound to some extent, and the way that it vibrates affects the frequencies that are emphasized and suppressed.
3. Coupling between the driver and the resonators, which determines how the two interact with each other.

In a real instrument, the drivers and the resonators are often very closely connected. They interact with each other – the hammer hitting a piano string causes the string to vibrate, but the vibration of the string is affected by the fact that the hammer is touching the string, has probably stretched the string slightly when it moved the string and has added in a low-frequency thump. The act of setting the string vibrating depends on the hammer – you cannot have the sound without it, but the hammer affects the sound. The two are inextricably interconnected. In source-filter synthesis, the two are separated, but the

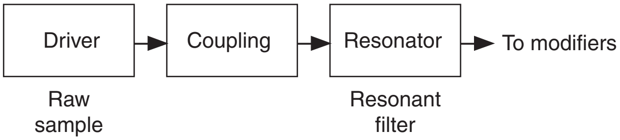


FIGURE 5.3.1 The driver produces a raw sample sound which has had the effect of any resonance removed artificially. This is then coupled to a resonator section through a coupler section, which allows control by the performer.

same interactions can be produced by controlling the way that the driver and the resonator are connected together.

The basis of this technique is to separate the driver and the resonator, and then couple them together so that they can interact. Instead of trying to model the driver, the technique assumes that the raw driver is more or less fixed, whilst the coupling to the resonator is the important aspect. This means that a driver 'sample' can be used to provide the stimulus for a resonator model through a coupling device – there is no need to try and create a model for the driver at all. Modeling resonators is much easier, since they are just filters, and filter theory is well understood. This means that it is easy to produce a number of driver 'samples', and resonator specifications, and couple them together. This approach means that a large number of possibilities are opened up without any need for careful research into musical instruments.

The coupling part of source-filter synthesis deals with the interconnection and interaction between the driver and the resonator. This is probably the major part of the technique to use the same approach as 'physical modeling'. A bowed string is a good analogy for the process. The player of a stringed instrument can control parameters like the position of the bow on the string, and how hard the bow is pressed onto the string. The resonator can be changed as well, for example, it may be a fixed resonance or the one that changes with the playing pressure. The combination of a simple model for the coupling, plus the fixed driver 'sample' and the variable resonance, produces a versatile synthesis 'engine'.

The driver output is not a conventional audio sample. Because this is the raw driving force without any modification by a resonator, it is not possible to actually place a microphone and sample it directly. One approach to determining what it would sound like is to take the final sound of the instrument and then remove the effect of the resonances. If you listen to a raw driver signal, then it will sound very bright with an emphasized initial transient, almost like high-pass filtering. But since most resonators act as band-pass or low-pass filters, coupling this driver signal to a resonator transforms it into a sound that suddenly takes on a more normal sound. In fact, it sounds much like the sample that you would actually hear in a recording (which is the sample of course – the result of a driver coupled to a resonator). The difference is that by separating out the driver and the resonator and by changing the parameters that control the resonator, you can change the timbre. This is not possible with a conventional sample at all. It is easy to design resonators that behave like strings, tubes, cones, flared tubes, drums and even customized ones. Most will have a combination of band-pass or low-pass response, combined with one or more narrow peaks or notches.

Although this may sound like the S&S 'pre-packaged' sample concept, in fact, the combination of driver 'sample', coupling and resonator produces sounds that can change their harmonic content much more than any S&S sample that can be merely filtered. Remember that this is not a physical modeling

The Technics WSA1 keyboard, released in 1995, used source-filter synthesis to produce its sounds. Although widely praised for its sound, there were no follow-up instruments using the same technique.

instrument, although it is similar in some respects, especially the coupling section. What you lose is the transition between the notes and the behavior outside of the basic sound generation; therefore, whereas an instrument based on physical modeling will move from 1 note to another in much the same way as a real instrument, one using source-filter synthesis will merely play 2 notes, one after the other. This is most noticeable for brass sounds, where a physically modeled instrument such as the Yamaha VL1 will exhibit the characteristically 'overblown' brassy natural series of notes when the pitch is changed with the pitch-bend control, whilst a source-filter synthesis instrument will merely bend the note.

It remains to be seen whether source-filter synthesis will reappear in the future as a major method of synthesis or will it remain as part of the hybrid digital synthesis methods (see Section 5.8) or even just part of the tools used to produce the inharmonics and transient samples used in S&S synthesizers to augment the basic instrument sounds.

5.3.2 Physical modeling

The 'physical modeling' technique uses DSP chips to create a mathematical model of how some real musical instruments work. Instead of the conventional 'source and modifier' approach used by many S&S instruments, where a basic sample sound is modified by a filter and envelopes to produce a finished sound, a physical modeling instrument uses its internal model of an instrument to create the whole sound in one operation. Because the model covers the entire instrument, it behaves like the actual thing, and therefore, it also produces realistic transitions between notes, not just the notes themselves. It can produce sounds that emulate the behavior of the real thing, often with astonishing realism. But the depth of detail that is required is formidable: you need to know a huge amount about the physics of musical instrument, acoustics and mathematics and then you need to convert this into software and electronics.

The techniques and algorithms for modeling musical instruments did not reach the level of sophistication where they could be done in real time without the aid of rooms full of supercomputers until the mid-1990s, and the number of types of instrument that can be adequately described is still quite small. The future may produce additional instrument descriptions, and physical modeling will be able to utilise these, but physical modeling has so far been only a limited success. In particular, it tends to be used for minor variations on existing instruments rather than in producing new synthetic sounds. Paradoxically, it may be the very precision and detail that is required to produce a physical model that prevents it from being a user-programmable synthesis tool.

Mathematical models

Using mathematics to make models of real-world objects is common in engineering, but it is more unusual to find it used in musical applications. The underlying concept is the same for any model: you look at the inputs, outputs, their interconnections and dependencies and then determine the equations that connect them all together. Imagine a tap and a bucket with a hole in it. Suppose that the tap can provide anything up to 10 litres of water per minute, the bucket holds 20 litres and that the hole leaks at the rate of 1 litre per minute. Ignoring the leak, the fastest time taken to 'fill' the bucket by the tap (when full on) is the time it takes for the tap to provide 20 litres of water, which would be 2 minutes (i.e., 20 litres at 10 litres per minute = 2 minutes) (Figure 5.3.2).

When the effect of the hole is taken into account, the figures change correspondingly. In the first minute, 1 litre of water will escape out of the hole, and therefore only 9 out of the 10 litres supplied by the tap will be in the bucket at the end of the first minute. During the second minute, another litre of water leaks away, and therefore, there will only be 18 litres in the bucket, and thus, it will obviously take slightly longer than the original estimate of 2 minutes because the tap will still need to provide just over 2 more litres of water …

By using this simple 'tap and bucket with hole' model, it is possible to make several other deductions based on how the system works. For example, if the tap supplies less than 1 litre per minute, then the bucket will never fill up because the hole leaks at 1 litre per minute. When there are 20 litres in the bucket, then it will begin to overflow and if you subtract the 1 litre per minute leak, then the overflow rate is the tap supply rate (from just over 1 to 10 litres per minute) minus the leak rate; therefore for the tap fully on, the bucket will overflow after just over 2 minutes have passed, and the overflow rate will be 9 litres per minute.

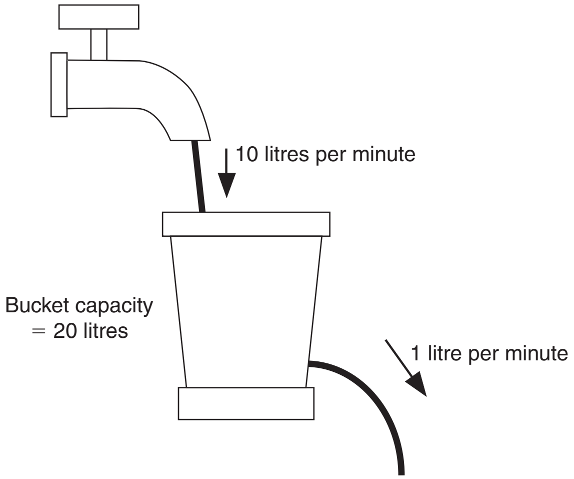


FIGURE 5.3.2 This 'bucket' diagram shows the power of a mathematical model in predicting the behavior of a real-world system. Physical modeling uses much more complex models of musical instruments to produce sounds.

As you can see, with just simple calculations we can make some quite complex predictions about the way that the real-world works. The models of how musical instruments, which are used in physical modeling wave, are obviously more complex than this example, but it is based on the same principles: you measure what happens, produce a description of what is happening and then you use this information to work out what will happen.

Model types

Physical modeling synthesis falls into two distinct areas: continuous and impulsive.

1. Continuous models deal with blown or bowed instruments, where there is a continuous transfer of energy into the instrument from the air flow or the bow. The sound that is produced thus carries on as long as the energy is transferred. Typical examples include a trumpet and a violin.
2. Impulsive models are for plucked or struck instruments, where a sudden 'impulse' of energy is transferred to the instrument, which then produces a sound as it responds to this input. The sound decays away naturally since energy is lost as friction, sound and movement once the initial input is taken away. Typical examples include a piano and a snare drum.

Apparently continuous events that are actually discrete are more common than many people expect. A narrow stream of water from a tap may appear to be continuous, but high-speed cameras show that many are formed from many individual droplets of water.

Sometimes, the distinction between a continuous and an impulsive model is not immediately obvious. In the case of a violin, the bow scraping on the string transfers energy to the string because it is rough.The string catches on the rough surface of the bow and is pulled away from its rest position; it is released when the tension in the string exceeds the friction, and the string then jumps back to its original resting position. Each of these tiny movements of the string is an impulse, but they happen quickly enough to have much the same effect as a continuous transfer of energy.

In some modeling terminology, the drivers are referred to as the excitation signal.

For continuous models, the two major parts of most blown/bowed musical instruments are: the bit that you blow or move and the part that vibrates. In a reed instrument, air is blown into a mouthpiece, whilst for a trumpet, the lips move and control the flow of air. For a stringed instrument, the bow scrapes across the string. In all of these, the player is forcing the instrument to make a sound; hence, these are called drivers, just as in source-filter synthesis. In contrast, the air inside a saxophone or a trumpet vibrates inside a tube and therefore makes a sound, or the string vibrates and moves the air around to make a sound, and these make up the resonator part of the model. Although in a real instrument there are normally fixed combinations of drivers and their corresponding resonators, with physical modeling a reed type of driver feeding into a string-type resonator is entirely possible, even though a real-world equivalent would be difficult to construct.

The drivers in continuous models transfer energy into the resonator, but in order for this to be converted into a sound, the energy needs to be converted from a steady stream into a repeated cycle of variations in air currents to produce a sound. In the case of a violin, the bow rubbing against the string produces vibrations in the string, and the resonator formed by the string and the body of the violin then reinforces some vibrations and dampen others. For a stream of air in an oboe, the opening and closing of the reed produces a stream of air that varies in pressure, and the resonator formed by the tube and holes in the oboe reinforces some of the variations and dampen others. The driver specification thus needs to take into account how these initial vibrations are produced, and how they are coupled to the resonator.

The 'Karplus–Strong' (Karplus and Strong, 1983) plucked string algorithm is just one example of many impulsive models. This algorithm uses a damped resonator and a step input of energy to simulate what happens when a string or a bar is plucked or struck. The resonator produces a note at its resonant frequency, with additional harmonics caused by its other resonances, and the decay of the sound occurs because the resonator has no source of power except for the initial input of energy. Therefore as the energy leaks away, the sound decays. The way that the resonator loses energy, and the way that it produces the sound output are critical to the harmonic content and the way that it changes with time. The damping depends on the way that the string or bar is mounted or supported, whilst the mass of the string or bar, the tension in the string and the dimensions of the bar can all affect how the sound changes with time.

The Karplus–Strong algorithm is simulated by using a time delay to model the movement of waves along the string or bar. The reflections at the end of the bar or string are set so that some energy is removed from the wave, and therefore the reflected wave is reduced in amplitude. The initial step input can be just a sudden change in level, but it can also be a brief pulse of noise. More complex models may also take into account more details of the initial input of energy, which need not be a sudden step input of energy, but may have an 'envelope' and other characteristics that affect the way that the energy is transferred to the resonator.

The hammer of a piano is one example of the complexity of characteristics that needs to be considered in an impulsive model. The hammer is accelerated by the piano action and hits the string. It then moves the string away from its rest position, but this is cushioned by the felt; therefore, the transfer of energy does not happen instantaneously. Although the felt is being compressed by being pressed against the string, the string itself is starting to vibrate. The hammer continues to move the string away from the rest position until the tension in the string is equal to the force expended by the hammer, and the string then moves back towards its rest position, and the hammer bounces off it. This is not a simple 'step change' transfer of energy to a resonator, but a coupled system where the string is part of the driver and the resonator; and the felt acts to smooth the transfer of energy to the string both when the hammer hits the string and when it bounces away from the string.

Practicalities

The complexity of the mathematical models that have been used so far in physical modeling synthesis have been such that the manufacturers of commercial units have usually chosen to present a number of fixed preset instrumental sounds. The user cannot program these sounds, other than changing their response to performance controllers and changing some modifiers. Although this is very different to most previous synthesizers, it is exactly how real instruments are treated – you do not take a drill to a saxophone and try making holes in the metalwork! Instead, you use the mouthpiece to control the sound through a combination of air pressure, lip pressure, throat resonance, vocal cords and your tongue.

The ordinary household bath can be used to illustrate how a digital waveguide works. Having filled the bath to about half the capacity, a hand is used to cyclically move the water back and forth by a few centimetres at a frequency of approximately 1 Hz at one end of the bath. Some experimentation on the frequency of movement will be needed, but when the correct frequency is reached, then the ripples or the waves in the water will travel along the bath, bounce back from the far end and return to the end where the hand is still moving the water. At the right frequency, the returning ripples or waves will reinforce the ripples generated by the hand, and the size of the ripples or waves will increase. The movement of the hand should be stopped before the waves are large enough to go over the side of the bath.

The models that have been used in the initial physical modeling instruments are complex enough to provide exactly the sort of subtle and expressive control over timbre and pitch that you would expect from a real instrument. And there appears to be quite a lot of scope for modeling a wide range of instruments, but several academic papers have commented that there are only good models for a limited number of real instruments and that much more research still needs to be carried out. Digital waveguides are mentioned several times in the research literature of physical modeling and are a very computationally efficient way of simulating a resonator pipe or string by using DSPs and can be used in continuous and impulse models. A digital waveguide is essentially a delay line that has one or more time taps for feedback from the output to the input, and where the input is not a conventional audio signal, but a driver signal consisting of a series of shaped pulses.

Digital waveguides are used in different ways to produce different types of resonator. Simple tube-based instruments can be modeled with a simple waveguide for the tube, but often require complex driver models. Stringed instruments can be modeled with two waveguides: one for each side of the point where the string is plucked or bowed. Brass instruments can be modeled with several linked waveguides for the exponential horn.

Physical modeling can require a large amount of data to specify a specific timbre. For example, the Yamaha VL1 duophonic 'virtual acoustic' synthesizer uses 387 Kbytes to store 128 patches, which is roughly 3000 bytes per patch. For comparison, a DX7 FM patch uses only 155 bytes and only 128 bytes in the compressed form! Even so, the size of the VL1 file is still tiny compared to the size of a sample in an S&S synthesizer, where about 88 Kbytes of storage are required for each second's worth of sample.

Controlling the instruments provided by a physical modeling synthesizer can be difficult because of the large number of parameters that may need to be manipulated. Keyboard control is useful for pitch and velocity control, but it is not as natural and interfaces as a wind controller. Keyboards have the disadvantage of a naturally polyphonic keyboard, whilst a blown instrument is normally monophonic. Unfortunately, despite the advantages of a wind-instrument-like controller, the keyboard has still appeared on the first generation of commercial physical modeling instruments. Blowing can be easily simulated by using a breath controller, but lip or bow pressure, muting or string damping are less obvious, and foot controllers, velocity and after-touch can be used, although it requires practice for a keyboard player to become familiar with the use of additional controllers.

Experimentation

It is not necessary to have sophisticated digital workstations to experiment with physical modeling synthesis. Using conventional recording studio equipment, it is possible to try out the underlying principles for real. All that is needed is an audio delay line (Figure 5.3.3) with a few milliseconds of delay (almost any effect processors with an echo or a delay setting will do), a limiter or compressor/limiter, a noise generator (or synthesizer with noise generator) and a non-linear amplifier (or a dynamics processor or a fuzz box). The non-linear amplifier is an analogue equivalent of the waveshaper described earlier in this chapter – almost any operational amplifier (op-amp) can be used to provide this function (Clayton, 1975).

The basic idea is to connect the output of the delay line to the limiter, the output of the limiter to the non-linear amplifier and then the output of the amplifier back to the input of the delay line. The noise generator should be mixed into the input of the delay line as well. The output of the delay line also serves as the output of the system (Sound on Sound, February 1996). By adjusting the feedback and injecting pulses of noise into the system, it should be possible to get percussive sounds that decay away as per Karplus–Strong synthesis, whilst with higher levels of feedback, sustained continuous tones should be produced, whose timbre can be changed by adjusting the non-linear amplifier settings. By sampling the results into a sampler, some of the more interesting or useful timbres can be stored for future use.

Notice that the amount of delay is inversely proportional to the pitch. The minimum delay time thus determines the highest pitch that can be

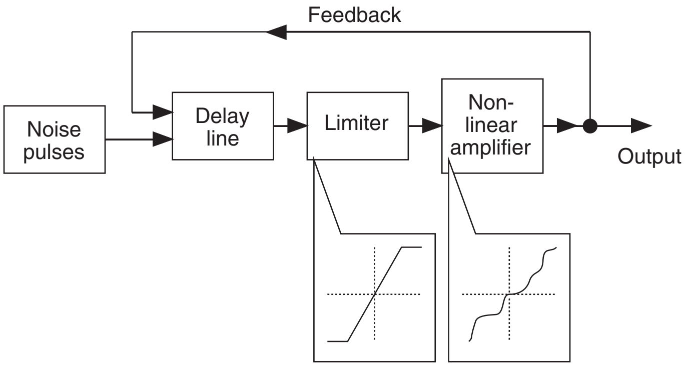


FIGURE 5.3.3 A delay line can be used as the basis for experimentation into physical modeling using analogue audio equipment.

produced. Also notice that the delay time needs to be very precisely controllable to produce specific pitches. For example, a 440-Hz note requires a delay of 2.2727 recurring milliseconds. Table 5.3.1 shows the relationship between time delays and frequency for this experiment.

Table 5.3.1 The Relationship Between Time Delays and Frequency

Delay Time (milliseconds)	Frequency (Hz)	Delay Time (milliseconds)	Frequency (Hz)	Delay Time (milliseconds)	Frequency (Hz)	Delay Time (milliseconds)	Frequency (Hz)
		3	333.33	6	166.66	9	111.11
0.1	10000	3.1	322.58	6.1	163.93	9.1	109.89
0.2	5000	3.2	312.5	6.2	161.29	9.2	108.69
0.3	3333.33	3.3	303.03	6.3	158.73	9.3	107.52
0.4	2500	3.4	294.11	6.4	156.25	9.4	106.38
0.5	2000	3.5	285.71	6.5	153.84	9.5	105.26
0.6	1666.66	3.6	277.77	6.6	151.51	9.6	104.16
0.7	1428.57	3.7	270.27	6.7	149.25	9.7	103.09
0.8	1250	3.8	263.15	6.8	147.05	9.8	102.04
0.9	1111.11	3.9	256.41	6.9	144.92	9.9	101.01
1	1000	4	250	7	142.85	10	100
1.1	909.09	4.1	243.9	7.1	140.84	10.1	99
1.2	833.33	4.2	238.09	7.2	138.88	10.2	98.03
1.3	769.23	4.3	232.55	7.3	136.98	10.3	97.08
1.4	714.28	4.4	227.27	7.4	135.13	10.4	96.15
1.5	666.66	4.5	222.22	7.5	133.33	10.5	95.23
1.6	625	4.6	217.39	7.6	131.57	10.6	94.33
1.7	588.23	4.7	212.76	7.7	129.87	10.7	93.45
1.8	555.55	4.8	208.33	7.8	128.2	10.8	92.59
1.9	526.31	4.9	204.08	7.9	126.58	10.9	91.74
2	500	5	200	8	125	11	90.9
2.1	476.19	5.1	196.07	8.1	123.45	11.1	90.09
2.2	454.54	5.2	192.3	8.2	121.95	11.2	89.28
2.3	434.78	5.3	188.67	8.3	120.48	11.3	88.49
2.4	416.66	5.4	185.18	8.4	119.04	11.4	87.71
2.5	400	5.5	181.81	8.5	117.64	11.5	86.95
2.6	384.61	5.6	178.57	8.6	116.27	11.6	86.2
2.7	370.37	5.7	175.43	8.7	114.94	11.7	85.47
2.8	357.14	5.8	172.41	8.8	113.63	11.8	84.74
2.9	344.82	5.9	169.49	8.9	112.35	11.9	84.03

Summary

Physical modeling is just one of the many possible methods of digital synthesis based on sophisticated software rather than just DSP hardware. It can produce expressive, astonishingly 'real' feeling instrument sounds, and this can apply even to the impossible synthetic ones extrapolated from the models. In common with other synthesized sounds, these are not a replacement for real instruments, more a whole new set of them.

Physical modeling technology began to appear in a range of products in the mid-1990s. Technics produced a source-filter-based physical modeling synthesizer in 1995, whilst Yamaha and Korg produced several physical modeling products, and MediaVision produced a PC card using physical modeling techniques. These were the first examples of physical modeling in commercial instruments, and whilst successful, they were limited in the instruments that they could model, and the lack of user control meant that they were seen in many ways as being the equivalent of samplers that could only replay the sounds of a few instruments. Although this replay was very good, and in many cases better than a sampler in terms of performance accuracy, the limitations were not appealing. When the first physical modeling instruments appeared, they were expensive and monophonic or duophonic, whereas the first source-filter synthesis instruments that appeared were polyphonic for about the same price. Unfortunately, source-filter instruments did not seem to be a huge advance on S&S with a simple audition, and S&S had the advantage of a simple and a familiar control metaphor. Physical modeling had limited polyphony, and either preset sounds or sounds with very restricted ranges of variation.

By the twenty-first century, abstracted controls appeared in both hardware synthesizers and computer software. Physical models for electric pianos, strings, guitars, drums and many others became available.

5.4 Analogue modeling

Analogue synthesizers are a mixture of the mathematics (waveforms) with electronic engineering (filters), and underneath, both are just numbers turned into voltages and circuitry. Therefore if physical modeling is complex, then analogue modeling (also known as virtual analogue) is merely a matter of converting analogue circuits into software. And after a slow start, with the Clavia Nord Lead taking the early lead, everyone else seemed to have played catch-up and succeeded. The last years of the twentieth century saw analogue modeling gradually gaining popularity, and the twenty-first century has seen analogue modeling become very widely implemented, with some examples at very low cost indeed.

Low-cost analogue modeling reflects the low entry cost and the excellent support that now exists for programming DSP chips like the Motorola 56000 series. One analogue modeled synthesizer recently cost less to purchase than a mid-range DVD player.

Simple 'two-oscillator, low-pass VCF, twin envelope with VCA and LFO modulating everything' type analogue modeled synthesizers were available in 2003 as synthesizers, as tabletop units, as modules, on small plug-in cards and in software to run on general-purpose computers as plug-ins. In order to find a differentiator, the manufacturers have explored morphing between sounds, adding FM, feedback around the signal path, sample playback, complex modulation routings and controllers, subtle distortion and noise to mimic the limitations of the original analogue circuitry and more. There is considerable attention to the details of implementation.

In 1995, the Clavia Nord Lead provided a very standard analogue monosynth type of synthesizer, but in four-note polyphony and with a distinctive red case. Korg's Prophecy added a number of additional physical models and let the programmer mix analogue modeling with FM with S&S with physical modeling simultaneously, but in a monosynth. Two years later, Korg's Z1 provided the same type of sound generation as the Prophecy, but in a 12-note polyphonic synthesizer. The Z1's architecture allows you to combine sound modules to produce the final sound. The modules include a two VCO, VCF, VCA analogue synthesizer; a comb filter; variable phase modulation, also known as FM; ring modulation; oscillator sync; a resonant filter bank; additive synthesis; an electric piano physical impulsive model; a reed physical continuous model; a plucked string physical impulsive model and a bowed string physical continuous model. Of the major synthesizer manufacturers, Korg seem to have the broadest range of modeling capability in production instruments, and this is probably due to their investment in their Open Architecture SYnthesis System (OASYS) development system, which is the basis for the development of their modeling technologies.

The sounds of analogue modeled instruments are close emulations of analogue synthesizers. The controls are the same, and whilst the early implementations had noticeable stepping or quantization as some of the control knobs altered the modeling values, the 2003 models behave like an analogue. Where things are different, it is in the additions made possible by digital modeling. FM or cross-modulation of analogue VCOs exposes every slight non-linearity or lack of tuning or scaling match, whilst on a modeled synthesizer, the results are predictable and consistent.

There are two very different types of oscillators that are used:

1. Waveform playback, where a sample of the analogue waveform is replayed.
2. Oscillator modeling, where the oscillator itself is modeled mathematically.

The waveform playback is simpler to implement, but suffers from a number of problems: the sample itself is not perfect, and therefore any unwanted noise or frequencies will be pitch-shifted as the waveform is played back at different pitches, which gives a characteristic 'pitched buzz and noise' effect. Oscillator modeling requires more careful study of the source oscillator's fine detail in terms of how it performs when outputting various pitches, but produces more consistent results at different pitches.

Modeled filters have a similar division into 'perfect' mathematical filters that behave as the theory suggests, and modeled filters that reproduce the behaviour of real-world filter circuits. A hybrid technique also exists where a 'perfect' filter is deliberately degraded by a number of techniques:

- Adding noise to the cut-off frequency control, the feedback circuitry or to the resonance control so that the stability of the filter is compromised.
- The resonance is reduced as the cut-off frequency drops to emulate the behavior of some analogue filters.
- The high-frequency response is reduced to mimic the losses in some analogue filter circuits.

Most analogue synthesizers had a resonant low-pass filter, with either a 12- or 2 -dB/octave cut-off slope. By the early 2000s, modeled synthesizers had the capability to model different types of analogue filters from many of the manufacturers of the 1970s, that is, 30 years of progress in a selection from a menu.

Envelopes can also be modeled. Again, the 'perfect' text-book shapes can be markedly different from the reality, and the responses of VCAs to control signals may not be as linear (or exponential) as expected, which can also change the effect of the envelope. The VCAs in analogue synthesizers can also produce distortion. In fact, a detailed examination of an analogue synthesizer will reveal a number of distortions, inaccuracies, variabilities, drifts, slope limits and other characteristics that can affect the final sound and that can be modeled.

It is now clear that modeling represents the same sort of technological leap that the GS1/DX7 did in the early 1980s, when analogue synthesizers were replaced by digital FM-based ones almost at a stroke. But it is not physical modeling that has changed things. Modeling of analogue synthesizers has been the dominant growth area in the early years of the twenty-first century, with true analogue (also sometimes known as 'pure' or 'true' analogue) now seen as an expensive luxury, and physical modeling seen as a very specific solution for producing real-sounding instruments.

The wide adoption and availability of modeling is reflected in the terminology used in commercial synthesizer adverts. In the twenty-first century, modeling has come to mean both the modeling of analogue synthesizers and the physical modeling for specific instruments.

Physical modeling's role could almost be seen as showing that it was possible to use DSP chips to create musical sounds with modeling techniques, and this then opened the way for the modeling of analogue synthesizers on more general-purpose computers.

What is still very curious is that whilst there are many forms of synthesis that could be modeled in software, there are a very large number of examples of the 'classic' analogue synthesizer with two VCOs, a VCF, a VCA, an LFO and two EGs. In contrast, other types of synthesis are much rarer. Software emulations for these are available in all of the popular plug-in formats, for all platforms and many are available for free.

5.5 Granular synthesis

Granular synthesis is regarded as an unusual technique. Unlike many of the other methods of synthesis described so far, it has not been used in commercial hardware synthesizers, although it has been used by some composers working in the academic and research fields. It does not fit into the source and modifier model, but instead approaches the production of sound from a bottom-up point of view, which is very different to most other methods of sound synthesis. But software synthesis has opened up new opportunities for otherwise obscure techniques for making sounds, and granular synthesis is now available as software for use on computers within commercial sound creation programs. Reason, from Propellerhead software in Sweden, is one example of a commercial granular-inspired plug-in.

Granular synthesis builds up sounds from short segments of sounds called 'grains'. In much the same way that many pictures in color magazines are made up from lots of dots, granular synthesis uses the tiny sound fragments to produce sounds. The grains are of very short duration: 10–100 milliseconds, which is close to the 10–50-millisecond timing 'resolution' of the human hearing system audio events which occur closer together than this tend to be heard as one event instead of two. The controls are relatively straightforward; the number of grains in a given time period, their frequency content and their amplitude are the major parameters. The difficulty lies in controlling these parameters: rather like the large number of parameters in additive synthesis, manipulating a large number of grains requires envelopes, function generators and other controllers and can become a very large overhead.

Grains are normally enveloped so that they start and finish at zero amplitude, so that sudden discontinuities are avoided; any sharp change in the resulting waveshape would create lots of additional unwanted harmonics and the result would sound like a series of clicks. Grains may contain single frequencies with specific waveforms, or band-pass filtered noise, and each grain can be different. In some ways, granular synthesis can be considered as the limiting case of wavetable synthesis, where the table of waveforms is swept very rapidly to give a constantly changing waveshape, but few wavetable synthesizers have the control of wavetable selection and the zero-crossing smoothly enveloped grains that are found in granular synthesis. In fact, granular synthesis is normally produced by software, and therefore the grains can be produced using a number of techniques from additive sine waves to filtered noise or even processed samples of real sounds. Some experimenters have worked on coupling granular synthesis with mathematical systems like chaos theory, John Conway's 'life' and fractals (Figure 5.5.1).

Granular synthesis seems to be somewhat analogous to the way that film projectors work. By presenting a series of slightly different still images at a rate that is just about the limit of the eye's response to changes, the impression is one of a smooth continuous movement. In granular synthesis, the rapid

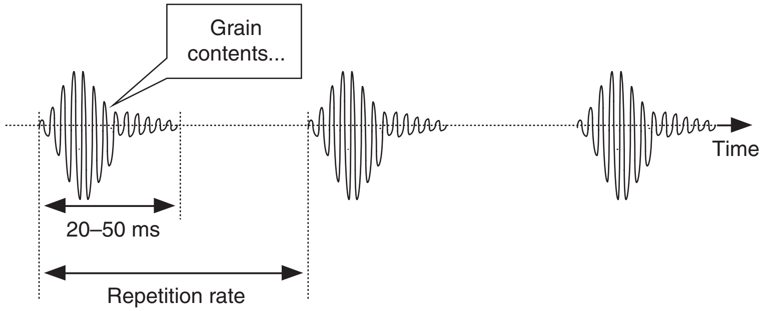


FIGURE 5.5.1 Granular synthesis uses small 'grains': short segments of audio which are arranged in groups. The contents can be waveforms, noise or samples. The major controls include the number of grains, their lengths and their repetition rate.

succession of tiny fragments of spectra combines into an apparently continuously changing spectrum. This constant change of grains is reflected in the timbres that are produced by granular synthesis; words like 'glistening' or 'shimmering' are often used to describe the complex and busy sounds that can result, although the technique is also capable of producing more subtle, detailed sounds too.

As digital synthesizers have become increasingly software-based, granular synthesis has become one of the synthesis techniques that are offered in commercial software-based plug-ins, and maybe the future will see it appearing in real instruments. Despite several attempts to produce a musically and commercially acceptable computer with a music keyboard for stage use, there is still a gap between what can be achieved on a computer and on stage.

It is interesting to note that the granular-inspired 'grain-wave' synth in Reason provides a granular source of waveforms in an S&S type structure, with conventional VCF, VCA, LFO and EGs. The subtractive source-modifier model for synthesis continues to be a powerful metaphor in commercial synthesis.

5.6 FOF and other techniques

Mass-market digital synthesis technology first appeared with the Yamaha DX7 in 1983. After a pause whilst the other manufacturers looked around for other viable methods of digital synthesis, the additive and S&S instruments began to appear. Over the next 10 years, S&S gradually took over until by the early 1990s, it was virtually the only digital method of synthesis. After such a slow and steady development over 10 years, the mid-1990s marked a sudden change when a number of sophisticated instruments were released that could utilise combinations of additive, subtractive and FM synthesis, and these were soon joined by instruments based on physical modeling techniques.

One example of the twenty-first century programmability was the Chameleon from Spanish company Soundart. This was a rack-mounting DSP engine from 2002 that could be configured through MIDI system exclusive dumps or from a PC. It was a general-purpose audio box, and it was completely programmable; it could be an effects unit, a polysynth, a monosynth, amplifier emulation and more. The manufacturer provided extensive support for developers through the Internet, including lots of documentation, including some examples from Motorola on how to program 56000 series DSP chips as sine wave generators, or as 10-band stereo graphic equalizers. There was even a Soundart tutorial on programming a complete monosynth. Soundart seems to have gone out of business in 2005, and the website changed to being run by fans and owners. It seems that innovation and commercial success are not always linked.

It is strange that commercial S&S instruments have not been joined by the large number of techniques that are still used in academic research. Since digital techniques are making it increasingly easy to implement these alternatives, then maybe the problem is the metaphor used for the representation. Analogue modeling has been very successful, perhaps because it has presented exactly the same user interface and programming model as that of the analogue synthesizers 30 years ago.

This section looks at some of the synthesis techniques that may well be incorporated into the digital synthesizers of the near future. They all have a common theme, which is derived from a combination of research into musical sounds, acoustics and human speech and singing. Many are the result of a fusion of the world of telecommunications, computing and music.

5.6.1 Formants

All of these methods are focused around the sounds that are produced by strong resonances, wherever you get a fixed set of 'formant' frequencies (see also Section 2.4.4). The human voice is one example of this sort of system – the mouth, nose and throat can be thought of as a complicated tube-like arrangement where particular frequencies are emphasized whilst others are suppressed, and therefore, the resulting frequency response is a series of peaks. The vocal cords produce a spiky pulse-like waveform that has lots of harmonics in it, and this is then processed by the vocal tract (the mouth, nose and throat) that acts as a filtering mechanism. The result of the filtering is to produce an output that contains predominantly those frequencies, from the original pulse sound, that match the resonant peaks of the filter. Since you can only make minor changes to the physical shape of the tubing formed by the mouth, nose and throat (e.g., changing the size and shape of your mouth cavity with your tongue), then the peaks are mostly fixed, and so what comes out is a set of harmonics that have peaks that are fixed by the formant frequencies, regardless of the pitch of the note being sung! The only things that do change are the fundamental and the underlying harmonics (Figure 5.6.1).

This can be regarded as another type of 'source and modifier' model, where the source is the vocal cords and the modifier is the filter or resonator formed by the mouth, noise and throat. The vocal cords can be emulated by using a short burst of sound whose frequency is fixed and then by triggering this at the rate of the fundamental frequency that you want to produce. The pulse repeats, producing the harmonics associated with the fixed resonances of the formants that it represents, whilst the pitch that you hear is the repetition rate. The modifier part can be emulated by combining several band-pass and notch filters; although since changes of the shape of the 'tube' can happen, these filters need to be dynamically changeable in real time. In fact, the human ear is very sensitive to exactly these changes in formant structure.

Instruments exhibit the same sort of formant structures: the analogy between the human vocal apparatus and some of the woodwind and brass instruments is probably the strongest. The abstraction of a source of sound connected to a 'resonant set of formants' acting as a modifier can be applied to almost any instrument. For string instruments, the formants are determined by the string characteristics, its mountings and the structure of the body of the instrument. For some instruments, other external factors can be very important: an electric guitar is designed to provide a rigid support for the vibrating string, and the heavy wooden body is not a very strong resonant system. But the combination of the guitar string, amplifier, speaker, speaker cabinet and feedback between the acoustic output and the guitar pickups forms a very complex resonant system that is often exploited to great effect in live performance. In contrast, synthesizers and most other amplified musical instruments tend to be used as self-contained systems, and the amplification is merely used to make them louder.

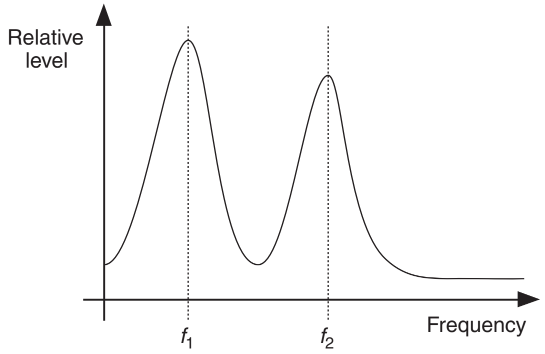
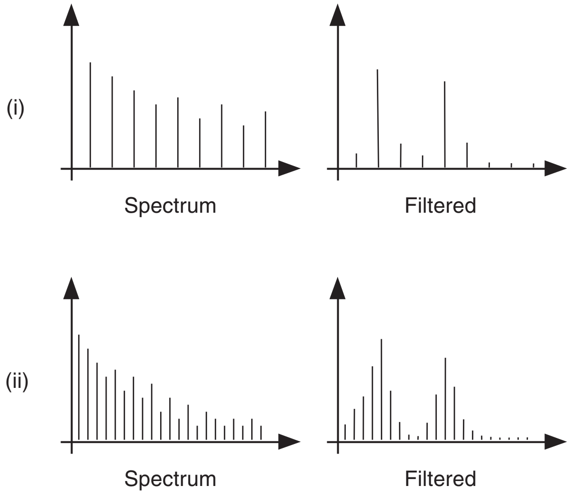


FIGURE 5.6.1
Formants are peaks in the frequency spectrum of a sound. This example shows two large peaks in the output spectrum, regardless of the spectrum or frequency of the input.



5.6.2 Vocoder

Finding more efficient ways to transmit human speech along wires has been one of the major activities of telecommunication research for many years. Most of the raw information content of speech can be found between 300 and 3400Hz, and therefore telephone systems are designed with a bandwidth of about 3 kHz. Frequencies outside of this range add to the clarity and personality of the voice, which is why it is difficult to distinguish between an 's' and an 'f' on the telephone, or why people may sound very different in real life to hearing them over the telephone.

Research at the Bell Telephone Laboratories in New Jersey, USA, in the early 1930s, was looking at how different parts of this 3-kHz bandwidth were used by speech signals. By using band-pass filters, the speech could be split into several separate 'bands' of frequencies, and the contribution of each band to the speech could then be determined. By using an envelope follower, the envelope of the contents of each frequency band could be determined. Once split into these bands, the audio signal could be mixed back together again in different proportions, and even have new envelopes applied to each band. Basic research into the properties of speech yielded results that were interesting (you need the entire 3-kHz bandwidth – removing bands alters the timbre of the speech too radically to be useful for telephony), but they had no practical application at that time. It was not until digital processing techniques became available in the 1960s and 1970s that vocoders were to found reuse in telecommunications.

The band-pass filters are similar to the graphic equalizers that are found in applications as diverse as recording studios and car radios.

But the vocoder proved to be a powerful tool for processing audio signals. By splitting an audio signal into separate bands, analysing the contents and then allowing separate processing of these bands, it allows sophisticated control over the timbre of the sound. More importantly, by separating the analysis and processing functions of the vocoder, it is also able to extract the spectral characteristics of one sound and apply them to another (Figure 5.6.2). The fidelity with which this can happen depends on both the number of bands and the characteristics of the envelope followers. As the bandwidth of the bands decreases, more filters are required to cover the audio spectrum. For 'octave' bands, each covering a doubling of frequency, only eight filters are required – six band-pass, one low-pass and one high-pass. This produces only a coarse indication of the spectral content of the audio signal that is being analyzed and correspondingly the coarse changes to the signal that is being processed. For 'third-octave' bands, 30 or 31 filters are required, and the resulting finer resolution significantly improves the processing quality. The envelope followers determine how quickly the spectrum can be imposed on the processed signal: if the time constant of the envelope follower is too long, then the bands will not accurately follow the changes in the signal that is being analyzed, whilst if the time constant is too short, then the controlling of the amplitude of the bands can become noticeable.

Vocoders began to be used to process musical sounds in the 1950s. The basic vocoder structure had some features that were specific to processing

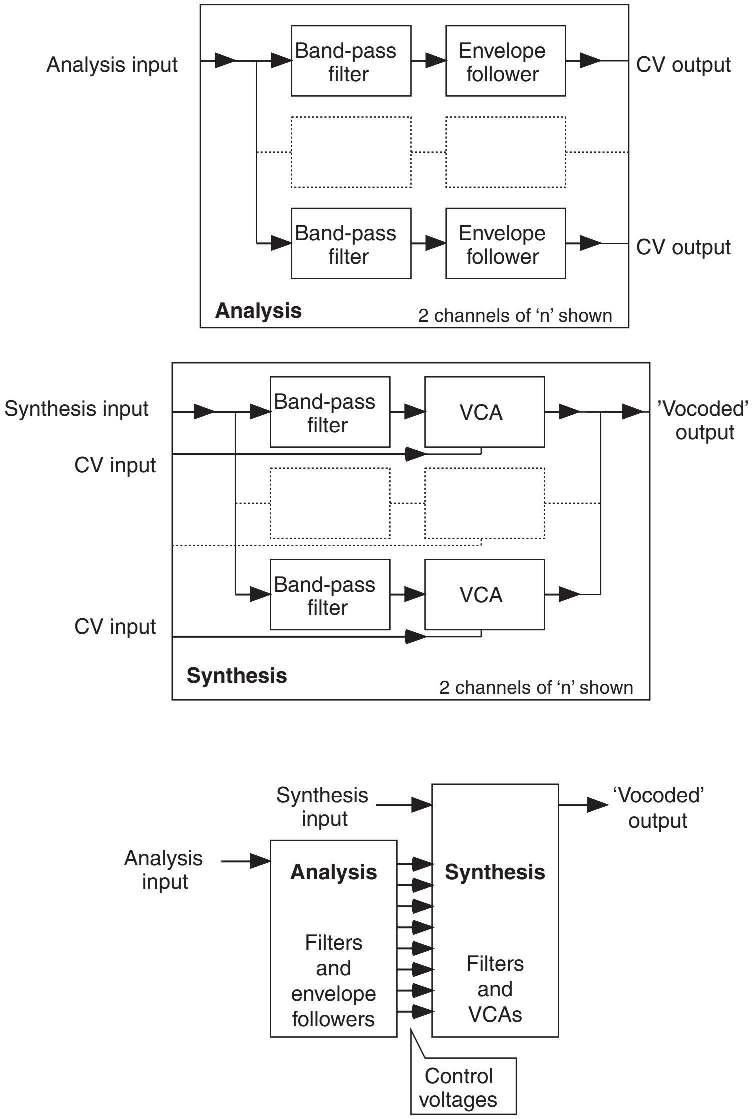


FIGURE 5.6.2 A vocoder is made up of two parts: analysis and synthesis. The analysis section converts the incoming audio signal into frequency bands and produces a CV proportional to the envelope of the contents of that frequency band. The synthesizer section has identical band-pass filtering, but this time it acts on a different audio signal. Each band is controlled by a VCA driven from the analysis section. The characteristics of the analyzed signal are thus superimposed on the synthesized signal. Although this diagram shows analogue blocks, implementing a vocoder is now easier in digital circuitry or on a DSP chip.

speech, most importantly the voiced/unvoiced detection. This determines if the speech sound is produced by the vocal cords or by the noise. Voiced sounds are produced by the vocal cords and modified by the resonant filter formants in the mouth, nose and throat: 'ah', 'ee', 'mm' and 'oh' are examples of voiced sounds. Unvoiced sounds are modifications of noise produced by forcing air through gaps formed by the mouth, tongue, teeth and lips: 'sh' and 'f', 't' and 'puh' are examples of unvoiced sounds. Many vocal sounds are combinations of these two basic types: 'vee', 'kah' and 'bee' have a mixture of noise and voiced parts. The noise tends to be wide-band and therefore can be detected by looking for a simultaneous output in many bands of the analysis filters. In order to produce intelligible speech in the processing section, a noise signal needs to be substituted for the audio signal when an unvoiced sound is detected.

With this emphasis on speech, the first uses of the vocoder were to superimpose the spectrum of speech onto other sounds. The processing requires a harmonically rich source of sound in order to be able to produce good results – using a sine wave will give an output that occurs only when that band is activated by the analysis section, for any other bands there will be no output. The voiced/unvoiced detector can be used as a substitute for noise that is present in the analyzed signal, but this only affects unvoiced sounds, not voiced sounds. Some military communication systems use the minimalistic technique of providing either noise or fixed frequencies in the bands for the processing section. The only information that then needs to be transferred along a communication line is the parameters for the bands and the voiced/unvoiced detection. This results in a very robotic sound that has high intelligibility but almost no personality. Using a vocoder to superimpose the spectral changes of speech onto music instruments has a similar effect – the output has a robotic quality and sounds synthetic. This has been used for producing special effects such as singing pianos, laughing brass instruments and even talking windstorms.

Implementing large numbers of filters in analogue circuitry is expensive, and therefore analogue vocoders tend to have restricted numbers of filters, whereas digital vocoders can have much finer resolution. Digital vocoders can also extract additional information about the audio signals in the bands, and the 'phase vocoder' is one example – it can work with narrow, high-resolution bands and can output both amplitude and phase information, which improves the processing quality and enhances the creative possibilities for altering musical signals.

5.6.3 VOSIM

VOSIM is an abbreviation for VOice SIMulation and uses a simple oscillator to produce a wide range of voice-like and instrumental timbres, although the original intention was to use it for speech synthesis. The original hardware was developed in the 1970s at the University of Utrecht and has since been adapted for software-based digital generation. The oscillator produces asymmetrical waveforms that are made up of repetitions of a series of raised sine-squared waveforms called a 'pulse train'. The series of waveforms reduces in amplitude with time, and therefore, only a small number of parameters are required: the width of the pulses, the decay rate of the amplitude, the number of pulses and the repetition rate of the pulse trains. Because the spectrum that is produced is dependent only on the parameters that control the pulse trains and not on the repetition rate, the harmonic content is independent of the pitch. This is exactly the opposite of a sample playback system and is useful for simulating the fixed formant frequencies that are found in vocal and instrumental sounds (Figure 5.6.3).

The simple controls, versatility and small number of parameters used in VOSIM are ideally suited to the real-time control requirements of a speech synthesis system. In many ways VOSIM has a similar 'minimal parameter' interface to FM, although FM has been commercially successful in musical applications and has only seen limited use as a speech synthesis method, VOSIM is more suited to speech synthesis and has not been used for mass-market musical applications.

5.6.4 FOF

FOF was first developed by Xavier Rodet in Paris in the early 1980s. It is a French acronym for Fonctions d'Onde Formantique, which translates to something like formant-wave-function synthesis, and it is sometimes referred to as formant synthesis. It can be used to produce simulation of vocal-type sounds and incorporates similar frequency splitting elements to vocoding, and the oscillators use a more complicated variation of VOSIM.

The basic idea is to generate each required formant separately and then combine them to form the final output. Each formant 'oscillator' produces an output that deals with just one formant, and instead of having an oscillator and a resonant filter, it combines the effect of the filter on the oscillator output into the oscillator itself. The oscillator produces a series of pulses that are

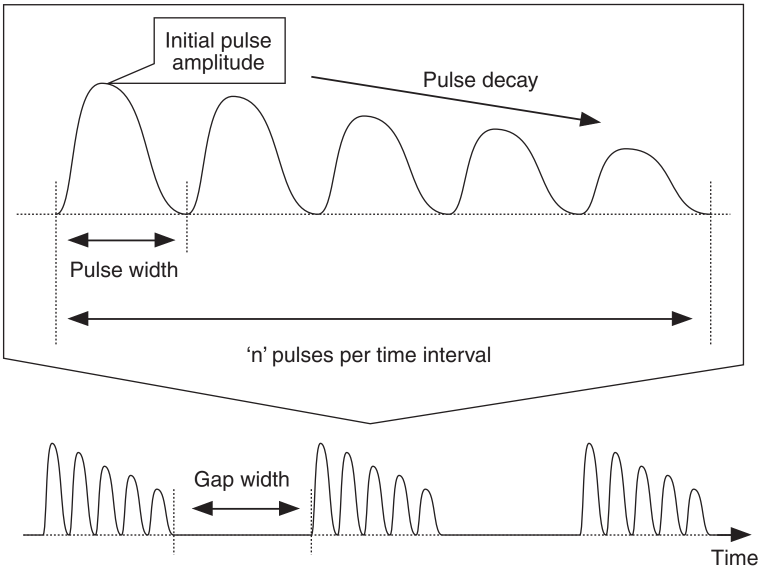


FIGURE 5.6.3
VOSIM produces pulse trains with controllable pulse width, repetition rate, amplitude decay and gap width. It is similar to FOF in some ways.

each the equivalent to what would be the output from the filter if a single rapid step signal was passed through it, called the impulse response of the filter. The pulse contents are thus derived from the impulse response of the filter, and if a series of these pulses is then output, the resulting sound is the same as if the filter was still processing the original step signal. More importantly, the rate of outputting these pulses can alter the frequency of the sound that is produced, but the filtering will remain the same, since it is the shape and contents of the pulse that determine the apparent 'filtering', not the repetition rate of the pulses (Figure 5.6.4).

The output from a typical FOF oscillator is a succession of smoothly enveloped (as in granular synthesis) audio bursts that happen at a repetition rate that is the same as the pitch of the required sound. Each burst of audio has a peak in its spectrum that is the same as the required formant frequency. If the repetition rate is above 25 Hz, then these bursts produce the effect of a single formant with spectral characteristics determined by the audio burst itself. For lower repetition rates, it provides a variant of granular synthesis. Digital implementations of FOF normally provide both FOF and granular modes (FOG), and this allows continuous transformations to be made between vocal imitations and granular textures. Each FOF oscillator produces a single formant, and the output of four or more of these can be combined to produce sounds that have a vocal-type quality.

FOF can be produced using conventional synthesizers by taking a sound that has a fast attack and decay time, with no sustain or release, and then

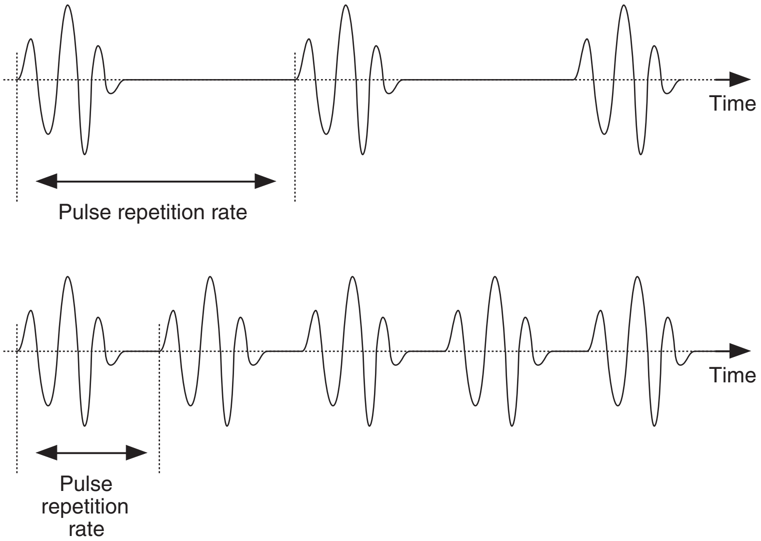


FIGURE 5.6.4 FOF produces pulses whose shape is determined by the impulse function of the sound which is required. The repetition rate determines the frequency of the sound, whilst the pulse contents determine the formants of the sound.

triggering it repeatedly so that it produces a rapid series of short bursts of audio. If the synthesizer produces these short audio bursts at 100 Hz, then the fundamental frequency of the output will be at 100 Hz, but the apparent filtering of the signal will be determined by the contents of the sound itself, and therefore, changing the repetition rate will change only the pitch – the formants (filtering) will remain the same because the sound which is being repeated is also staying the same.

In MIDI terms, this usually means choosing a single note and making a simple and a very short enveloped sound that has the right harmonic content and then sending note on and off messages very rapidly for just that one note, where the repeat rate sets the fundamental frequency, and thus the pitch, of the resulting sound. This is easy to do by creating lots of messages and then changing the tempo of playback! Unfortunately, MIDI is too slow to create high-frequency note repetitions. This limits the maximum frequency that can be generated using this method to monophonic sounds at just under 800Hz under ideal conditions (see also Table 5.3.1).

Producing suitable sounds for FOF involves throwing away some of the instinctive approaches that many sound programmers have. In fact, it is not necessary to use sounds which approximate to the impulse response of a filter; all you need is a quick burst of harmonics. For simulating real instruments and voices, you need to have something which sounds like a single click processed by whatever it is you want to sound like, whilst for synthetic tones almost anything will do.

5.6.5 Dynamic filtering

There are a large number of techniques that utilise the same model of the throat, mouth, nose and vocal cords as the other methods in this section, but which approach the design from the opposite viewpoint. Most were originally developed for use in telecommunication speech coding applications, but they can also often be used to synthesize formant filter-based sounds. One of the best known is LPC, which is an acronym for linear predictive coding. LPC techniques can also be used in resynthesis to help design suitable filters. Other techniques include CELP, PARCOR and the 'Z-plane' dynamic filters used by E-mu, initially in their Morpheus and UltraProteus products, and later in many other products including samplers.

To generalize the dynamic filtering method, a digital filter is used to approximate the formants, and this filter is used to process a source waveform into the desired output. This is very different to extracting the formants and synthesizing them individually, since a single multi-formant filter can produce the equivalent of several separate FOF oscillators simultaneously. The filter shape is controlled by a number of parameters and can usually be changed in real time to emulate the changes which can occur in a real-world resonant system such as the mouth, nose and throat.

5.6.6 Software

For stand-alone instruments, digital synthesis is a combination of digital hardware and software, although strictly there is usually an analogue output stage and low-pass filter connected to the output of the DAC. But it is also possible to use digital synthesis to produce sounds using a general-purpose computer. In this case, the software is normally independent of any hardware constraints – the use of specialized DSP chips to carry out the DSP is often only required to improve the calculation speed. The output of such software is in the form of 'sound files'. Some of the common formats are shown in Table 5.6.1.

The 1996 computer platforms were the Amiga, Atari ST, Macintosh, PC and Unix.

2006 has just the Macintosh, PC and Unix/Linux.

These sound files can be used as the basis for further processing, transferred to samplers for replay or replayed using a computer sound card or built-in audio facilities. It should be noted that in the first edition of this book, in 1996, there were at least five different types of computer platform in general use for music,

Table 5.6.1 File Formats for Sound Files

Suffix	Type	Format
.aif	audio	AIFF
.aifc	audio	AIFF
.aiff	audio	AIFF
.au	audio	μ-law
.au.gsm	audio	GSM μ-law
.avi	data	Intel Video
.gm	data	MIDI
.gmf	data	MIDI
.mid	data	MIDI
.mov	movie	QuickTime
.mp2	audio	MPEG Audio
.qt	movie	QuickTime
.ra	audio	Real Audio
.sds	audio	MIDI Sample Dump Standard
.smf	data	MIDI
.snd	audio	SND: System Resource
.voc	data	SoundBlaster
.wav	audio	WAV
.mod	data	MOD specification
.mp3	audio	MPEG Audio
.asf	data	Streaming format
.dls	data	MIDI DLS

and many file types were restricted to specific platforms. In 2003, there were only three major platforms, and the file formats are almost always usable on any platform.

This software-only synthesis comes in several forms. Commercial software tends to be either simple sample editing programs or sophisticated audio processing software. Freeware and Shareware software is much more varied: ranging from complete digital synthesis systems to sample processing programs, although there is less emphasis on the detailed audio editing that is found in the commercial software.

5.7 Analysis–synthesis

Analysis–synthesis techniques are the basis for the resynthesizer, which takes a sample of a sound, extracts a set of descriptive parameters and then uses these parameters to recreate the sound using a suitable synthesis technique. There are two major problems in achieving this:

1. converting the sample into meaningful parameters
2. choosing a suitable synthesis method.

The conversion is between a sample of a sound and a set of parameters that describe that sound is not straightforward. There is also the issue of mapping those parameters to the chosen synthesis method (Figure 5.7.1).

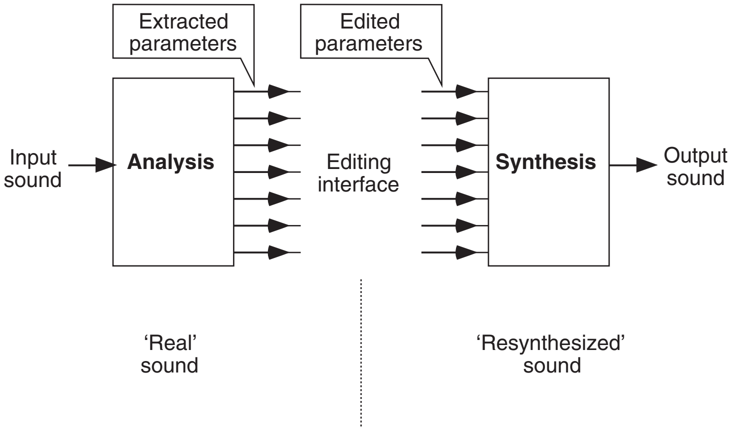


FIGURE 5.7.1 Resynthesis takes an existing sound sample and analyses it to produce a set of parameters. These parameters can then be edited and used to control a synthesizer which produces an edited version of the original sample.

5.7.1 Analysis

The first stage is to analyse the sample. Parameters that might be required to describe the sound adequately to allow subsequent synthesis include the following:

- pitch information
- pitch modulation: LFO and/or envelope
- harmonic structure
- formant structure
- envelope of complete sound
- envelopes of individual harmonics
- relative phase information for individual harmonics
- dynamic changes to any parameter in response to performance controls.

There are a number of techniques that can be employed to produce this information.

Fast Fourier transforms

Fast Fourier transforms (FFTs) are a way of transforming sample data into frequency data, and they are widely used for spectrum analysis. FFTs require considerable computation in order to convert from the time domain (a waveform) into the frequency domain (a spectrum). The detail that can be obtained from an FFT is inversely proportional to the length of the sample that is analyzed. Therefore, short samples have only coarse frequency resolution, whilst long samples have fine resolution – if a sample of 20 milliseconds is converted, then the resolution will be 50 Hz. If the harmonic content of the sample is changing quickly, then a compromise will need to be made between the length of sample that is analyzed and the required frequency resolution. Successive FFTs can move the sample 'window' in time, overlapping the previous sample, and therefore build up detailed spectrum information, even though the majority of the sample data is the same. An alternative approach is to use interpolation between the spectral 'snapshots' (Figure 5.7.2).

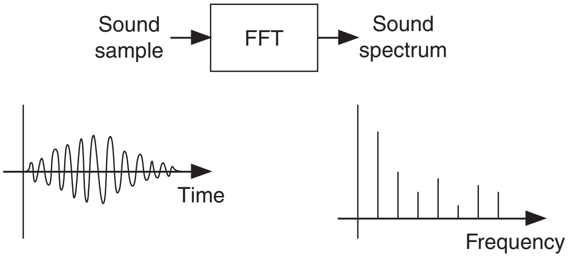


FIGURE 5.7.2 FFTs convert from the time domain to the frequency domain by processing blocks of samples. The larger the block of sample material, the better the resolution of the spectrum: provided that the sample material has a constant.

Linear predictive

Linear predictive methods, derived from speech coding technology, can be used for formant analysis, since they output the parameters that describe a filter that emulates those formants.

Principal Component Analysis

Principal component analysis (PCA) comes from statistical analysis, and it can provide a very simple overview of a complex set of information. It is very useful in finding patterns – outliers, trends, groups and so on – and presenting them to human beings in meaningful ways – diagrams and graphs instead of pages of numbers.

PCA is normally described in mathematical terms, but is easy to grasp with a simple example. Suppose we take all the people in the United Nations Council Chamber in New York and try to divide them into groups. We could try some obvious differentiators like gender (two main values) or nationality, but what would be really useful would be to know what the definitive way of telling all these people apart from each other.

PCA would do this by taking all of the available information about the people and plotting it in a multi-dimensional space. For a simple approximation, we can use gender, age and nationality, which gives us a 3D cube where we can plot each person. If we then examine the cube we will see that the gender shows two clusters of values (male and female), whilst the nationality has a larger number of clusters (people from the same country), and the age has a more or less continuous distribution of ages in the adult range. PCA looks for the biggest range of variations that are represented by the most examples, and therefore here, 'age' meets those criteria. The principal component is thus age, followed by nationality and then gender.

In musical terms, PCA allows information to be pulled apart into useful parts and then used as the components for synthesis. Example applications could be the following:

- Extract the wavetables for a sound so that the timbral changes in the sound can be emulated by changing wavetable.
- Extract a different set of wavetables that could be used in an additive synthesizer, where the basic tone is the first waveform, and additional harmonics are added by the second waveform and so on.
- Extract two spectral plots of the extremes of the timbral change in the sound, and then allow dynamic blending from one spectrum to the other (known as cross-synthesis).

PCA is a general-purpose analysis tool that can be used in a wide variety of ways in a number of musical applications.

Pitch Extraction

Pitch extraction employs a number of techniques in order to determine the pitch of a sampled sound. Because the perceived pitch of a sound is concerned more with the periodicity rather than the frequency of the fundamental, pitch extraction can be difficult. Methods include the following:

- *Zero-crossing*: The simplest is to count the number of zero-crossings, but this is prone to errors because of harmonics causing additional zero-crossings. Filtering the sample sound to remove harmonics and then counting the zero-crossings can be more successful, but a better technique is to use the peaks of the filtered sample since the harmonics have been removed and a simple sine-like waveform is all that should be left after the filtering. This method has problems when the fundamental frequency is weak, since filtering the harmonics still leaves a noisy, low-level signal.
- *Auto-correlation*: Auto-correlation is a technique that compares the waveform with a time-delayed version of itself and looks for a match over several cycles. When a delay equal to the periodicity of the waveform is reached, then the two waveshapes will match. This assumes that the sample sound does not change rapidly and that there are no beat frequencies or large inharmonics.
- *Spectral interpretation*: Spectrum plots derived from FFTs can be used to determine the pitch. The spectrum is examined and the lowest common divisor for the harmonics shown is calculated. For example, if harmonics at 500, 600, 1000 and 1200 Hz were present, then the fundamental frequency would probably be 100 Hz. Again, beat frequencies and large inharmonics can produce significant errors with this technique, normally producing fundamental frequencies that are too low (a few or tens of hertz).
- *Cepstral analysis*: By further processing the spectrum, it is possible to produce plots that quite clearly show peaks for the fundamental frequencies. The process involves converting the amplitude axis of the spectrum into a decibel or logarithmic representation instead of the normal linear form and then calculating the spectrum of this new shape, that is, using an FFT to treat the spectrum as if it is a waveform! The resulting 'cepstrum' (a reworking of the word 'spectrum') will show a peak in the upper part of the time or 'frequency' axis that indicates the fundamental frequency of the sound. The cepstrum merely indicates the underlying spacing of the harmonics shown in the spectrum, and therefore, spectra with only odd harmonics or very sparse harmonics (like a sine wave!) can be difficult to interpret because of processing artifacts that may obscure the important information (Figure 5.7.3).

Envelope following

Extracting the envelope from a sample sound is relatively straightforward in comparison to pitch extraction. The sample sound is low-pass filtered, and then a 'leaky' peak detector is used to produce a simple curve that approximates to the original volume envelope. The setting of the low-pass filtering and

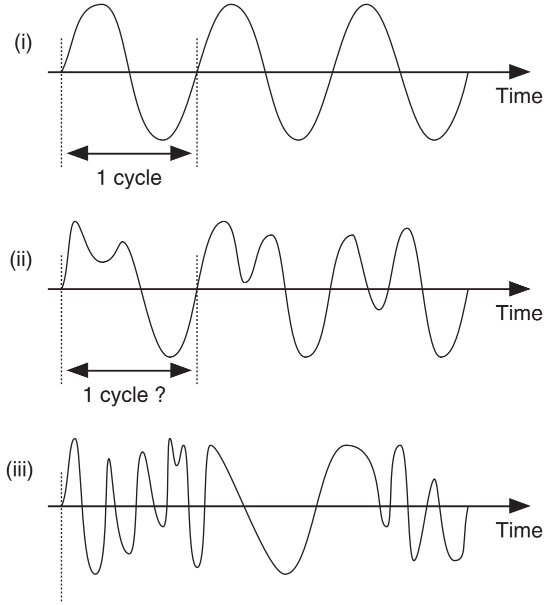


FIGURE 5.7.3 Pitch extraction needs to be able to cope with a range of inputs: from simple sine waves (i) which can be processed by a zero-crossing method; through waveforms which change slightly from cycle to cycle (ii) where auto-correlation or cepstral analysis can produce useful pitch outputs; and finally noise (iii) where the pitch extractor should indicate that it is noise rather than a rapidly changing pitch. Although the human ear can readily achieve this, the process is less straightforward for electronics and computers.{link}

the peak detector decay time constant govern the effectiveness of the envelope detection. The low-pass filter should be set so that its cut-off frequency is lower than the lowest expected frequency in the input sample, but setting it too low can slow down the response of the envelope, resulting in slow attack, decay or release times.

Additional parameters

Pitch and formant analysis may also produce outputs that change with time, and therefore these may need to be converted into envelope format. Pitch modulation is likely to be in two parts – cyclic modulation (vibrato) and time-varying (pitch bending) – and therefore further processing may need to be employed to separate these two parts.

In order to produce a realistic sound from a resynthesizer, it is not sufficient to take a single sample of the instrument sound and analyse it. The characteristics of the sound that is being analyzed may change under the influence of external parameters used in performance or when different notes are played. There is thus a need to take into account any changes caused by performance controls and different playing pitches. One example is the change in timbre

Some sounds require interactions between notes to be taken into account. For example, the sympathetic vibrations that are set up in other strings on a piano when a note is played.

that happens when an instrument is played harder or more vigorously – hitting a piano key harder or bowing a string with more pressure. Other examples include damping strings or muting a brass instrument. Several samples will be required in order to measure the dynamic changes to parameters that occur in response to these performance controls. Different pitches can be dealt with by making several samples of the instrument throughout its playable range. The outputs of these dynamic measurements can then be interpolated to give approximations for all notes and performance control settings.

5.7.2 Synthesis

Almost any synthesis technique could be a candidate for the synthesis 'engine' for a resynthesizer. The most important consideration is how the parameters of the technique map to the parameters that can be extracted from the sample. The mapping needs to be complete and unambiguous, but it also needs to produce a parameter set that can be manipulated by the end user of the resynthesizer.

Additive

Additive synthesis appears to offer perhaps the simplest approach to resynthesizing sounds from parameters. The only parameters that are required are detailed pitch, amplitude and perhaps phase information for each of the harmonics that are present in the sample sound. Unfortunately, this is likely to be a large number of harmonics, each with complicated multi-stage envelopes for the changes in the pitch, amplitude and phase parameters with time and performance controller settings. Therefore, although the extraction of the parameters is relatively straightforward, presenting them to the end user in a manageable form is more difficult.

Analysis–synthesis using sine waves is often abbreviated to A/S.

In 1999, Xavier Rodet, at IRCAM in Paris, published a paper describing SINOLA, which uses a measure of the peaks in a complex spectrum as the analysis part and combines additive synthesis with sine waves and wavetable synthesis for the synthesis part. Work at IRCAM on analysis–synthesis techniques still continues.

FM

This modulation has a much smaller set of required parameters than additive synthesis. In this case, the problem is how to convert the extracted parameter information about pitch, amplitude and phase for each harmonic into suitable parameters to control FM. There is no simple way to work backwards from a sound to calculate the FM parameters that produced it – a process called deconvolution. An iterative process that tests possible solutions against the given parameters might be successful, but it is likely to require considerable processing power as well as time.

Subtractive

Subtractive synthesis requires more parameters than FM, but it provides a smaller set of controls than additive synthesis. The major problems with using subtractive synthesis are the fundamental limitations of the technique – the filtering is often a simple resonant low-pass filter; and there is a limited set of source waveforms. The combination of these problems means that subtractive synthesis has a very limited set of possible sounds, and this seriously restricts the possibility of being able to resynthesize a given sound.

Formant

Formant synthesis techniques such as FOF and VOSIM have small numbers of parameters, and the conceptual model is similar to subtractive synthesis. But unlike subtractive synthesis, formant synthesis techniques are not restricted to simple filtering, but can recreate complex and changing formant structures. Although the source waveforms may be simple to control, the dynamic formant filter presents a considerable problem to a user interface designer.

In fact, FOF is part of a complete software package called CHANT, written at IRCAM in Paris by Xavier Rodet and others in the early 1980s. CHANT can be used to analyse a sampled sound and extract the harmonic peaks and then use these formants as the basis of an FOF resynthesis of the sound.

Physical modeling

Physical modeling can be considered to be a type of analysis–synthesis technique, although the analysis process is more sophisticated since it involves a study of the physics of the instrument and its sound and then the building up of a physical model of that instrument. The synthesis part is then relatively simple – just run the model to simulate the instrument's behavior. At the moment, the process of analysing a real instrument is a time-consuming one, although the commercial development of physical modeling may facilitate the development of software tools for this task.

5.7.3 Resynthesis

Any resynthesis technique requires a compromise between the depth of required detail to describe the original sound and the ability of the user to make meaningful changes to the sound. There are two types of editing methods that can be used to control the resynthesis of a sound:

1. *Extracted parameters*: Editing the transforms that are used to map the extracted parameters to the synthesizer parameters. This requires a good knowledge of the analysis technique.
2. *Synthesizer parameters*: Editing the synthesizer parameters. This only requires knowledge of how the synthesizer produces sounds.

Because analysis–synthesis techniques tend to produce information on the spectrum of the input sound during specific time windows, then the conversion of the extracted parameters into continuous controls for the synthesizer tends to be iterative. The process requires knowledge of the synthesis technique – specifically the way that the spectrum can be controlled. The analysis output is then matched to possible ways to recreate that spectrum using the synthesizer. The iteration should ideally converge on a small number of possible solutions.

With enough parameters, it should be possible to resynthesize a specific sound very accurately, but it may not be possible for a user to make any useful changes to that sound because of the complexity of the controls and the number of parameters.

Because software can cope with large amounts of data easily and quickly, whereas complex mathematical processing often involves additional time, the two techniques that seem to offer the best resynthesis engine are additive and FOF/VOSIM. In both cases, the software would need to present some sort of abstracted user interface to the synthesis engine to avoid displaying all of the parameters. Commercial resynthesizers have not been very successful. Although the idea has been talked about for a long time, only a few minor manufacturers have attempted to produce a resynthesizer. Few have succeeded in combining a practical user interface, rapid analysis and a versatile synthesis engine at a reasonable cost.

In 2003, Hartmann Music released the Neuron Resynthesizer. The Neuron was actually in two parts: the stand-alone PC-based keyboard hardware that used modeling technology to replay the sounds and the software called ModelMaker that ran on a separate computer and allows the user to work with audio files to produce the models used by the Neuron. There were 10 underlying types of physical model including bowed strings, plucked strings, pianos, woodwinds and so on. The user selected a suitable (or unsuitable!) model, and ModelMaker then produced a new set of driver and resonator specifications that could be downloaded to the Neuron and played in just the same way as the factory-supplied models. The 'resyn'thesis oscill'ators' were called 'resynators', and they had two major groupings of parameters, namely 'scape' (driver or source) and 'sphere' (resonator or filter). These sound sources were followed by a complex set of mixing, panning, modulation, effects and filters with unusual naming conventions (and called 'silver'), that led to the 5.1 surround-sound output. The resynators provided parameters which could be used to control the driver and the resonator parts of the model, but as with many modeling-based synthesizers, the mapping of parameters to the changes they make to the sound was not always straightforward. The Neuron also used a number of unusual wheel- and stick-based front panel controls, which gave it a distinctive appearance.

Hartmann produced the hardware for the Neuron, but the software algorithms it used were developed by Prosoniq, a company which uses the software-based adaptive learning processes called neural networks to provide sophisticated and innovative audio capabilities. This has enabled them to produce a number of advanced audio processing software applications and plug-ins. Prosoniq called the Neuron's audio analysis technique 'Multiple Component Feature Extraction', and it provided information about the spectral evolution in time of the amplitudes, phases and frequencies of the frequencies in the audio signal. This was probably achieved using PCA as described in Section 5.7.1.

For the replay of the sounds, Prosoniq used what they called 'audio rendering', which appeared to consist of a number of techniques including wavelets and modeling, but which was optimized for the particular model being played, again probably by using PCA to determine which was the optimum technique. This novel approach seems to be rather like having a synthesizer that configures itself as an FM synthesizer for gongs and as an S&S synthesizer for piano sounds.

The Neuron used unfamiliar metaphors and a complex user interface and it seemed to be powerful and flexible, and as with many leading-edge synthesizers, it was undeniably expensive. The learning curve was increased by the unfamiliar naming conventions used, which made it difficult to assess exactly how truly innovative it was in comparison to other modeling-based instruments. The Neuron seems to be a good example of the difficulty of achieving the right mix of capabilities, metaphors and presentation in a resynthesizer. As with many new synthesis techniques, the true mark of success might only occur with the second or third iteration; as with Yamaha's FM synthesis, where the DX1 was described in very similar words to those at the start of this paragraph, it was not until the DX7 that FM found broad appeal and success.

Unfortunately, commercial difficulties related to the manufacture of the hardware led to Hartmann going into liquidation in 2005. The purchasers of Neurons congregated on an Internet forum called SurroundSFX, which also had a Prosoniq forum, and the forum is still active. The future of the Neuron is uncertain.

5.8 Hybrid techniques

With a wealth of powerful techniques becoming available, digital synthesis has increasingly used software-based methods. Instruments are gradually relying less on a specific technology and more on a mixture or combination of synthesis techniques. This provides a wide range of sounds and avoids any specific limitations of a particular technique.

One example is the FM synthesis implementation found in the first generation of Yamaha instruments such as the DX7 – the 'weak' areas include rich string or pad sounds, as well as filter sweeps. By combining more than one synthesis method, there is also scope for producing sounds that are not possible using any of the separate methods in isolation.

5.8.1 Examples

- The Yamaha SY99 and SY77 mix together FM (AFM or advanced FM) and AWM2 (advanced wave modulation 2), which makes the most of FM's flexibility and S&S's realism, and adds resonant filtering. By allowing the S&S waveform to modulate the FM operators, the S&S

sound can be processed as part of the FM synthesis. Yamaha call this real-time convolution modulation or RCM. FM with non-sine-shaped waveforms produces lots of harmonics, and RCM is useful for adding harmonics and then removing them using the digital filtering. This is an underexploited technique – few of the sounds produced on the SY99 and SY77 make use of RCM.

- Yamaha's S&S instruments have a plug-in card architecture that allows the addition of physical modeling as per the VL-series, or analogue modeling as per the AN-series.
- Korg's Prophecy mixes several digital techniques to give a sophisticated monophonic 'lead-line' instrument that has a very 'analogue' feel to some of its sounds. It provides a conventional 'analogue' synthesis emulation; FM, physical modeling of brass, reed and plucked instruments; and three variations on sync/cross-modulation and ring modulation analogue emulations. To control these methods, it has a wide range of performance controllers.
- Korg's Z1 extended the Prophecy's mix of synthesis to a polyphonic version, and is available as a plug-in card for Korg's S&S instruments.
- Technics' WSA1 mixed bits of 'physical modeling' with S&S to give a simplified 'driver and resonator', source-filter synthesis instrument that had the advantage of being polyphonic at a time when other physical modeling instruments are monophonic or duophonic. It was not followed by any further models.
- Kurzweil's variable architecture synthesis technique (VAST) provides many resources, but they are more like a modular approach to an S&S synthesizer than any combination of separate synthesis techniques.
- Roland has mixed sophisticated S&S technology with sampling in the Fantom-S workstation.
- Propellerhead's Reason is a combination of a sequencer, synthesizer modules, drum machine and effects units, but implemented in software. The synthesizers include wavetable and analogue modeling, plus a granular synthesizer.
- Native Instruments' Reaktor is a software S&S synthesizer, sampler, granular resynthesizer, effects and more. Running on Mac or PC, it provides powerful soft synthesis capability, and there are hundreds of instrument definitions (ensembles) available to download.

The future seems to lie with a combination of techniques, since none of the available methods offers a complete solution. As processing hardware becomes more powerful, the software functionality increases and also becomes more flexible. The limits are more likely to be the user interface and the processing power, rather than the synthesis methods.

Future synthesizers are likely to be general-purpose synthesis engines that can be configured to produce a number of different techniques, although it is unlikely that any standardised way of controlling these techniques will emerge in the near future. This means that even though the synthesis methods will converge, the user interfaces and sound storage formats will not. The commercial model for producing this type of general-purpose synthesis instrument is not clear, and it may be that the internal construction is common, whilst the external appearance may be very different. For hardware, there have been a number of commercial problems around making general-purpose platforms for audio and music, some examples of which include Soundart's Chameleon, Creamware's Noah and Hartmann's Neuron.

There have been a number of examples of generic DSP-based audio engines in rack-mount units: Soundart's Chameleon (discontinued); Creamware's Noah (discontinued); Manifold Lab's Plugzilla (website last updated in 2007) and Symbolic Sound Corporation's Kyma (still active).

Hybrid instruments are thus similar to the pre-MIDI analogue instruments – 'closed' systems where interconnecting synthesizers were not possible without sophisticated hardware. With complex software-based synthesis, the possibilities for interfacing become more remote which is very useful for commercial synthesizer manufacturers, but not as good for users.

5.9 Topology

One of the interesting things about the development of the internal topology of synthesizers from analogue to digital is that by the time you get to digital synthesizers, the restrictions are not there any longer. Even more interesting is the way that computer-based software solutions have imposed their own topologies in order to provide a framework for the complexity that they provide.

Digital synthesizers have less constraints than samplers. Samplers have a very specific job to do and do that function well, but it does not intrinsically require more flexibility than that described for the hybrid S&S described in Section 4.6.

But some digital synthesis allows considerable reconfiguration. FM synthesis uses operators arranged in a large number of configurations called algorithms, and these change the roles of some of the operators from carriers to modulators, as well as their position in stacks of operators. This is far more radical than an S&S synthesizer offering flexible use of elements of two parallel sound making paths. Digital modular synths provide the same topological freedom as their older analogue ancestors, although with less cables and about the same potential for confusion and making patches read-only.

5.10 Implementations

Yamaha's FM was one of the first all-digital synthesizers to see commercial success, and its development saw racks of transistor–transistor logic (TTL) chips from the prototype compressed down into just a few 'ASIC' application specific integrated circuits for the final hardware. The rest of the 1980s and the early 1990s saw an increasing exploitation of this 'make your own chip' technology, and this continued into the twenty-first century. But DSP chips also began to leave the laboratory and move into effects units and then synthesizers, and both ASICs and DSPs are used in many designs.

CD technology has been exploited by the output circuits of synthesizers and samplers, and as over-sampling and higher numbers of bits have become available, these have been included in synthesizer output stages. AES/SPDIF, mLAN and other digital audio output formats have been slower in adoption and tend to be offered as options on only the most expensive and studio-oriented equipment.

The design time for many digital synthesizers, samplers and other musical equipment does seem to be longer than computers, because there does seem to be a time-lag before storage media are included, and correspondingly a shorter time before obsolescence kicks in. For example, the storage cards used in one synthesizer manufacturer's products were recently declared obsolete whilst the equipment was still on sale. This is not a new phenomenon – the Akai S612 sampler used 2.8-inch QuickDisk floppy disks at a time when the Sony 3.5-inch floppy was still one of a number of contenders.

One implementation detail that has interesting consequences, particularly when compared to computer software, is the operating system software. Digital synthesizers and samplers normally use embedded computers to provide the control over the hardware and to provide the user interface. The software that runs on the embedded computer processor is likely to be in either code specific to the processor (known as assembler or machine code) or in an intermediate programming language such as C, and both these place limitations on the sophistication of the user interface that can be provided particularly since the design time for synthesizers needs to be short, and once launched, hardware with embedded computers is normally not updated. Up until the late 1990s, the code produced by the design team for a synthesizer or a sampler would be burnt into ROM chips and placed in the hardware, and would only be updated when the hardware was serviced or when the purchaser complained about a bug, and the ROMs would be replaced. By the late 1990s, reprogrammable memory was beginning to be used, albeit slowly, in some devices, initially mostly to replace battery-backed random-access memory (RAM), but using it to enable operating systems to be altered or updated did not become widespread until the twenty-first century. Even with this capability, the number of versions of embedded operating systems in most digital synthesizers and samplers is very low, often only in single digits.

5.11 Digital samplers

A sampler is the name given to a piece of electronic musical equipment that records a sound, stores it and then replays it on demand. There are thus three important functions:

1. record the sound
2. store the recording on some sort of storage medium
3. replay the stored sound.

A sampler combines all of these functions into one unit, and this makes it very different from almost all of the other examples of synthesizers described in this book. Most synthesizers can fulfil the last two functions – store and replay but the distinguishing feature of a sampler is its ability to record sounds.

This definition of a sampler in terms of its functionality is important because it enables a wide range of equipment to be classified as being samplers, whereas the commonly used term is often restricted to merely electronic music equipments that store sounds in RAM. Using the functional description, the following can all be described as samplers:

- tape recorder
- cassette recorder
- video recorders
- personal video recorders (PVRs, e.g., Sky+)
- digital audio tape (DAT) recorder
- digital optical recorder (MiniDisc, CD-R, and so on.)
- MP3 recorder/player (e.g., iTunes plus iPod)
- echo effects unit
- music samplers
- computers with sound input and output facilities.

All of these 'samplers' represent ways to record, store and subsequently replay sounds. In some of these cases, the sounds will probably be naturally occurring sounds that can be recorded with a microphone, but this does not prevent the process of collecting the sounds, storing and manipulating them, and then replaying them from being called 'sound synthesis'. Within this wider context, any of the techniques that have already been described can become part of a larger synthesis system by utilizing sampling. The 'source and modifier' model can be used to describe the working of an analogue subtractive synthesizer, but it can also be used to describe the process of using a synthesizer merely as the source of sounds that are then recorded, stored, modified and finally replayed using a sampler that acts as the modifier of those sounds.

Samplers thus form a bridge between the analogue and the digital synthesizer, since they span the two technologies with very similar instruments. Analogue sampling can be tape-based or chip-based, although analogue sound storage chips have been largely ignored since digital technology became available. Digital sampling has increasingly used the technology and approach of synthesis, and this has led to the convergence of sampling and synthesis.

One popular usage of the word 'sampler' that is not covered by this type of definition is the recorded collections of material from more than one source, which are also called samplers.

5.11.1 Digital sampling background

Digital sampling is based on three electronic devices:

1. analogue-to-digital converters (ADCs)
2. memory devices (RAM, flash electrically programmable read-only memory (EPROM) and so on.)
3. digital-to-analogue converters (DACs).

These three devices carry out the three major sampling functions:

- the ADC records the sound
- the memory devices store the recording
- the DAC replays the stored sound.

Before the early 2000s, hardware samplers were the primary way in which digital sampling took place, with computers being used for some editing tasks and perhaps for backing up sample sets. But by the mid-2000s, the hardware sampler had been largely replaced by computer based samplers that actually combine sample replay with sequencing (often MIDI sequencing too). But just as a reader of this book may be asked to use old analogue synthesizers, the same is true for hardware samplers, and therefore this chapter attempts to cover both hardware and computer-based samplers.

A sampler works in three modes: record, edit/store and replay. The record mode is used to convert signals from a continuous analogue form into a numeric digital representation. The digital data that represents the sound is then held in RAM memory inside the sampler, and this is edited in the second mode, edit/store. When an audio signal is recorded by a sampler, the start of recording is normally set to before the actual start of the sound, so that the initial attack part of the sound is captured. Once in RAM memory inside the sampler, the sample data needs to be edited so that the start of the sound is at the start of the sample data. This ensures that when the sample is replayed, it will start playing without any time delay. Once edited, the sample data is then stored in some sort of permanent storage such as a hard disk. This sample data can be reloaded into the sampler's RAM memory when it is required. The replay mode takes the sample data in the sampler memory and converts it back to an analogue audio signal.

The major division between an S&S instrument and a sampler can be considered to be the type of memory; S&S instruments use fixed ROM memory and so the samples cannot be edited, whilst samplers use volatile RAM memory where the samples can be edited.

The actual process of sampling sounds is often forgotten because of the vast range of pre-recorded material that has become available. But at the heart of almost all samplers is the capability to record sounds.

Before actually making the recordings, it can be useful to plan out the samples that will be required:

- How many pitches should be sampled? (Minimum of one, maximum limited by the range of the sound source or the 128 MIDI note numbers.)
- How many levels of intonation or variation should be sampled? (What performance variations can be mapped to velocity or other controllers? How is the intonation going to be measured so that the individual samples for each pitch have similar levels of intonation overall?)
- How many 'takes' should be recorded? (How confident are you that you will be able to capture all of the sample source material that you will

need for getting a smooth multi-sample set, good loops and consistent intonation changes? Remember that it may be very difficult to go back and record additional samples under exactly the same recording conditions!)

- Are there any associated sounds that should be sampled? (For example, slapping the body of an acoustic guitar, the noise made by fingers sliding along wire-wound strings, fret buzz and hammer-on noise.)

Care should also be taken to ensure that the recording process is matched to the sampler with respect to any metering and headroom. The metering should be set so that the sources of sounds are not always as apparent as might be imagined. Some possibilities are given in detail.

Singing

Recording a vocal line into a sampler can be useful in several ways. If recorded early in the song creation process, then it can provide a guide vocal for building up the arrangement, or for working on vocal harmonies, or it can serve as a baseline recording for later improvement by the singer. As with any recording or composition process, the 'sleep-on-it' test can be harsh, but very useful – a sound, vocal performance or song that sounds perfect 1 day can seem rather less perfect the following day when it is auditioned again.

Real instruments

Sampling real instruments is a difficult and an exacting task that requires skilled performance ability, determination, patience and time. Unless the instrument is not readily available already as samples, then this process is not recommended. Trying to record clean, correctly pitched samples of different notes with similar intonations at similar volumes and with smooth transitions between multi-samples across the keyboard range is much harder than it sounds, and looping those sounds so that they have inaudible looping artifacts can be very challenging.

Other electronic musical instruments

Sampling other electronic musical instruments is easier than real instruments. The pitching is more repeatable in most cases, the noise floor and the maximum output level (MOL) set the limits to the dynamic range and by using a sequencer or special-purpose software to send consistent MIDI velocity values, then the intonation can be made consistent. Assembling large stacks of keyboards and lots of effects units can indeed produce very big and complex sounds, but these are not always useful in all musical contexts. Using factory presets sounds on some of the keyboards is not a good idea, which means that the sounds all need to be custom programmed in order to ensure that the samples are unique and that the factory presets are not recognizable. Whenever a factory programmer produces a complex and a clever special effect sound from a new synthesizer it will automatically become well known and almost unusable in a real performance context. Producing sounds which are distinctive and useful is considerably more challenging.

Real world

The real world is an astonishing source of sounds, but these present great challenges when they are going to be reproduced by a sampler. Wind noise, aircraft noise and general background noises can be either unwanted distractions or the sounds being recorded, but usually the former. Sounds that are pitched are unlikely to be tuned to a note based on A-440, and changing the pitch of many real-world sounds can completely destroy their characteristic timbre. The triangle is one example a pitch-shifted triangle sounds nothing like a triangle. The usual technique when capturing real-world sounds is to use a portable DAT/hard disk/flash drive recorder and a detailed notebook. Making a safety backup copy of the source recording immediately before starting any sampling editing is strongly recommended.

Real-world sounds are sometimes found in unexpected places. The author had an ancient oven where the grill door hinge required lubrication, but it had not been lubricated because the sound it made when it was opened was very similar to the sound made by Klingon ships when they fire their main phase disruptor weapon. Sadly, the oven has now been replaced by a modern, quieter version.

CD-ROMs

Pre-recorded samples on compact disk-ROMs (CD-ROMs) are a popular and potentially expensive source of sounds. Because the sounds have already been sampled, looped and assigned to notes on the keyboard with smooth transitions between the multi-samples, then these are an almost ideal source of sounds that other people have produced. If the required sounds are not available on CD-ROM, then they are less than ideal.

CDs

Many samplers include facilities which are designed to ease the recording of sounds from this type of media, although these facilities are normally intended for use with sample CDs, where the sounds are presented in sequence of pitch, intonation, and so on. Sampling audio CDs other than specially licensed sample CDs will require permission from the copyright owner.

DVDs

Movies are noted for sound-bites short pithy phrases or sentences that capture a mood or express an emotion. Sampling these from movie soundtracks also requires permission from the copyright holder, although there are a number of ways of producing close emulations of the originals, ranging from actors and actresses who specialize in sound-alikes, through to specialist sample CDs.

Once the sample has been recorded, then it will probably need to be edited …

5.12 Editing

The most important editing function that is required by a sampler is the normalization of the level of the samples. The recording process will introduce variations in the level of samples, particularly when several 'takes' have been recorded. This means that the sample with the largest individual sample values will need to be located the 'loudest'. If this sample is very close to the limits of the recording process, then it should be close to the maximum limits of the sampler. If too much headroom has been allowed in the recording process, then this sample may be considerably lower than the maximum limits. Comparing this sample against pre-existing 'factory' samples should show if the level needs adjusting. Once the level of this sample has been set to its final value, then the other samples need to be compared to it and adjusted accordingly. The final result should be a set of raw samples with consistent apparent loudness across the available pitch range of the sound, and across the levels of intonation.

The next most important editing function is the trimming of the unwanted portions of the raw samples – 'before' and 'after' the wanted sample. This trimming or 'topping and tailing' process allows the sampler user to set the start and the end of the sound. This can be especially important if the raw sample is noisy, because the start of the sound may not be apparent, and a compromise has to be made between finding the true start of the sample and hearing some noise at the start of the sample. Listening to a sample in isolation may not be the best way to determine if this noise is intrusive, and it may be best to wait until enough of the samples are available to be played, and then to audition the sound, before making a decision to go back and read just the samples. Some samplers provide automatic functions that will trim a sample using criteria that be adjusted to suit the user. Although this trimming function is of great importance to a user who produces their own samples, the majority of sampler users merely use the sampler to replay pre-prepared samples, and therefore the trimming function is not as important as might be supposed. But the ability to manipulate segments of audio is essential for the user who wishes to use a sampler as a synthesizer rather than merely a sample-replay device.

With this in mind, it is not surprising that in the 1990s, the long-term focus moved from the sampler hardware to the editing software. The sampler thus became a box that records sounds for subsequent editing on a computer, and then receives the edited sounds and replays them. By the mid-2000s, the computer itself could carry out all the required sampling functionality, and the hardware synthesizer more or less vanished.

Many S&S synthesizers have only limited sample manipulation facilities and no sample editing facilities – the sample are in ROM memory, and therefore the only manipulations that are possible are changes in direction, start point and loop points. S&S instrument thus rely almost entirely on their synthesizer modifier section to make changes to the timbre of the samples. In contrast, samplers normally have a powerful sample manipulation and editing section as well as a synthesizer-type modifier section. The synthesis modifier facilities are thus less critical to the operation of the sampler, and in fact, many samples of filter sweeps and related modifier sounds are available so that the modifier section is less important. By having the sample editing facilities available, changes can be made to individual samples rather than the global filtering that is available in a modifier section.

three most powerful of these sample editing techniques are looping, stretching and re-sampling.

5.12.1 Looping

Looping consists of implementing the equivalent of a loop of tape, but in the digital domain instead of with a physical loop of magnetic tape. In the simplest case, this is merely a repetition of the same portion of the sample, but it may also be controlled by an EG so that the loop does not stay at a fixed volume. The transition from the end of the loop to the beginning of the loop is the equivalent of the splice in a physical tape loop, but the control of this transition can be much more sophisticated because the sample is stored in a digital form.

The basic method of joining the end of the loop to the beginning is to splice the two points together. If the end and start of the loop do not have the same level, then the resulting 'glitch' will be audible as a click in the loop. There are several approaches to avoid this problem. It is possible to arrange for the splice to be made only when the two levels meet one or more criteria:

- same level
- same slope
- same rate of change of slope.

The 'same level' criteria is often refined to when the audio waveform crosses the zero axis; this is called zero-crossing. Since the zero axis is normally the effective 'silence' level of the sample, splicing at zero-crossings can produce splices without clicks, although this is not guaranteed. Better techniques take into account the shape of the waveform at the transition. Matching the slopes of the two portions of the waveforms can reduce the level of any click, whilst matching the two waveforms so that the splice point occurs at similar points on both can minimize the click, although this restricts the available splicing points. By reversing the direction of playback at the splice point, some of the problems of matching the levels or slopes can be avoided, but this works only for short samples where the backwards and forwards playback of the loop will not be heard – long loops can sound very unusual if they are looped using this technique, although it is useful as a special effect. The length of the loop also affects the perceived pitch of the sound as it is replayed. In extreme cases, the looped section can shift its pitch markedly from the original pitch before looping. This is most obvious for short loops, especially where a single cycle of the waveform is being looped (Figure 5.12.1).

Even if clicks are not produced by the splice in a loop, the start and end of the loop may not have the same timbre. This produces a sudden change in the timbre that can be almost as noticeable as a click. The abrupt change in spectrum of the sound is often interpreted by the listener as a click or glitch even

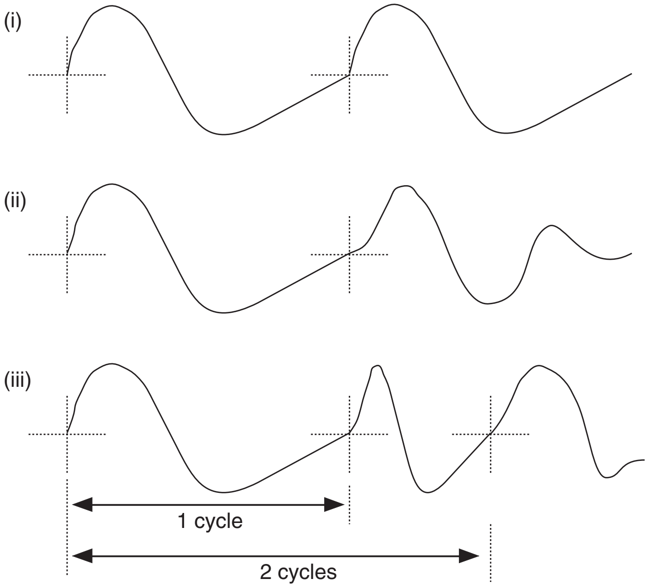


FIGURE 5.12.1 Splicing loops. (i) Choosing the same level does not guarantee a good splice. (ii) Matching the level and the slope can give a good splice. (iii) Even a good splice can alter the frequency of the loop if the same cycle time (or an integer multiple of it) is not maintained.

though examination of the waveform at the spice point shows no obvious mismatch of level or slope.

The second method of joining the loop is cross-fading overlapping the audio, and then fading the end out as the start is faded in. Cross-fading between the start and end of the loop can be used to reduce the effect of a sudden change into a smoother transition between the two contrasting timbres. Unfortunately, even a cross-faded loop can still produce a cyclic variation in timbre, or an obvious fade, both of which can be apparent when the loop repeats. For a single audio edit, a minor inconsistency is often not noticed as it happens once and is then gone, but a loop edit may be heard hundreds of times for a held note, and therefore the audibility of any defect is correspondingly magnified.

The subject of producing usable looped samples of sounds is a complex one, outside of the remit of this book, but it is often covered in considerable detail in the literature and support material produced by the manufacturers of samplers.

Once looped, the timbre of a sample is fixed. In real instruments, the timbre tends to change rapidly during the initial attack and decay portions of the envelope, and changes more slowly, if at all, during the sustain and release portions. Looped samples are thus most frequently used for the sustain and release parts of sounds, with cross-fading between two samples or a modifier filter used to provide a changing timbre in the other parts.

As with advanced S&S and wavetable synthesis techniques, it is possible for samples to have multiple loops each with an envelope, pitch modulation and velocity switching facilities. The transitions between multiple samples can also be modified with velocity and note-position cross-fades, that can help to minimize the abrupt changes in timbre which can be present between multi-samples. The hardware and software of the sampler itself determines exactly which methods can be applied to the samples. The sound manipulation possibilities which are opened by these techniques should not be underestimated – most samplers provide powerful synthesis capability even without using the 'synthesis' modifier section.

Looping a sound can markedly reduce the amount of storage that is required. For a 10-second 'CD-quality' stereo sound without any looping, about 2 Mbytes of storage is required. If 7 seconds can be produced by looping part of the sample during the sustain segment, then this can be reduced to about half a megabyte. Reducing the storage requirements can reduce the amount of RAM memory required in the sampler that may reduce the manufacturing cost, or can allow more samples to be stored in the RAM memory.

5.12.2 Stretching

Stretching is the name given to the process of independently adjusting the timing or pitch characteristics of a sample. Transposition changes both the pitch and the time of a sample if the sample is shifted up by an octave, then it plays back twice as fast and a '1 second' sample lasts only half a second. In contrast, stretching aims to change the time without changing the pitch or the pitch without altering the timing.

To change the timing, involves analysing the existing sample either by removing or adding sections, depending on whether the sample is being lengthened or shortened. For pitch changes, the sample is either lengthened or shortened by the pitch change, and therefore some of the individual sample values need to be changed.

The simplest approach to doubling the time is to just repeat all the sample values, which doubles the length of the sample, but preserves the pitch. But just repeating sample values is a little crude, and interpolation filtering can be used as in Section 4.5.4 to create new sample values by effectively estimating what the sample value should be. The simplest approach to halving the time would be to remove half of the sample values, but the use of interpolation filters can give a better set of sample values. An alternative approach is to take complete cycles and repeat them to double the time and to remove complete cycles to halve the time. The length of the sections that are repeated or removed is normally quite short: at least one cycle, but short enough that the repeated sections are not heard as repeats. Repeating cycles works quite well, and although it requires some analysis of the waveform to find complete cycles, it requires less processing power than interpolation filtering.

For pitch changes without altering the timing, the process is similar, but this time the sample stays the same length, and samples are added, removed or interpolated to increase or decrease the number of cycles.

More sophisticated techniques do more analysis of the waveform in order to produce better interpolations or identify better cycles, and some samplers allow control over the technique that is used.

For time or pitch changes other than doubling or halving, the same process of interpolation or cycle repeating/removing can be used, but with different ratios between the original sample values and the changes. Because any changes to the sample values are altering the original sample, this makes time and pitch stretching prone to audio quality problems – although this can be used as an effect on rapidly changing material to give a result that is similar to granular synthesis.

5.12.3 Re-sampling

Re-sampling is the name for using the sample record facility of a sampler to record the output of a sample replay. In a digital sampler, it is the digital signals that are used, and therefore the loss in quality is not dependent on the analogue-to-digital or digital-to-analogue conversions. Re-sampling allows the sample rate of a sample to be changed, or for an LFO modulated sound to be stored as a sample, or for a filter sweep to be stored as part of a sample. It enables 'snapshots' to be made of the output of the sampler, and then the reuse of these sample snapshots as the raw material for further sounds by reprocessing the sample snapshots through the sample manipulation and modifier sections. Because the sampling process does not capture all the a sound nor reproduce it perfectly, then re-sampling can reduce the audio quality. This can be especially noticeable if electrical hum or noise is introduced into the re-sampled version. Pitch-shifting also produces distortions, typically because of the limitations of the interpolation and other techniques that are used. Each time when re-sampling occurs, any degradation will accumulate.

It is also possible to re-sample digitally, and therefore avoid the digital to analogue conversions and vice versa. But the quality of the audio still degrades, and re-sampling (particularly of pitch/time changes) should be used sparingly.

The ideal way to re-sample is to record a sample that is played back without any pitch change (i.e., at the pitch it was originally recorded) so that any distortions caused by interpolation, and so on are minimized.

5.12.4 Multi-sampling

Multi-sampling is normally used either to provide changes in sound across the note range, or to maintain a sound across the note range. It is typically used for instruments that have marked differences in their harmonic structure for high and low pitches, most notably the piano. Samples are taken of the source sound played at different pitches, normally all at the same sample rate.

The limiting case of multi-sampling is that when each note on a music keyboard is sampled separately, then each note will be reproduced using a different set of sample values. This uses large amounts of memory, but provides the potential for the most accurate reproduction. Most multi-sampling is less extravagant than this, with samples being transposed to provide spans of an octave or perhaps a fifth rather than individual notes. Some instruments are very sensitive to transposition and therefore require lots of multi-samples, although others can sound surprisingly usable when a single sample is spread across the whole of the note range. Bowed string instruments are one example, although the effect of transposing violins down in pitch is more like a big low-pitched violin than a cello, which gives them a quality of false reality that can be useful as background pads. For solo instruments that are going to have the full attention of a listener, then multi-sampling of some form is probably more appropriate.

For multi-sampling where two or more samples are used across the whole keyboard range, the transition between samples can be important. As an example, consider two samples made of a piano one from each extreme of the keyboard. The low-pitched sound would be rich in harmonics, whilst the high-pitched sound would be a sine wave plus a 'plink' transient hammer noise from the hitting of the string. The changeover from one sample to the other in the middle of the keyboard is likely to be very noticeable to a listener! Another danger signal is if there are lots of chromatic arpeggios in the piece of music being produced by a sampler, since the changes from one transposed multi-sample to another can again become apparent.

Most multi-sampling does not use two samples taken from extreme ends of the range of an instrument. Instead, the aim is to provide enough samples to capture the characteristic sound of the instrument whilst minimizing the unwanted effects of transposing samples. Extreme transpositions of samples produce an effect called 'munchkinization', where the changes of pitch and timing emphasize the pitch change and give it a comic effect. This is particularly apparent on the spoken word or singing, although many instruments change their character noticeably when they are transposed by a large amount.

Since most playing of an instrument concentrates on the middle portion of the range of the instrument, most multi-sampling schemes involve having the most detail in this area. Note that this requires knowledge of the range of the instrument and its suitability for sampling and transposition. The percussive triangle has already been noted as one instrument that has a limited range, and another example is the tambourine.

The transpose range of the multi-samples is thus small where the detail and the transitions between multi-samples are important, but increases at the extremes of the range. This can be observed in many piano multi-sample sets. The bass notes often use a single transposed sample, whilst the high notes use a single 'plink' sample, with the smallest multi-sample ranges being present in the middle area of the keyboard. The transitions between these two extreme samples and those used in the central area are often the most striking, since this is where the largest compromise is made between choosing a suitable sample and ensuring a smooth transition between adjacent samples.

For accurate reproduction of sounds that have timbre changes with dynamics of playing, then additional multi-samples are used where different intonations or key velocities are used. Pianos are a good example of an instrument where the timbre varies with how hard the note is pressed.

The mapping between samples, their 'home' (untransposed, as originally recorded) pitch, the different intonations or dynamics and the notes as an output by the sampler is called a keymap. Pianos often have complex keymaps with many samples both across the note range and at a range of intonations or dynamics.

For instruments with large changes in timbre across their range, producing multi-sample sets can be complex and exacting work. Instruments that have a restricted range can also be a problem because most samplers will enable the playback of a sample over the complete range of the sampler, even if the source instrument cannot! This means that whilst a single piano sample can be utilized across the entire note range, it is only useful for providing synthetic textures that have some of the characteristics of the real instrument, and it is not suitable as a means of emulating a real piano. The most extreme example of the failure of transposition occurs in percussion instruments, where the fixed parts of the spectrum are essential to the timbre. Transposing a sample of a triangle or a tambourine produces instruments that merely sound wrong!

5.13 Storage

There are two forms of storage used in a sampler. The short-term internal storage is usually inside the sampler itself, whilst the longer term storage is often external to the sampler and is frequently removable.

The storage that is used to hold the samples as they are made or replayed is normally fast read–write memory called RAM. RAM is an acronym for random-access memory, and the name refers to the ability to rapidly access any location in the memory device at random. In contrast, a tape recorder is much more restricted in its access it either plays back the audio, or it requires to be wound or rewound to reach an alternative location on the tape. RAM storage does not have this problem any location can be accessed as quickly as any other. RAM storage comes in two forms: static and dynamic. Static RAM chips will hold their contents for as long as they are powered up, which makes them ideal for short-term storage using battery backup. Dynamic RAM chips lose their contents if they are not continuously 'refreshed' by the host microprocessor chip. Dynamic RAM chips are considerably cheaper than the static version, and therefore low-cost samplers are more likely to have dynamic RAM that will require backing up to another more permanent type of storage before powering down the sampler.

Longer-term storage is often associated with magnetic or even optical media, although the variation of ROM technology called 'flash' memory allows long-term storage of samples in memory chips that do not require a backup battery. Flash memory can be internal to the sampler or on a plug-in memory card. Suitable magnetic and optical media include the once ubiquitous but now almost completely obsolete floppy disks, as well as hard disks and CD-ROMs, in either fixed or removable forms. Memory cards can be either RAM or flash-based, or may include a miniature hard disk drive, and will typically use one of the many flash memory card formats. Samplers in the 1980s and 1990s typically used the parallel-organized small computer system interface (SCSI) bus to interface to external memory devices, and this allowed additional protocols the SCSI variant of MIDI (SMIDI) to be used to transfer samples at higher such as rates. In the twenty-first century, USB, USB 2.0 (as well as the 'once popular but now fading' IEEE488 or FireWire) serial connectors provide fast external storage interconnections with lighter cabling and smaller connectors. Networking of samplers together over a local area network, or LAN, allows samplers to share common storage devices. The use of large amounts of online storage forces the use of detailed management of the storage to enable specific samples to be located, and then loaded samples into memory for editing and playback, with the edited versions then being cataloged and stored again.

Digital audio signals require large amounts of storage. For a 44.1 kHz sample rate, stereo 16-bit samples produce just over 1.4 megabits per second or about 600 Mbytes per hour. This is easy to remember when you consider that an audio CD lasts for about an hour, and contains about 600 Mbytes. The 8-bit resolution samples halve these figures, but with a very significant loss in quality. Reducing the sample rate restricts the bandwidth, which is only useful with sounds that have limited bandwidths like some bass and drum sounds. Table 5.13.1 shows some examples of storage requirements for sampled audio.

Storage on hard disks has seen a halving of cost every 12 months or so for some years. In 2007, the 500 Gbyte external USB 2.0 drive became a common sight, with a cost comparable to a mid-range DVD player. By 2010, 2 or even 4 Tbyte drives may well be replacing them.

The early twenty-first century has seen the rise in popularity of sophisticated audio compression schemes. MP3 was the first, with typical data rates of about 128 kilobits per second for stereo audio. Reducing the data rate to about a tenth of the CD uncompressed rate can be achieved only by removing redundancy first, and then by reducing quality. MP3 coding does this by hiding the deficiencies in parts of the spectrum where they are masked by other louder sounds. MP3 also exploits the wide acceptance of small light headphones, and it is quite instructional to listen to MP3 encoded audio on a hi-fi system. AAC and other

The 'MP3' standard is actually just the audio encoding part (audio layer 3) of the moving pictures expert group (MPEG) video and audio encoding standard called MPEG-1.

Table 5.13.1 Storage Requirements for Sampled Audio

Resolution (bits)	Sample Rate (kHz)	Mono/Stereo	Time (seconds)	Storage (kilobits)	Storage (Kbytes)	Storage (Mbytes)	Time
8	32	Mono	1	256	31.3		
8	32	Mono	10	2560	312.5	31.3	
8	32	Mono	60	15360	1875	1.8	1 minute
8	32	Mono	3600	921600	112500	109.9	1 hour
12	32	Mono	1	384	46.9		
12	32	Mono	10	3840	468.8	0.5	
12	32	Mono	60	23040	2812.5	2.7	1 minute
12	32	Mono	3600	1382400	168750	164.8	1 hour
16	44.1	Mono	1	705.6	86.1		
16	44.1	Mono	10	7056	861.3	0.8	
16	44.1	Mono	60	42336	5168	5	1 minute
16	44.1	Mono	3600	2540160	310078	302.8	1 hour
16	44.1	Stereo	1	705.6	86.1		
16	44.1	Stereo	10	7056	861.3	0.8	
16	44.1	Stereo	60	42336	5168	5	1 minute
16	44.1	Stereo	3600	2540160	310078	302.8	1 hour
24	48	Stereo	1	1152	140.6		
24	48	Stereo	10	11520	1406.3	1.4	
24	48	Stereo	60	69120	8437.5	8.2	1 minute
24	48	Stereo	3600	4147200	506250	494.4	1 hour

coding schemes are reducing the data rate even lower. Although some gains can be made by increasing the processing, the ultimate loser is the quality, even if it is well hidden. Extreme compression algorithms also have the side effect of making the audio very sensitive to errors: a single error can have very serious effects on the audio output.

In samplers, there are different criteria to meet. Unlike music tracks, samples do not have the same broad spectrum of sounds offering places to hide distortion. Compressing sounds that feature one instrument at one pitch is not straightforward, and can easily expose any weakness in the compression technique. But there are some genres of music that thrive on distortion, and some genres that use broad-spectrum sounds, and for these compression may be appropriate.

5.13.1 Transfer of samples

In the early 1990s, using an external SCSI hard drive, or using a computer as a storage and editing device, was the limit of most sample transfers. The MIDI Sample Dump Standard (SDS) was intended to allow sample data to be transferred between samplers, but this was slow because of the large size of 16-bit, 44.1 kHz sample rate samples and the slow transmission rate of MIDI. SCSI-MIDI, or SMIDI, was an attempt to use SCSI as the transport for samples, but it did not see wide acceptance. The music industry lagged behind the computer industry slightly by continuing to use SCSI even as FireWire increased in popularity, and mLAN continues this trend by using FireWire even though USB 2.0 is now much more popular. The 2000s have seen a gradual change in rear panels to reflect the wide adoption of USB2 and FireWire, whilst front panels have increasingly seen flash-based memory cards (mainly used in digital cameras) replacing floppy disks.

The 1990s saw an increasing dependence on the CD-ROM as the medium of sample exchange, especially as the cost of CD-writers, and then CD-rewriters, dropped to affordable prices 600 Mbytes of samples on one CD-ROM was sufficient to store more than the complete sample RAM of many samplers. Removable hard drives gained some acceptance at the end of the 1990s, with the Iomega Zip drive of 100 Mbytes upwards being one of the longest surviving and most widespread examples. But the twenty-first century has seen removable hard drives being rapidly replaced by flash drives. The low cost, robustness and re-recordability of the CD-RW has made it very popular, whilst the one-time write CD-R has become very and very quick to write chapter 52 times recorders produce a 600-Mbyte CD in just over a minute. Many different variants of DVD-R seem to have gradually become readable and writable on all of the increasingly low-cost drives, and after a long battle, Blu-Ray emerged as the 'HD' optical standard, with HD-DVD rapidly vanishing from shelves in 2008.

SoundFonts and MIDI downloadable sounds (DLS) are formats that allow samples to be transferred over computer-to-computer connections, typically a LAN or the Internet. These are descended from the .MOD files that were first used in the 1980s to create music on computers from very simple sound generating resources.

Networking in audio recording has not really been the success than many expected or hoped. The mLAN has yet to see wide adoption. The RTP-MIDI (the competing IEEE-P1639 proposal has stopped) transport for MIDI over IP is implemented in Apple's Mac OSX, and is gradually spreading.

For protocols beyond MIDI there are two main contenders:

1. OSC, the Open Sound Control protocol, developed by Matt Wright at Berkeley for transferring music data over IP, seems to be gaining support from programmers and manufacturers, and is from the same team that proposed ZIPI.
2. HD-MIDI, from the MIDI Manufacturer's Association, seems to be a 'high definition' MIDI modernization, sample and audio file transfer solution.

What has become apparent is that the computer industry moved quickly with changes like the move from SCSI to FireWire/USB 2.0, whilst the hardware part of the music industry lagged behind it by a couple of years, and preferred to adopt well-established standards once divergent competitors had faded. This is changing as the music business moves increasingly to software, since then the interfacing is in step with the computer business. But a similar time-lag (and divergence) is apparent with the new digital audio networking standards, and MIDI may have been an exception that provided a brief 'Golden Age' period of ubiquitous inter-connectivity: one that we may never see again.

5.14 Topology

5.14.1 Types

There are three main types of digital sampler:

1. stand-alone
2. keyboard
3. computer based.

Stand-alone

Stand-alone samplers are normally designed to fit into a 19-inch rack-mount case. Control and editing functions are carried out using the MIDI protocol, although some samplers also have provision for an external monitor and a keyboard to provide improved access to the editing functions. Samplers are often controlled from a master keyboard or a synthesizer keyboard, but some samplers are designed to be controlled from the front panel; for example, for adding sound effects or replaying drum sounds.

Keyboard

Keyboard samplers are essentially a stand-alone sampler placed in a larger case and with an added keyboard. Although S&S instruments have seen considerable success with this format of instrument, keyboard samplers have been less successful commercially. S&S sample players where part of the ROM memory is replaced by RAM have been slightly more successful, although these are better described as user sample replayers, since they usually lack any way for the user to actually sample sounds.

Computer based

Computer-based samplers were initially manufactured in the form of plug-in 'sound cards'. Some of the early cards were very large and complex, although advances in electronics have meant that the more recent peripheral component interconnect (PCI) bus equivalents are considerably smaller in many cases. Some computer-based samplers have taken advantage of the built-in audio capabilities and increased processing power of modern computers and are then merely software, but their audio performance is very much dependent on the computer's audio circuitry. For computers where the audio system is not adequate, the cards are merely converters where the audio storage uses the computer's own RAM. Other cards may provide special-purpose processors to carry out DSP functions these are sometimes referred to as DSP 'farms'. It is also possible to find all of these separate parts on a single card.

The conversion from analogue audio signals to and from digital data is sometimes carried out in a separate box outside of the computer in order to optimize the conversion accuracy – the interior of a computer case is not an ideal location for a sensitive conversion system. In some systems, the external box is merely used to provide a convenient way to house all the connection sockets because plug-in cards for computers normally provide only a very small area of panel in which input and output sockets are located.

Direct-to-disk or hard disk recording can be thought of as a variation on computer-based sampling, although it has a different set of design goals. Although a sampler will normally record into RAM memory, and this sets a time limit on the length of the sample that can be recorded, a direct-to-disk recording unit will store the converted audio data on the hard disk directly, which means that the length of the recorded audio is limited only by the available hard disk size. This process places considerable demands on the computer and the hard disk, and in fact, the number of tracks that can be recorded and/or replayed simultaneously is determined by the computer's processing power and the rate at which data can be transferred to the hard disk storage.

As processing power and hard disk throughput has increased, the general-purpose computer has increasingly been capable of providing hard disk recording and playback capability, and the subsequent mixing down of multiple tracks into stereo or surround-sound mixes that are then stored on hard disk. This has resulted in computer-based samplers that never output analogue audio signals, and which are more like sample-based workstations than just samplers or sample replay devices.

5.14.2 Sample sound-sets

The open nature of samplers means that they are considerably more customizable than S&S ROM-based sample-replay instruments. Although it is possible to populate a sampler's RAM with the single-cycle waveforms that might be found in an S&S sample set, it is more common to use longer samples. The demands of the computer industry for plug-in RAM in the form of single and dual inline memory modules (SIMMs and DIMMs, respectively), as well as ever larger and faster hard drives, have meant that samplers have acquired very long total sample times, and large libraries of samples on hard disk to fill the RAM.

Apart from the extremely detailed pianos, violins and other orchestral instruments, sample sound-sets are available for a very wide range of vintage and ethnic instruments – a much wider range than what is found in S&S sample-replay units even where specialist sound-sets are available in ROM cards.

But a huge number of sample sound-sets are available which are intended as pads and special effects sounds. Complex evolving textures and ambient soundscapes can make further processing inside the sampler almost unnecessary.

Samplers are also widely used in a field where S&S sample replay has very limited facilities: loops. Loops are one or more bars of rhythmic patterns made up from drums, bass and other accompaniment instruments, sometimes with melodies as well. They are intended to be the raw material from which pieces can be constructed in much the same way that a groove box or a phrase sequencer works (see Chapter 8). Longer loops exist, but these are often intended for audio-visual presentations where background music is required.

Samplers often allow loops, and particularly drum loops, to be separated out into short samples, often on a 'per beat' basis. This means that a single drum beat in a loop can be extracted for use independently or can be moved in time inside the loop. Some samplers allow shuffling of these loop fragments to be carried out with varying degrees of randomness. The loop is one of the evolving areas of sampler technology, especially in the live performance context. The rapid evolution of computer software, as opposed to sampler hardware, has resulted in loops seeing quick adoption on computers, whilst hardware samplers have lagged behind. The end result has been that hardware samplers had almost completely disappeared by the mid-2000s, whilst computer-based sample replay and loop-based music generation had exploded in popularity and availability.

5.14.3 Using samplers

Samplers can be used as pure replay instruments in much the same way as S&S synthesizers. But this is not exploiting the capability of the sampler to change its complete sound-set, and especially to reproduce sounds that are not purely instrumental.

Because a sampler can be loaded with a specific sample set made up of a number of sounds, even the sounds produced by synthesizers, it can be used as a way of producing the sounds from a large variety of instruments from one piece of hardware or software. It is also possible to record samples of several synthesizers played together and mixed into one complex sound, and then to use the sampler to replay the sound. A complete rack of hardware synthesizer modules can be replaced with one or more sample sets, and therefore the combination of a single keyboard and a sampler can be used to replace complete racks of synthesizers.

Samplers are also good when either a complete backing track or loop is required without needing a sequencer, several synthesizers, a mixer and some outboard effects units. Vocal performances can also be sampled and used as backing vocals, sometimes even as solo lead vocals. Special effects are another type of sounds where samplers can be very useful for playing back a range of sounds chosen from a larger library.

The flexibility of samplers requires considerable investment in time and sampled sounds if the capability is to be exploited properly. Auditioning sample sounds from CDs or CD-ROMs is a slow task, and turning these selected sounds into sample sets for specific songs takes careful planning and consistency of assignment of sounds to MIDI channels or sequencer tracks. Making backups of any sample set definitions is also essential with many samplers, although some samplers are now incorporating flash memory, which reduces the need to take backups but does not remove it completely.

5.14.4 Convergence of sampling with S&S

The fundamental differences between an S&S synthesizer and a sampler are often described as being related to the sample memory and the sample processing.

- *Sample memory*: There is a popular misconception that S&S synthesizers have permanent ROM memory whilst samplers have volatile RAM memory. This view ignores the way that both types of instruments have evolved. S&S synthesizers have acquired user sample RAM, whilst the wide usage of pre-prepared samples in samplers virtually relegate them to replay-only status.
- *Sample processing*: S&S synthesizers normally have a restricted set of controls for the replay of samples, but this is usually compensated for by the provision of a sophisticated synthesis section with a resonant filter and voltage-controlled amplifier (VCA). Samplers often concentrate more on the sample-replay controls, with multi-sampling, looping, sample stretching and interpolation between one sample and another, although their subsequent processing is often just as capable as many S&S instruments.

The differences are thus less apparent than is often supposed. There is an ongoing convergence of functionality in both instruments. S&S instruments can have user sample RAM memory, and external sampling units can provide samples, although CD-ROMs are more frequently used to provide additional 'off-the-shelf' sounds, much as with a sampler. Samplers now use CD-ROMs and hard drives to provide rapid access to raw sounds in much the same way that S&S instruments provide sample replay from ROM or RAM memory.

Samplers are sometimes used merely as replay devices. This wastes the creative potential of the synthesis sections that can be used to give great effect in processing the samples and providing new sounds. There is some evidence of a stigma being associated with samplers because of this 'replay-only' reputation, with some people preferring to use S&S instruments with user sample RAM memory instead of a 'sampler'. The convergence between S&S and sampling should soon produce instruments that are so difficult to categorize into either of the two types that the bias against samplers may change.

5.15 Digital effects

Digital synthesizers often include effects. Flagship workstations tend to have equivalent effects, whilst lesser instruments have more modest effects. There are two advantages to having effects 'built-in' rather than as external or 'outboard' effects.

1. The effects can be included as part of the sound, which means that not only are they selected automatically when you select a sound, but also parameters used in the synthesizer or sampler can be used to control effects as well. (It is possible to make external effects units choose an appropriate effect by using MIDI program change messages, but then you need to map all of your sounds to appropriate effects, which is time consuming and awkward to manage.)
2. The effects are carried out digitally inside the instrument, whereas with an external effects unit the digital signals are converted to audio, then converted back to digital for processing and finally converted back to audio again to be sent to the mixer, which can affect the quality of the audio.

Using parameters inside the instrument provides a number of possibilities for controlling the effect so that it is affected in context, whilst also allowing for separate asynchronous operation as well. Some of the ways that internal parameters can be used include the following:

- The LFO speed can be used for vibrato in the instrument, and chorus in the effects (maybe in anti-phase).
- After-touch can be used to control the reverb mix.
- Echoes can be synchronized to tempo.
- The modulation wheel can control the resonance of the filter and the phase angle of the phaser.

One use of an external effects unit might be echoes that are deliberately not synchronized to the tempo!

5.16 Digital mixers

Digital mixers make working with digital synthesizers and samplers easier because the motorized faders allow the storing and recall of mix settings that can be set for positions in songs or just named scenes. Many mixers provide much more store and recall capability than this, often the whole of the user interface can be saved and recalled. This opens up the possibility of making the mixer an extension of the instrument by deliberately utilizing the mixer capability during a song.

Digital mixers often have built-in effects, and these can be used to augment the effects present in the digital instruments, or can be used as an overall effect: reverb, for example.

Although many digital mixers provide digital outputs, digital inputs are not as common. Of course, there are not very many digital instruments with digital outputs, although it is gradually appearing as an option, particularly where the instrument allows hardware options to be added therefore expansion bays.

5.17 Drum machines

Traditional drum kits are large, heavy and loud. They also require considerable time to assemble, and recording drums and percussion in a studio can be a time consuming, exacting process. But drum and percussion sounds are the perfect accompaniment and contrast to the strongly pitched sounds produced by keyboard-based synthesizers, and therefore the application of electronics and synthesis to creating drum sounds has a long history.

5.17.1 History

The earliest electromechanical devices to produce drum sounds as a rhythmic accompaniment were tape based and were probably derived from the practice of splicing tape into a loop so that it would play repeatedly. Harry Chamberlin produced a few of the first purpose-built stand-alone tape-loop rhythm units in 1949: the Rhythmate 40. This type of tape playback unit is the basis for the many later tape replay devices such as the Chamberlin and the Mellotron.

Ten years later, in 1959, the Wurlitzer organ company released the 'Sideman', a rhythm unit that had a rotating disk to actuate electrical contacts that timed the 12 rhythms using 10 drum sounds produced with valve filtering and shaping circuitry. This was a reworking of a musical box: a disk instead of a drum with pins as the timing mechanism combined with sophisticated sound generating circuits to replace the metal tines. It is interesting to note that the technology of the time was very much based on combinations of electrical motors to provide rotary motion, mechanical linkages, magnetic induction for signal generation and valves electronics for signal processing. The same technology was used in organs, rotary speakers and drum machines.

After another 10 years of incremental development, including the Donca-Matic DA-20 produced by a Japanese company called the Keio Organ Company, or Korg, the transistor replaced both the electromechanical discs and the sound generation circuitry. One of the first products made by Roland in 1972 was one of these early transistor rhythm units, the TR-33. In the 1970s, organs quickly acquired rhythm units, and development was rapid, with rhythm units gradually moving from the home organ into other areas of music. Roland's TR-77, from 1972, was one of these crossover products that was used by non-organists, featuring on several hit records. In 1975, PAiA, a built-it-yourself electronics kit company, produced one of the first programmable drum sets with eight drum sounds. Hobbyist electronic magazines at the time were full of kits for drum machines, analogue synthesizers and audio processing equipment of all kinds. It probably comes as no surprise to the reader to discover that the author of this book was an active builder of many of these devices from the early 1970s onwards and went on from this to repairing synthesizers professionally in the late 1970s.

In 1978, Roland launched their first user-programmable drum machine, the CR-78 (CompuRhythm). This was large, being housed in a wooden box that was almost a cube in shape, and it echoed the styling of the early tape and disk-based rhythm units. A year later, the TR-808 was released, and this was very differently styled, being intended for use by synthesizer users rather than home organ players. Although not a huge success initially, it provided limited control over the drum sounds themselves and complete user programmability with a very clear interactive display of when drum sounds would play- a row of switches and light-emitting diodes (LEDs), with the switches selecting when a drum would sound, and this being indicated by the LED being lit. When the pattern plays, the LEDs light up in sequence as time scans across the switches. This type of intuitive interface has been widely adopted for subsequent drum machines and other live performance devices.

In 1981, the TR-808 ceased production, and a scaled-down version, the TR-606, with a smaller case, chrome styling and a simplified user interface, was released by Roland as the drum part of a pair of devices. The other device was the TB-303, a dedicated sequencer driving a bass synthesizer. This linking of a drum pattern with a bass sequence was intended to replace most of a rhythm section for guitarists and keyboard players, but it was actually the starting point for the later phrase sequencers or 'groove boxes' (see Section 8.7).

The TR-808 was rediscovered in 1982 by hip-hop dance track producers and eventually became the trigger for much of subsequent dance-oriented, electro- and techno-music genres. But it became a huge success only after it was no longer in production, a phenomenon that is still seen in a world of short product lifetimes but longer cycles of musical fashion. The drum sounds in the TR-808 are produced using ringing filters and filtered noise and have become very popular as part of the definitive 'analogue drum machine' sound-set. The 1984's TR-909 from Roland saw the same thing happening all over again. It was an improved TR-808, with more accenting detail possible than the TR808, and it provided a shuffle control to provide swing in the patterns. Once again, it became the machine to be used for dance music almost as soon as it stopped being manufactured.

Because of the continuing popularity of discontinued drum machines, a number of manufacturers, mostly European companies, have started making drum machines that are strongly influenced by them, but brought up to date and with additional features. The Jomox X-Base 09 that was released in 1997 is one example from Germany that has much of the look, feel and sound of the Roland TR-909, but which adds more pattern memory and a much better MIDI implementation. Some manufacturers have even re-released equipment because of demand. For example, E-mu's SP1200 sampling drum machine was first released in 1987 and discontinued in 1990. But, as with many other drum machines, it was being used extensively in hip-hop, and therefore E-mu revised it and re-released it in 1993, with production continuing until 1998.

The 1979's Linn LM-1 drum machine was influential because of its use of sample drum sounds instead of using analogue circuitry, but only a few hundreds were made. The LinnDrum, which followed in 1982, was probably the first commercially successful drum machine to feature digitally sampled drum sounds, and it had a better sampling rate and some new samples compared to the LM-1. The LinnDrum was widely used in the early 1980s, and development was rapid.

In 1983, E-mu released the Drumulator, which had a tiny 64-Kbyte sample RAM, 8-bit samples, and therefore very short sample times for the 12 drum sounds. The Oberheim DMX in 1980 was more powerful, and by the mid-1980s the Japanese manufacturers were producing sophisticated drum machines with sample replay. Yamaha's 1986 RX5 was one example that featured lots of pads, programmable drum pitch and drums sounds on plug-in cartridges. The 1991s RY30 drum machine had sound generation that was simple S&S and featured a real-time controller wheel. The year 1992 saw the start of an alternative to the desktop: pocket-sized Yamaha's RY10 drum machine that was in a VHS videocassette-sized case.

In 1991, General MIDI (GM) standardized the assignment of drum sounds to MIDI note numbers, and this may have signaled the end of the drum machine as a stand-alone tool. When drum machines were separate, and had their own individual or proprietary assignment of drum sounds to MIDI note numbers, then it was not easy to transfer drum patterns from one machine to another or from one MIDI system to another. GM standardized the drum allocations, and the MIDI file was used to transfer drum patterns. Drum machines made just before and just after GM have very different approaches to how they map drum sounds to MIDI note numbers. Yamaha's RY30 has several mapping tables, later drum machines have several different drum kits, all using the same MIDI note numbers, but with different sounds. It was now very easy to take drum patterns and move them from one set of sounds to another and from one drum machine to another.

By the mid-1990s, the Japanese manufacturers were including drum sounds as standard in many keyboards and modules, and drum machine releases began to slow. For example, Yamaha's last separate dedicated drum machines were released in 1994 (the RY20 and RY8, both derived from the RY10). Roland's last drum machine was the CR-80 Rhythm Player in 1991, although they continue to make electronic drum pads, and their guitar-oriented Boss name continues to make drum machines.

By the start of the twenty-first century, the major manufacturers of drum machines were mainly companies who also made effects processors and guitar accessories-Alesis and Zoom. Akai and Roland (as Boss) are also active.

Many dance music producers no longer use drum machines; instead they just use samplers or software sequencers. The sample loop has replaced the drum pattern in many applications (Figure 5.17.1).

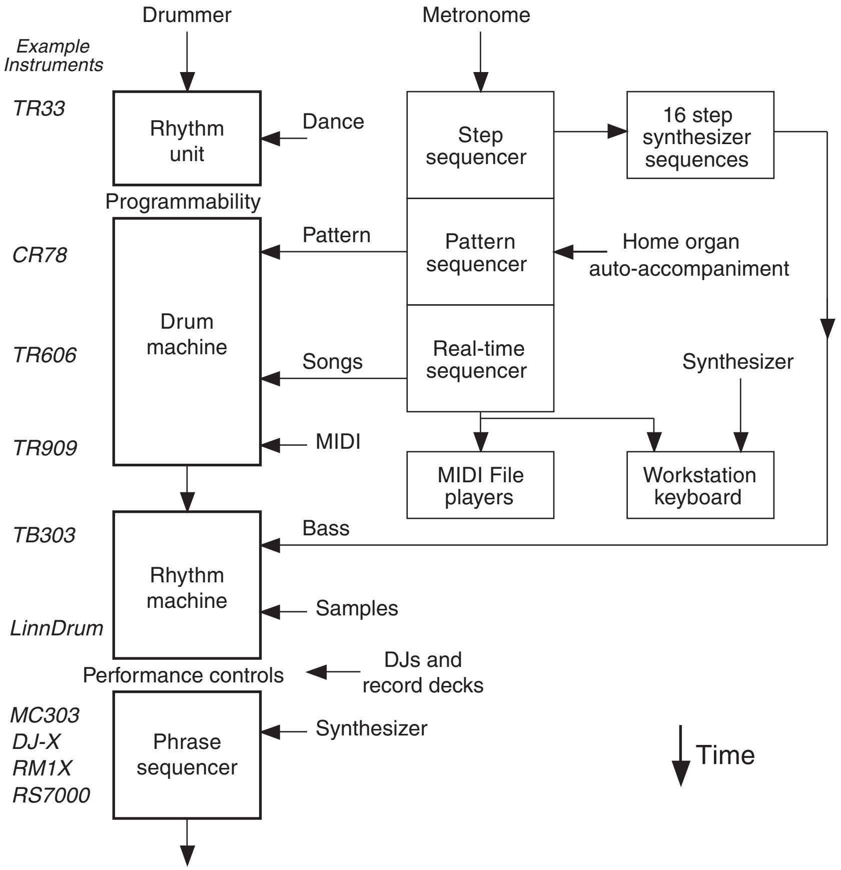


FIGURE 5.17.1

Drum evolution.

5.17.2 Inside a drum machine

The electronics used to produce drum machines has become widely available, and so basic drum machines have become very affordable, whilst computer-based sequencers and more sophisticated hardware sequencers have replaced drum machines for many professional users. But the internal operation of a twenty-first century drum machine is still a good starting point for learning about sequencers, although actually the basic design and hardware have changed only in detail since the 1980s.

A drum machine combines a cyclic timing device with a number of drum sounds. The timing uses a clock to set the tempo, and this can either be local to the drum machine, or derived from another MIDI device through the MIDI Clock messages. The clock is counted to produce beats, and these beats are separated or demultiplexed to provide individual outputs for each beat. Further counting circuitry is used to provide a count of the number of bars and this is used to derive the timing for the overall song. If the individual beat outputs of the counter were connected directly to the drum sound circuits, then the drums would sound for each beat, and therefore a pattern buffer is used to hold the details of which beat actually produces a sound. The pattern buffer is effectively a set of switches that reflect the pattern that is held in memory. The patterns that are loaded into the pattern buffer are controlled by the song memory, that uses the bar count to determine which pattern is played in each bar.

The outputs of the pattern buffer are then mapped to the actual drum sounds using the electronic equivalent of a patch-bay. The patterns are thus independent of the drum sounds, and by changing the assignment of drums to outputs, the hi-hat could be replaced by a snare, the bass drum by a side drum, and so on. Drum sounds are normally also mapped to MIDI note numbers when they are transmitted from the drum machine's MIDI output, and if this is connected to a synthesizer, then the results are rarely melodious. Conversely, connecting a keyboard instrument to the MIDI input of a drum machine will give a keyboard where some of the keys will cause drum sounds to occur. There is some standardization of drum sound mappings in the GM specification, but this is not mandatory and does not cover all possible drum sounds. The loss of the apparent coherence of drum machine patterns when they are played by alternative sounds or by pitched sounds instead of drum sounds is a fascinating topic that has some parallels to cryptography and ciphers.

Once the beat outputs are mapped to the drum sound circuits, then the sounds are produced and mixed together to produce the audio output. There are a number of different ways of producing drum sounds. Early electronic drum machines used similar circuitry to the electromechanical disk-based rhythm units: ringing filters and gated filtered noise. Ringing filters produce bursts of tone when they are triggered by the beat output and are used for bass drum, tom and other pitched drum sounds. Gated filtered noise uses the beat output to trigger a short decaying envelope for a noise source, and this is then filtered with a band-pass filter. This technique can be used for percussive sounds like hi-hats, brush and cymbals. Snares and side drums can be produced using a mixture of these two circuits.

Digital drum machines often use sample replay to produce the drum sounds. These can be samples of the analogue circuits described earlier, or recordings of real instruments, or specially synthesized emulations of drum sounds. Some drum machines allow user samples to be used. The late 1990s and early twenty-first century has seen an increasing number of drum machines that use modeling techniques to produce sounds, and these are capable of producing realistic sounds as well as being able to alter the sounds in ways which would not be physically possible in the real world.

Manual triggering of the drum sounds is normally through small pads that are now normally velocity sensitive; until the early 1990s only the most expensive machines had this feature. These pads are also used to fill the pattern buffer when recording a drum pattern, and often find reuse as control buttons and a numeric keypad. Since the mid-1990s the drum pads in drum machines have increasingly

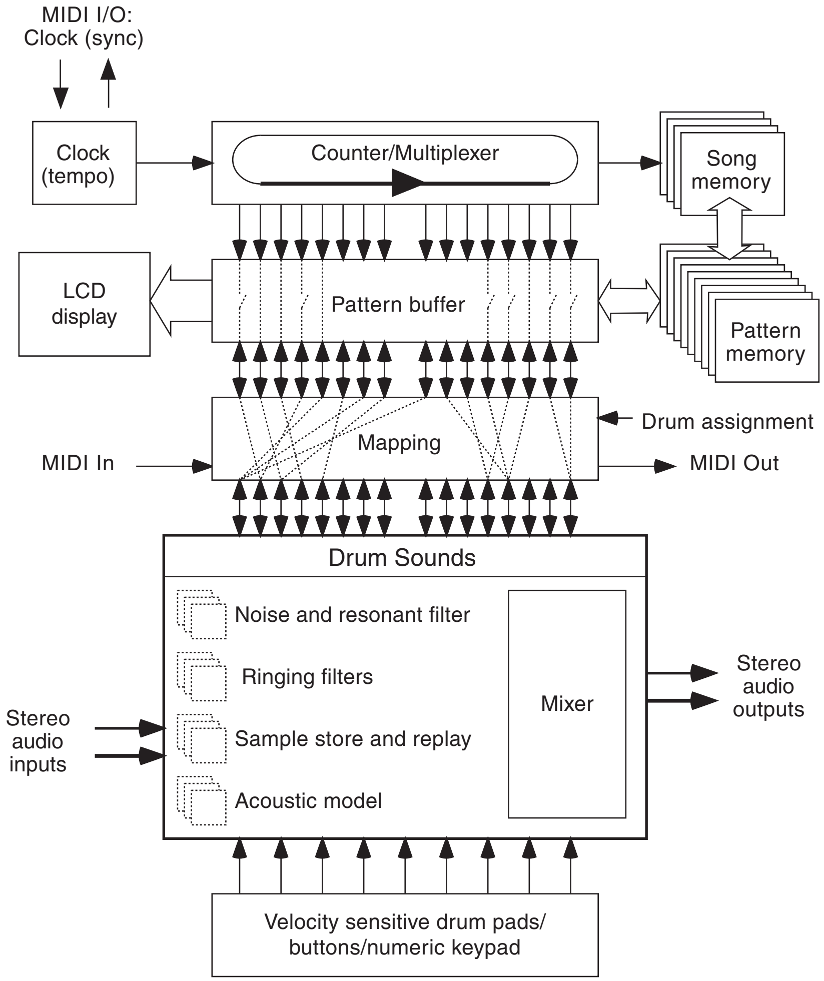


FIGURE 5.17.2
Drum machine schematic.

been laid out in a way that suggests the black-and-white arrangement of keys on a keyboard. This design approach enables the same pads to be used to control the pitch of pitched drum sounds, or even of samples of bass guitar and other sounds, and is even more important in the phrase sequencers described in Section 8.7 (Figure 5.17.2).

5.17.3 Drum machine operation

Figure 5.17.3 shows a typical low-cost drum machine. The velocity-sensitive drum pads are at the front, arranged in a keyboard pattern of 'black' and 'white' notes. These pads are also frequently used to control the operation through menus and enter values for parameters by acting as a numeric keypad, and therefore care needs to be taken when using them to determine which mode

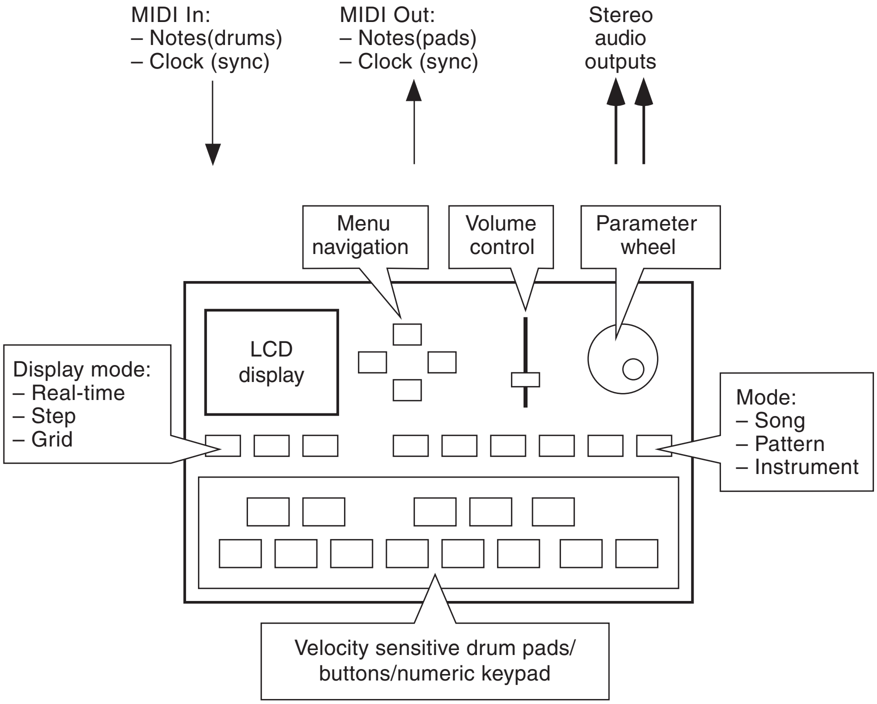


FIGURE 5.17.3 Typical low-cost drum machine.

they are in. Most drum machines divide the operation of the drum machine into modes like the following:

- song creation (chaining patterns)
- pattern creation (recording patterns)
- instrument settings (drum sounds and mappings)
- MIDI settings (inputs and outputs, clock sync,)
- play mode (active pads).

The use of the pads may well be different in each of these. When the drum machine is actually playing, then this 'play' mode usually forces the pads to become active and able to manually trigger the drum sounds. This is very useful for manually removing some of the repetition from long sequences of the same pattern by adding in some additional hi-hat or snare hits. Of course, if the pattern and song memory allow an alternative programmatic approach to produce several slightly different patterns and chain them together (Figure 5.17.3).

The patterns that are found on a drum machine are very much dependent on the current musical fashion. Early rhythm units were intended as accompaniment to organs, and therefore had dance names: like Waltz; Bossa Nova, Rock 'n'Roll; Mambo; Cha-cha; Beguine; March; Tango; Fox Trot and Rhumba.

From the 1970s to the end of the century, drum machines reflected the fall of progressive rock, the rise and fall of disco, the rise of dance music and most recently the rise of R'n'B. The music market has become divided into a number of separate areas, with little crossover between them, and drum machines have become locked into specific parts of these areas. So an early twenty-first century drum machine aimed at the high-tech market might have no reference at all in its patterns to traditional ballroom dances or guitar-based music, instead providing patterns based on musical genres like: Techno, House, Breaks, Trance, Hip-hop, Trip-Hop, Drum'n'Bass, Ambient. Home organs, though, now have built-in drum and rhythm facilities that reflect a wide range of musical influences.

For the synthesist, the factory preset rhythms are merely an illustration of the basic use of the drum machine, and replacing them with variations or new patterns is as much a part of the creative process as creating new sounds on a synthesizer. As Section 8.7 shows, the drum machine is rarely the stand-alone device, which it once was, and composition now encompasses the whole of percussion, rhythm accompaniment and melody.

Recording drum patterns can use any of the following three metaphors:

1. In real-time recording, the pattern loops continuously round its bar length, and any drum pads that are played will be played on that beat and bar in subsequent repeats. The drum machine thus behaves like a simple tape recorder, although a time-saving convention is that a recorded beat can be erased by holding the same pad down for the repeat when the pattern loops around.
2. In step recording, the pattern can be advanced manually by one beat at a time, and for each beat, the pads can be pressed to control which drum sounds happen on that beat. This is useful for complex drum patterns or for transcribing a pattern from a score.
3. In grid recording, the pattern loops continuously round its bar length as in real-time mode, but this time the pads are all assigned to the same drum sound, with each pad determining when in the pattern the sound will occur. The pads thus become on–off toggles for the drum sounds on specific beats. This method is useful for musicians with a strong visual feel for drum patterns.

All of these methods are reinforced by the liquid crystal display (LCD), which shows one or more sets of drum sounds as either a line with blobs to represent drum hits or a grid to represent several drum sounds simultaneously (Figure 5.17.4). This 'blob' display is the same as the grid mode pad layout. When working with a drum machine, the sound, the display and any feedback from LEDs on the front panel or pads should all be gathered by the performer as inputs that provide different aspects of information on the drum pattern. Keyboard synthesizers and sequencers tend not to provide as much information, and therefore the drum machine can be a valuable resource when performing.

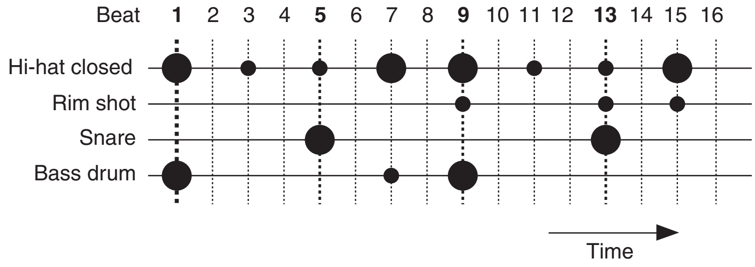


FIGURE 5.17.4
Drum grid (the larger blob size indicates accented beats).

Once one or more patterns have been recorded, then songs are created by chaining patterns together. The default is often set so that playing a song with just one pattern set will repeat that pattern, but songs are fully described by setting the pattern for a bar, then the number of repeats of that bar, or alternatively, the bar at which the pattern changes. The representation of songs in drum machines is not always as good as the pattern grid, and many drum machines merely provide a list of patterns against bar numbers. Again, more recent performance devices have improved on this, and software sequencers are generally much stronger in their graphical representations of song structure.

5.18 Sequencers

Early humans had two main ways of making sounds: the mouth and the hands. The mouth could sing, whistle and pop, whilst the hands could click, bang, slap and hit just about everything with anything that they could grasp. Given the right location, then environmental effects like echo could provide hours of slap-back entertainment for mouth and hands. With the right stimulus, a dog could be persuaded to stop its plesio-rhythmical barking, and to howl in an approximation of accompaniment instead. And when several mouths and hands were gathered together, then the resulting human orchestra had huge possibilities ... as well as enormous organizational barriers. Perhaps, speech is evolution's solution to trying to get acapella singers to do four-part harmony or attempting to persuade the percussion section to play three beats against four beats …

There seems to be an in-built dissatisfaction with solo music. Although many people can appreciate a neat melody, there is something about adding a warm rolling bass line, a splashy bright percussion track and a silky smooth pad that makes it so much more complete. But people are also lazy, and therefore becoming a composer, writing down lots of parts and conducting an orchestra is probably too much like hard work. What is needed is something that produces music automatically and semi-independently. Ideally this would be a compliant, intelligent, skilled fellow musician with infinite reserves of patience, but this specification would need to be open to compromise. And that is where the sequencer comes into play, or is that 'in to play'?

5.18.1 Beginnings

The wind-chime may have started out as a bird scarer, or vice versa, but it certainly offers a very primitive method of making music automatically. Once set on this mechanical path, then human ingenuity quickly explores the possibilities- friction bowing provides the hurdy-gurdy, and water or steam power armed with cams and punched control tablets opens up almost every conventional instrument to the on-demand replay of stored performances. Clock-making skills can be re-purposed into musical boxes, reprising the chime sounds of the wind-chime. But mechanical ingenuity has limits, and although it is possible to construct player pianos, steam organs and musical boxes with user-definable control mechanisms, most musical instrument retailers are not full of customers looking for them. Replaying fixed patterns is okay as far as it goes, but …

Electronics changes everything. As with rhythm units, once you realize that there is no longer any need for the physical movement of a mechanical device, then there are less constraints and you can easily achieve some very sophisticated control possibilities. Relaxation oscillators are an example. They are simple two-transistor circuits where the frequency of the sawtooth waveform that it produces is related to the current flowing through the circuit. Change the current that flows through a relaxation oscillator and you have a simple way of controlling the pitch. But best of all, if you put a low-frequency relaxation oscillator in the wire that supplies the current to another relaxation oscillator, then the pitch changes at the rate set by the slow oscillator. Stack three or more of these oscillators together and you have the sort of device that makes most people produce comments like 'stop that noise!' But to the ears of a synthesist that cacophony can be something much more significant; with just a few knobs it is possible to produce a vast range of complex rhythmic warbles, whistles and whizzes. Minimal compositional effort, no score-writing and no performers to conduct!

Relaxation oscillators have severe limitations. But if you transform that current control technique into one based on voltages, add a keyboard stolen from an electronic organ modified to produce fixed discrete voltages and mix in lots of other electronic processing goodies, then you eventually get an analogue sound synthesizer. Extending that idea of using one oscillator to control another- a low-frequency square wave gives possibly the simplest electronic automation that has a musical purpose rather than merely sounding like an alien siren: the trill. In the limiting case, this is just 2 notes repeated one after the other that matches the two levels in the square wave perfectly. Trills sound best when the interval is an integer number of semitones, and although this sounds simple to achieve, skill and expense are needed to make it happen, and to keep it happening. Much more challenging is how to produce longer and more complex sets of CVs or currents.

The multiplexer in the drum machine schematic (Figure 5.17.2) behaves in much the same way, but with drum triggers instead of CVs.

The answer is to produce a sequence of CVs using a counter or a multiplexer- essentially a switch that selects different CVs, and which moves cyclically round those voltages, repeating the 'sequence' of voltages. Connect those voltages to an oscillator where pitch is controlled by the voltage, and you have a sequencer which will repeat that sequence of notes. The easiest way to make a multiplexer in the 1970s was to misuse chips used in the computer industry, and the simplest of these had eight outputs. The result was a sequencer that produced sequences of 8 notes, and by connecting together the two, sequences of 16 notes. In one of those curious serendipitous coincidences, it turns out that a 1- or 2-bar, 8- or 16-note sequence is probably the shortest sequence that is quite interesting to listen to for more than a few bars, as well as being economic to produce as a circuit. It is these 16-note sequences that form the basis of much of the electronic music of the 1960s and 1970s, plus more recent retro revivals.

Analogue step sequencers are prone to problems with setting the pitch of the notes. The obvious approach is to put a control knob for each step of the sequence, but then each knob needs to control pitch over several octaves, which makes them very sensitive. It also means that changes in temperature or accidental knocks can all too easily detune a sequence. One answer to this problem might be to try and produce a derivative of the keyboard divider chain or resistors, and replace the continuous pitch control knobs with switches. But the solution that became adopted was much neater, and it has much the same effect.

What is required is to change the pitch knob from one which produces continuous changes in pitch as it is moved to one which works chromatically-jumping from one semitone to the one above or below. Producing a circuit that does it is rather like a specialized type of ADC where the pitch knob is the analogue input, and by setting the digital output so that each semitone produces a voltage change of 1.00057779 volts. Notice that even though we require an analogue step sequencer, the beat counter and multiplexer are digital, and so is the pitch quantizer or chromatic converter. This illustrates the difficulty of making a pure analogue performance device and shows that many are a hybrid of analogue and digital circuitry.

Step sequencers are very effective on stage, especially in darkness when the scanning of the LEDs across the 16 beats is visible. But 16 steps are also limiting, and in 1977, Roland, introduced the first computer-based hardware sequencer with significantly larger storage- the Roland MC-8. Roland called it a computer music composer. This was an expensive, professional device consisting of two boxes- one containing the digital eight-track sequencer and the other containing the DACs to produce CVs. The MC-8 was very straightforward to use, but rather tedious. The programmer entered step times, gate times and CVs individually into the tracks by entering numbers into a numeric keypad. Copying a pasting was an innovative addition to the feature set, and this improved the entry of numbers. It was also possible to enter notes using an analogue synthesizer keyboard to generate CVs, but this often required detailed editing with numbers to sort out the timing. Storage of completed sequences was through a compact audio cassette in much the same way as the home computers of the time, and was restricted to just 5300-note events in the expanded 16-Kbyte RAM models (1100 in the original 4-Kbyte RAM version). From the viewpoint of the twenty-first century, it is hard to believe that anyone could ever have used audio cassettes to store digital information, and patience and perseverance were valuable allies.

Although there were other computer-based sequencers from other manufacturers available before the MC-8, they were based on keyboard entry or did not have the depth of control or synchronization facilities of the MC-8. Roland improved on the MC-8 with subsequent release like the MC-4 and the MC-202, a novel device from 1983 consisting of a two-track sequencer and a single VCO synthesizer not unlike the SH-101 'sling it over your shoulder' performance synthesizer. Perhaps, the ultimate expression of this line of hardware 'enter by numbers' sequencers was the MC500 from 1986 and the revised MC-500 II from 1988, with standard 3.5-inch floppy disk storage.

Yamaha's QX1 computer-based sequencer in 1984 added dedicated keys for note lengths, but had a 5.25-inch floppy disk for storage, which was in a proprietary format. Ten years later the effect of two standardizing forces meant that a sequencer would have a 3.5-inch floppy disk for storage, and that it would read and write standard MIDI files. If you have a sequencer that can be used to create MIDI files, as well as read them, then adding sound generation turns it into a much more complete device, since in one box you can listen to the sequences without any need to connect the sequencer to an external sound source. Drum sounds were also standardized by GM, and therefore when Yamaha stopped making stand-alone hardware sequencer when they released the QY700 in 1996, it was a sequencer with floppy drive, instrumental sound source and therefore drum sound source. In fact, it actually added a third element, song chaining therefore preset phrases, rather like a drum machine (Figure 5.18.1).

It should be noted that hardware sequencers have a tendency to go out of date primarily because of the storage media used, and secondly for the on-board memory size and features. Longevity of storage devices is not a feature of the computer industry, one of the few exceptions being the 3.5-inch floppy disk. In a recent example, many late 1990s and early 2000s music devices used the SmartMedia flash card, which became obsolete in the mid-2000s.

In the studio, the gradual transition from hardware sequencers to software began in the mid-1980s with Apple Macintosh computers. For live performance then, the computer has probably been seen more for set dressing than serious use. Computers are not made for the stage. Unreliable electricity supply, interference from lighting controllers and a lack of anywhere flat to put a mouse are all factors that weigh against the computer, but probably the most significant one is the time it takes for a computer to restart after a power failure. A hardware sequencer can be back in operation after a few seconds, whilst

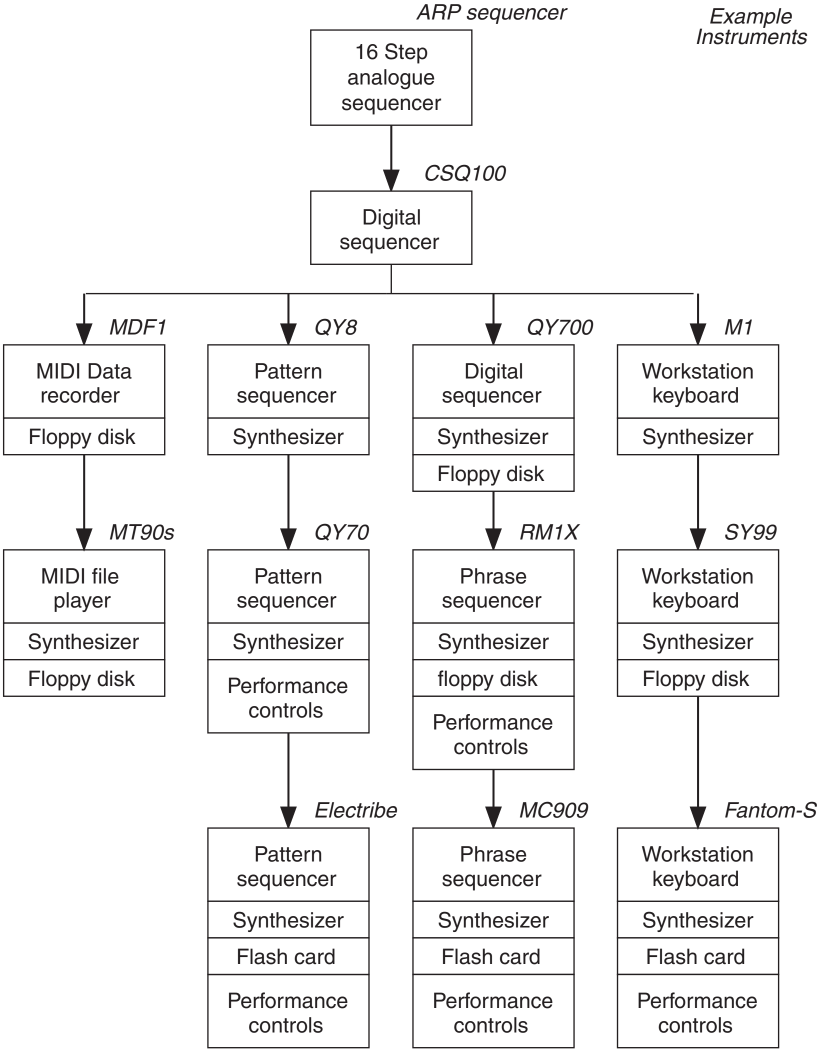


FIGURE 5.18.1
Hardware evolution.

for a computer it could be several minutes. What changed in the 1990s was the availability of laptop computers that were powerful, small, light and portable and that could run from internal batteries. Power failure was no longer a problem, and musicians had a sequencer that could move seamlessly from the studio to the stage.

Section 6.4 covers software sequencers.

A variation on the hardware sequencers is the MIDI data recorder or MDR. Early hardware sequencers were not well suited to receiving large MIDI system exclusive (sysex) dumps, and the MDR was a purpose-built device that was designed solely to record MIDI data and then play it back. With minimalistic controls derived from tape recorders (play, record and stop) mixed with computer filing systems (next file and previous file) and storing to 3.5-inch floppy disks, most MDRs are simple and functional.

MDRs that can interpret and play out MIDI files are called MIDI file players, and these can be used to provide backing tracks from the many floppy disks of pre-recorded MIDI file songs by using external MIDI sound sources. Some MIDI file players incorporate a GM-compatible sound source, and these provide a single piece of equipment that can be used as a general-purpose accompaniment unit.

5.19 Workstations

The late 1980s saw a change from the 'pure' synthesizer to the workstation: a combination of sound source and sequencer intended to form a single compositional device.

Although only conceptually combining two functions, sound generation with sequencing, a music workstation actually provides a larger number of distinct capabilities. These capabilities are normally implied by the word synthesizer or sequencer, but it is worth enumerating them in order to illustrate the changes required to merge them into a single coherent unit.

The sound source needed to be multi-timbral, provide piano, orchestral and band instruments, as well as drums, percussion, special effects and synthesizer sounds with a velocity and after-touch pressure-sensitive keyboard and an effects unit.

The sound-set provided by many synthesizers of the time needed to be widened to meet this specification, and two specific instruments usually needed to be added: drums and piano. Analogue synthesizers are not well suited to these instrument requirements (particularly the polyphony and multi-timbrality required by drums), and FM synthesizers were not well suited to providing the drum sounds, but sample replay was, and therefore S&S became the default sound source.

Yamaha allegedly returned to earlier research on adding pulse code modulation (PCM) sample playback to the DX7 FM synthesizer, in order to produce the SY77 FM and AWM (sample replay) workstation in 1989.

The sequencer needs to be a multi-track digital event recorder that can record, store, recall and replay the musical information for the composition: note events (pitch, timing, duration and volume), controller events (including note timbre, pitch-bend and modulation), drum events, timing events, effects settings, drum patterns, songs, sounds and setup data, as well as complete workstation status data. Ideally, storage should be in a removable form floppy-disk (1990s) or flash memory cards (2000s).

Hardware sequencers in the 1980s were capable of recording the MIDI sysex information for all of these events, but some integration was needed to simplify this storage, particularly for storing the complete workstation setup.

5.19.1 History

One of the first sampler-based workstations was the E-mu Emulator II (EII) in 1984. This used companded 27.7-kHz sample rate, 12-bit samples that were stored as 8-bit samples, and had a 16-track (eight internal plus eight external MIDI) sequencer. The 1987 Emax extended the sample rate to 42 kHz. Both of these workstations lacked an internal effects unit, and were limited in polyphony and multi-timbrality. Perhaps, a better description is a 'sampler with sequencer'.

By choosing a very different sound generation technology, Korg was able to release the M1 in 1988. This was arguably the first commercially successful workstation, using S&S with (at the time) a huge 4 Mbyte ROM as its sound source with a sequencer and effects unit. The M1 was a huge success for Korg, and it paved the way for the S&S domination of the synthesizer market for almost the whole of the next 10 years. One of the key elements in its success was the high quality of the factory preset sounds and the samples in the ROM. Hidden away in the sequencer is a very interesting feature- phrase-based sequencing built up from short patterns. This is very much a forerunner of phrase sequencers (see Section 5.21).

Roland's 1988 D20 was an S&S synthesizer with sequencer, and a floppy drive, but still called itself a synthesizer. A year later, Roland released the W-30 music workstation, a sample-based keyboard that also had on-board removable storage (a 3.5-inch floppy disk drive), whereas the Korg M1 only had a RAM card slot for storage. Having a floppy disk for storage has become a standard feature of workstations, whilst synthesizers generally lack them. As the 21st century progresses, it seems likely that flash memory cards will replace these floppy drives, although unlike the 3.5-inch floppy, there are so many different and incompatible formats that were almost back in the 1980s, when the 3.5-inch Sony floppy faced competition from sizes of 2-, 2.5-, 2.8-, 3-, 3.25- and 4-inch alternatives.

The Korg M1's success led to the inclusion of a sequencer and effects in many subsequent instruments, and by the end of the twentieth century the stand-alone synthesizer had become something of a rarity. Instruments that are described as synthesizers often include sequencers, and in many ways, the work 'workstation' has started to mean a professional expandable controller synthesizer/sampler/drum machine with more than a five-octave keyboard combined with a sophisticated sequencer and a storage device. Synthesizers tend to have five-octave keyboards or smaller and are less broad in their soundsets, but more specialized in their sound generation.

At the opposite extreme to this definition of 'workstation', the start of the twenty-first century has seen the development of a number of specialized desktop devices that have some of the elements of a workstation, but not all. Typically they have the appearance of a drum machine, and this is reinforced by a drum machine-style set of pads arranged in a single-octave music keyboard layout. Inside they contain a synthesizer, drum- or sample-based sound generator, and a simple 16-step, pattern-based sequencer. In many ways, they are drum machines that happen to play pitched sounds instead of drum sounds, and in fact, many of them also provide simple sample-replay facilities for drums and other accompaniment sounds to augment the main sound generator. Yamaha's 'Loop Factory' and Korg's 'Electribe' are two examples of these low-cost alternatives to 1U rack-mount modules.

Many manufacturers have taken hardware sequencers intended for the professional market, and added high-quality GM-compatible sound sources to produce keyboard-less workstations. For example, Yamaha's last stand-alone sequencer was the 3.5-inch floppy disk-equipped QX5FD that was released in 1988, but was followed in 1990 by the compact QY10, the first of a series of combined sequencers and sound sources. The top of the range desktop QY700 also provides a good example of another trend towards miniaturized and portable versions of existing products. The QY70 and the enhanced feature version, the QY100, released in 2001, are almost the same specification as the QY700, but a fraction of the size and can be battery powered. The QY series of 'workstations' are single pieces of equipment that combine sequencing with drum and instrument sounds and are very powerful compositional aids.

E-mu approached the sequencer plus sound source combination from the opposite direction – they built on their studio sound source experience and added sequencers and features intended for live performance use. The last Proteus-series hardware S&S module, the Proteus 2500, launched in 2002, included a sequencer, but the MP7 and LX7 (and the drum/percussion variant, the PX7) Command Stations were rugged desktop devices that incorporated a number of live performance controllers to make them more immediate and interactive outside of the studio. E-mu then moved to software, with computer audio interfaces as their only hardware products.

Another genre of music workstation is based around the low-cost fun/home keyboards that are the opposite of the professional workstation. These have taken elements of home organs like auto-accompaniment and made them accessible to people with limited musical ability and funds. By taking one of these keyboards and giving it more specialized contemporary dance-oriented sounds and drum patterns, Yamaha created the DJX keyboard in 1998, followed by a keyboard-less version which used a CD-style rotary controller to emulate a DJ operating record decks. Although low cost and limited in their sounds and facilities, these can be used in performance, and the techniques used are very transferable to more professional devices. The misuse of low-cost musical devices has always happened, but sophisticated electronics means that some of them are now very viable for use in real performances.

One notable instrument that sits on the boundary between the synthesizer and the workstation is the Korg Karma. This has the sounds, drums, keyboard and sequencer to make it a workstation, but it also has some very sophisticated patented automatic note generation facilities that make this a performer's instrument with a very unique way of augmenting playing. The 'generated effects' are like an 'intelligent' arpeggiator that can take the notes you play in a held chord plus special trigger buttons as the starting point for transpositions and other harmonic expansions, or that can use the held notes as they are and re-trigger them in rhythmic patterns, or that can use a rhythmic pattern as the basis for drums of other note sequences or all of these in combination. The breadth and complexity of Karma mean that any description is going to be incomplete. The best description might be that Karma does for performance what a synthesizer does with sound. The software behind Karma is now a feature of many Korg products, and it is still being developed further.

'Digital audio workstations' are the latest extension of the workstation concept.These are combinations of sequencer, motorized-fader mixer, effects, hard drive and CD-writer and are sometimes called hard disk recorders or multitrack studios.

5.19.2 Using workstations

Workstations can be used in many ways. They can be used as synthesizers by ignoring the sequencer, or as sequencers by ignoring the sound generator or even as master controller keyboards or drum machines. But they excel at rapid composition because of their integration – there is no need to wire anything up with MIDI cables or connect audio into a mixer. Familiarity with the operation of the workstation is also a key enabler to working at speed, and the importance of learning it thoroughly is just as important as with any musical instrument.

The starting point for composing music using a workstation can be a specific sound, a drum pattern or a short sequence of notes. Many people use workstations as musical notepads by capturing ideas and storing them away for later development, or as phrases to be used in live performance as the basis for extemporization. One of the key techniques here is to use the storage facilities of the workstation to support and facilitate performance – use tracks and memories so that variations and builds are instantly available, rather than trying to retain a wide variety of favourites. Learning to throw away unused sounds, patterns and sequences to make room for new material can actually be a compositional aid as well, but make sure to store the unused material for the future too.

Workstations are very good at providing accompaniment for live playing. An arrangement with a drum pattern, block or arpeggiated chords and a walking bass line can be used as the backing for singing, guitar playing or even playing a solo melody on the workstation. Muting one of the elements of a performance can enable that part to be worked on or if a human performer is available. The same workstation setup can thus be used with no additional musicians, or with any number, just by muting the appropriate parts as the relevant performer becomes available.

Transferring a composition from a workstation to a computer sequencer in order to make detailed edits, or to increase the available polyphony, or to provide for more diverse instrumentation, is not always straightforward. Exporting the song as a MIDI file and importing it into the computer sequencer often requires post-processing of the information in order to adjust it for the different instrumentation. Differences in timbres and velocity response can change the feel of an arrangement. Pitch-bend, modulation and other real-time performance control may behave differently with alternative instrumentation. Since many workstations provide dedicated additional sequencer tracks for external instruments through MIDI, copying internal tracks across to these external MIDI tracks, can ease the inclusion of additional instrumentation because A–B comparisons can be made using track mutes.

5.20 Accompaniment

An accompanist can be the piano that supports the singer or solo instrumentalist; or an orchestral backing for a piano soloist in a concerto. Solo piano was the accompaniment for silent black-and-white films at the start of the twentieth century, and the start of the twenty-first century still sees singers–songwriters accomplishing the demanding and exacting skill of accompanying themselves on their piano as they sing and play simultaneously.

A duet on a piano is one form of accompaniment and has some of the function of a sequencer, except that sequencers are happy playing boring repetitive parts that might tax some performers of a duet. Drum machines might be seen as a replacement for a drummer, except that programming good patterns still requires drumming skills, and there are a number of electronic pads and percussion sensors that allow drum machines to be played or programmed by real drummers. Perhaps, the true role of an accompanist is to play not what they are told to play, but what the performer requires. This is much harder, and maybe a descendant of the Korg Karma will feature in this role in the future.

In the past, the drum machine played drum patterns, repeatedly, until the player stopped them. Some organs had a feature that only started the built-in drum machine part when you started playing the keyboard, but the reverse did not seem to emerge- the player stopped the drums at the end of the performance. Changing the drum pattern whilst playing was possible, but required single-handed playing on one manual whilst the other quickly pressed the button at the end of a bar. Programming a song into a drum machine turns it from an accompanist to a conductor, if the drum machine starts the chorus and the player has lost a few bars because they did an extra repeat, then they had better play the chorus, because the drum machine is going to continue to play the programmed song sequence.

Organs also feature another accompaniment device- automation that produces walking bass patterns and chordal accompaniment based on the root note played by the left hand and the dance genre selected on the drum machine. Unlike the conducting drum machine, this is under the control of the player, and therefore an extra repeat does not affect when the chorus is played. This type of automatic accompaniment can be very sophisticated and is found in home organs and home/fun keyboards, but rarely on synthesizers. Synthesizers may share many common bits of functionality with other musical instruments, but user-programmed accompaniment is their preferred differentiator.

Taking automatic accompaniment, mixing it with drum patterns and releasing it as software is what happened in the late 1980s with PG Music's 'Band-in-a-Box'. Once a minimalistic song representation of chords and melody as in a busker's fake sheet has been entered, then choosing a song style creates a complete multi-part arrangement with drums, bass, chorded backing and even extemporized melodies. By muting some of the parts, you get either as much, or as little, accompaniment as you require. Interestingly, this type of automatic generation of accompaniment also appears in some workstations, especially the small portable devices such as the Yamaha QY100 or the Roland Boss JS5 JamStation, and the idea of a fake sheet is very strongly related to pattern-based drum machine song creation.

One part of the extraction of human accompaniment that is possible happens with drum patterns. The difference between an on-the-beat, equal-volume, simple drum machine pattern and a real drummer's performance is called a groove. It is all of those slight timing variations and inconsistencies in volume that help to humanize performances. Capturing grooves allows them to be added to otherwise machine-perfect patterns.

Sampling has become one of the major ways of working with sounds, and software has provided some useful facilities which can aid accompaniment. Pitch extraction can be used to provide control over further processing, like correcting vocal pitching. Even singers who do not need their pitching ability improving can benefit from the creative misuse of pitch-shifters and harmonizers, as several records in the late 1990s showed.

5.21 Groove boxes

Roland started calling out the MC-303 a 'groove box' but with success has come a price, because the phrase 'groove box' has increasingly become a generic term for any composite device that incorporates a pattern or phrase-based sequencer, drum machine, sound source, effects and live performance controls. Putting all of these components together reflects the increasing integration that happened during the 1990s, and the result is a powerful stand-alone performance tool.

The idea is very simple. The performer creates a number of phrases, and then puts those phrases together to produce the song in performance. Because the phrases will loop repeatedly unless you select a new one, then the structure and length of the song is not fixed. Repeating a line or two, or missing out part of a verse, is no longer a problem.

This type of functionality is not restricted to just the hardware unit. In the 1980s and early 1990s Opcode's Vision sequencer software had a feature that allowed you to label phrases with letters of the alphabet and then chain them together by typing the letters on the computer keyboard. Typing 'abacab' could actually be used to create a verse/chorus/verse/break type of song structure very quickly.

Groove boxes vary in their design and detailed implementation. Roland has a broad range from small and simple to large and complex and a very different D-Beam ultrasonic controller. Yamaha took a gradual approach, starting with the RM1X's S&S voices, a phrase-based sequencer and a large display. The SU700 took similar sequencer functionality but married it with a sampler. The RM1X and the SU700 were then combined and the specification adjusted to produce the RS7000. E-mu took solid sounds and lots of polyphony, and added a rugged box and sequencer, with the added bonus of plug-in extra sound ROMs. Korg has taken a different approach again with its 'Electribe' series, which are a collection of desktop units that provide a step sequencer with S&S sounds plus either drums, samples, modeled virtual analogue synthesis and more. Boss groove boxes are more oriented towards guitar and bass players, although they do incorporate some 2D controller pads that are useful in other genres.

Phrase sequencing requires pre-planning if it is to be used successfully. Most groove boxes provide a number of controls over the pitch and the selection of phrases, and using these to the full is key. Depending on the type of song or style of music, and how the user wants to work with it, the immediately available phrases need to cover categories like an intro, a verse, a chorus, a break or middle eight, an outro and some fills. Another useful phrase to have in some circumstances is a bar of silence. There are several techniques for building up song phrases, but the simple one is to build up from a drum pattern or build down from a melody, adding accompaniment, a bass line, harmonies and rhythm parts until you have a very dense, full arrangement. The individual phrases are then just this core phrase with different parts muted out. The intro might be just the bass line or perhaps the drum pattern. The core phrase might never actually be played with all the parts unmuted!

Phrases need not be of the same length. One useful approach is to record two complete sets of verse and chorus or a single repeat of an entire section and to use these when you need both hands to play a synthesizer part on a real keyboard. Making a complete run-through of the song available as a song can be used as a stand-by should something go wrong and you do not have the time to drive the groove box directly.

Most groove boxes allow you to choose the next phrase before the current one has completed, although an 'Opcode Vision' type-ahead can be overly prescriptive if too many phrases are entered at once since this removes any possibility for user control during the performance. Having selected the intro, and then moved to the verse, the pitch control (usually a set of pads or buttons laid out as per a music keyboard) can be used to transpose the verse to whatever 'cycle of fifths' or 'last repeat key change' variant the user chooses. Drum parts are almost always set so that they are not transposed, although this can sometimes be a useful special effect.

Other real-time controls that can be used in performance include ribbon controllers or pitch wheel that can be used to change the playback pitch or speed. Restarting the playback of the phrase before it has finished, or playing it at half or double speed is also found. Some groove boxes allow a pair of adjacent or a run of steps within a phrase to be cycled until the control button is released. Arpeggiators can be used to provide variations to parts, or even to generate bass lines. Muting individual parts of phrases can change their character, and there may be several mute 'memories' available. Using these controls requires that the phrases, their location, duration, key, tempo and purpose are all familiar and that the performer has mastered the required timing for the controls. Often buttons need to be pressed just slightly ahead of the beat if they are to work correctly, and this requires practice and experimentation. In particular, most groove boxes use several different 'modes' of operation in order to allow all of the controls and selections to be made, and the performer needs to be aware of the current mode before pressing any buttons.

Live performance using a groove box can also be augmented by the use of an effects box or a ribbon type of controller. The Korg KAOSS Pad (now in its third version) is one example that combines an effects unit that is designed to exploit tempo and allows real-time changes to the effects or the groove box sound generation using a 2D touch-sensitive control pad.

5.22 Dance, clubs and DJs

DJs have changed their role over the last few decades. In the 1970s, they were anonymous people who played vinyl records, and the sequencing of the records, plus a little linking patter over the transition from one record to the next, was all that was required for most performances. Most of the vinyl records were singles lasting only about 3 minutes. Despite the short length, with sufficient patter to pad out the gaps between the music, only one deck was required. In the 1980s, more interaction was introduced as scratching turned the turntable from a playback device to a performance instrument. Pairs of Technics SL-1200 Mk 2 turntables connected by a special mixer with a cross-fade slider to mix from one turntable to the other became the accepted standard equipment.

Transitions between records became more important, and by the end of the 1990s, a DJ was a music maker rather than a mere player of records. The tempo of adjacent tracks would be expected to be the same, and synchronized to each other so that when the cross-fade slider was moved from one turntable, the beats did not syncopate (unless this was the required effect). Scratching techniques would be used to extemporize around the material on the record, and samplers might be used to augment the available sounds from the two turntables. DJs increasingly became creators of music rather than replayers (Figure 5.22.1).

DJs of the 2000s are now skilled, named musical artists, and are capable of performing for several hours with perfect synchronization between the

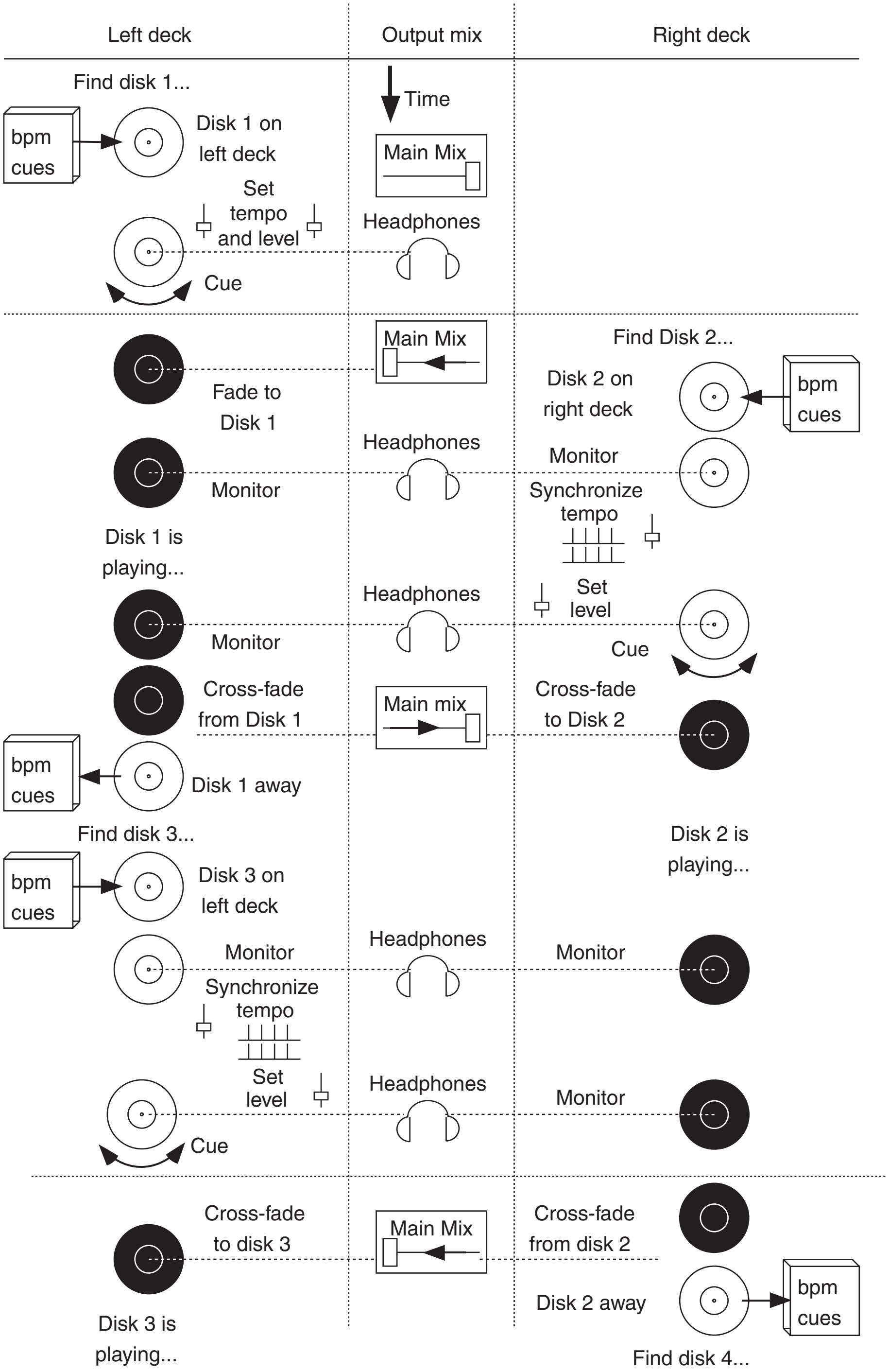


FIGURE 5.22.1 The workflow of a DJ playing vinyl disk in sequence requires a complex set of activities to be carried out in sequence. The lower section between the dotted lines is repeated with new discs for each repetition.

ever-changing vinyl records on two turntables, hitting exactly the part of the record and being on the correct beat every time. The tools they use are becoming increasingly sophisticated and tailored to the genre like the specially designed sampler units which can store and replay music or effects on demand. One of the distinguishing features of many pieces of DJ equipment is the lack of any MIDI sockets, something that has become almost a standard part of electronic musical equipment. But some devices can work in both environments: the Korg KAOSS Mixer takes a 2D touch-sensitive effects controller and embeds it in a two channel cross-fade DJ mixer to produce a powerful live performance device.

5.23 Sequencing

Sequencing in a digital environment has many forms. Workstations, groove boxes, drum machines and computers may all have built-in sequencers that contribute to the final music. This can make backing up difficult and can complicate the recording process, although MIDI files can be used to transfer from one sequencer to another, or as a last resort, the output of one embedded sequencer can be recorded by another.

If multiple sequencers are going to be used, then one should be allocated as the master, and MIDI used to distribute clock and start/stop messages to the slave devices. Changing the MIDI Clock or Sync source from 'internal' to 'external' is not always easy to do, and it is worth making sure that you have the appropriate manuals and have practiced making the changeover from self-sync to external sync, and back again.

Note that most MIDI devices will not indicate that they are set to external sync and are thus waiting for a MIDI Clock message or MIDI Start message they will just not play until they receive the message. Pressing the local 'Start' or 'Play' button on the device will not do anything either, since the device is waiting for a MIDI Clock message. But if pressing the local 'Start' or 'Play' button does start playback, then that device is set to internal sync and will probably ignore any external MIDI Start or Stop messages.

Setting multiple devices to internal sync and then getting fellow musicians to all press the 'Start' or 'Play' buttons at the same time is not recommended; although the timing in most digital equipment is good, the variations in timing are likely to cause the various devices to go out of sync, even though their internal clocks are set to the same tempo.

5.24 Recording

Recording digital musical equipment can vary from a solo live performance to a multi-device, synchronized ensemble piece with many devices all contributing to the mix, or even the traditional 'record a few tracks at a time' approach.

The breadth of sounds available from digital instruments means that it is best to treat them as real physical musical instruments and to check levels and EQ for each change of instrument. Perceived loudness can alter when music is heard on unfamiliar speakers, or when the sequences have been developed using headphones to do the monitoring. Replacing sounds used for testing arrangements in a home studio with sounds in a recording studio may not be as straightforward as it seems, since a slight change of sound may alter the context.

5.25 Performing – playing multiple keyboards

Looking back almost 50 years, it is now difficult to appreciate just how fundamental and far-reaching the effect of placing one keyboard alongside, or on top of, another keyboard was to keyboard players. There are four main types of performance criteria that changed:

1. sounds
2. polyphony
3. playing technique
4. controllers.

5.25.1 Sounds

Having more than one keyboard available means that the keyboard player can produce more than one sound and can even play two sounds simultaneously. For a keyboard player who played piano, this was a fundamental change in mental attitude the sound palette was no longer just the piano. More crucially, it allowed the keyboard player additional control over how they sounded, and how they played the notes. For example, if a piano and an organ were available, then a chord could be played on the piano or on the organ, or could be doubled on both, or any split of notes could be played on the two instruments or any note from an inversion of a chord could be played on the other instrument. This opens up a number of new ways of emphasizing harmonies, holding suspensions and hocketing arpeggios, and it allows the keyboard player much greater control over how the music is arranged.

In the case of a monosynth being added to a piano or an organ sound, then the contrast of timbre is very strong because of the familiarity of the piano and organ sound contrasted to the more unfamiliar timbre of the monosynth. The analogue synthesizers in the late 1970s and early 1980s predominantly used low-pass filters, and this means that they only pass high frequencies when the filter is wide open. The result can be a sound spectrum that emphasizes the lower frequencies. In contrast, the digital synthesizers from the mid-1980s onwards, like the DX7, often produced outputs with more formant-style or band-pass spectrums, and therefore could be used in combination with analogue synthesizers and still be heard in the mix.

It is interesting to note that the most successful keyboard of the late 1980s, the Korg M1, and arguably the most sophisticated keyboard of the early 1990s, the Korg Wavestation, both had low-pass filters without resonance.

The other characteristic that was widely exploited with the early analogue synthesizers was the resonance of the low-pass filters. When a resonant filter is just at the point of breaking into self-oscillation, then a very distinctive tone is produced that is not normally found in natural instruments. Excessive use of the unusual quickly renders it boring and familiar, at least for one cycle of musical fashion.

5.25.2 Polyphony

Pianos and organs are naturally polyphonic. In fact, since you can play all of the keys simultaneously, and every key will produce a note, then they could be termed omniphonic. In contrast, most synthesizers will only play a limited number of notes simultaneously. The very first synthesizer designed to be used as a live performance instrument was the Minimoog, and this was intended to be a monophonic solo instrument. The keyboard player would thus have to learn a rather different way of playing a keyboard in order to use an instrument that could only produce 1 note at once. The task was complicated because the design of the keyboard circuitry of early monophonic synthesizers was such that they always played the highest note being held down. This meant that if a chord was played, then the highest note would be the only one that would sound, assuming that the fingers all pressed the keys at the same time. If not, then the synthesizer would play one or more grace notes ascending up to the final highest note. A similar set of short notes would also appear if the fingers were not removed from the keys simultaneously. For the same reasons, legato playing of runs of notes would have different timing when ascending or descending the keyboard, because the higher note would always sound, regardless of any other notes that were being played.

The initial reaction to all these unwanted notes and timing changes was for the keyboard player to pick at notes with the fingers of the right hand with a slight staccato to avoid any overlapping notes. With practice, this initially unfamiliar technique could be mastered, although this was only the first of several specialized performance techniques that would be required to make the most of a monosynth. The second technique is the deliberate use of the left hand to hold down notes whilst the right hand plays. When the right hand is not playing a note, then the monosynth will play the left-hand-held note. This is rather like the open string on a guitar that sounds when the string is not held against a fret, except that in this case, the left hand can select any note to act as the 'open' note. In many cases, the left hand would play the root note of the chord, and the combined effect of both hands playing would be almost like two separate instruments the left hand playing a relatively static bass note, whilst the right hand played the melody.

A more demanding monosynth technique uses both hands playing staccato, but with only 1 note being held down at once, with control of the note that is sounding passing continually from one hand to the other. Skillful use of both hands in this way can produce startlingly complex runs of notes from a monosynth, although the modern alternative of using a polyphonic synthesizer and a sequencer to store the notes is far easier, if less impressive. Perfectionist synthesists might consider adding an exercise based on this dual staccato monophonic playing to their warm-up routines …

Using both hands to play a monosynth that can only play a single note may seem extravagant, but two-handed playing is almost essential in order to exploit the full expression capabilities offered by a monosynth. This will be explored further in the following sections.

5.25.3 Playing technique

The keyboards found on analogue monosynths were based on organ keyboards, and so were light, springy and responsive. They were thus familiar to organ players in one way, although achieving the right balance between the fixed volume monosynth and the 'swell pedal' controlled organ was not always easy.

For piano players, the analogue monosynth represented a keyboard with no action and almost no weight, and most importantly, no velocity sensitivity. This again meant that the balance between the fixed volume monosynth and the velocity-sensitive piano was critical.

In both cases, the simple solution was to adjust the output level of the monosynth, and this is another reason why both hands are frequently required to play a monosynth. Making quick adjustments to volume also looks good on stage, since it is easily misinterpreted by an audience as a far more demanding technical adjustment.

Additional complexity arose when polyphonic synthesizers became available in the late 1970s. Polysynths normally have organ-style keyboards, but are velocity sensitive. Therefore piano and organ players are both faced with an instrument that has an unfamiliar user interface.

In an attempt to find a solution that would please both types of player, some manufacturers started to add small metal weights to the underside of the keys on polyphonic synthesizers and MIDI master controller keyboards. The Poly Moog and Kawai K5 are just two examples of this type of 'weighted' organ-style keyboard. The 'weighted' keyboards were not popular, since they did very little to emulate a real piano action. Many manufacturers now put piano action keyboards onto pianos, polysynths and MIDI master controller keyboards, especially where keyboards with more than a five-octave span are fitted, or where the piano action will be more familiar to the player.

After-touch is a keyboard controller, which was unfamiliar to both piano and organ players, and it comes in two variants: monophonic and polyphonic. Polyphonic after-touch, where each key has independent sensing of the pressure applied to whilst the note is held down, is very rare. Paradoxically, the Yamaha CS80, one of the first commercial polysynths, had polyphonic after-touch, but only a few later polysynths had it. Monophonic after-touch uses a single pressure-sensitive bar under the whole of the keyboard, and pressing down on any key produces a global after-touch sense.

It is worth noting the differences between velocity (how quickly you depress a key) and after-touch (how hard you press the key once it is being held down). After-touch is normally monophonic, and therefore the whole of the keyboard is affected by how you press a key down. Velocity is normally polyphonic, and therefore each key press can control the timbre of that note. If a keyboard is not velocity sensitive, then since each key press would produce the same velocity value, it could be considered to be 'monophonic' in terms of velocity sensitivity, but this is never normally used in specifications.

Tying to use after-touch to control the timbre of a note during live performance is not straightforward, especially in fast fluid runs, and there are a number of techniques that can be used to overcome this problem. Some players developed a two-handed technique for use with monophonic after-touch-equipped keyboards. The right hand plays the melody notes without any attempt to press the keys to produce after-touch effects, whilst the left hand plays the root note or another harmonically related note as a drone or accompaniment. If the after-touch is not very sensitive, then a variation is to hold the key and the underside of the keyboard casing between the thumb and other fingers of the left hand and use this grip to activate the after-touch. A simpler solution is to remap the parameter and use a modulation wheel or foot pedal to achieve control over the timbre.

In general, a synthesizer with a five-octave keyboard is likely to have organ-style keys, whilst a keyboard that is wider than five octaves is more likely to have a piano action (Table 5.25.1).

Portamento

In a variation of the two-handed 'open' low-note technique mentioned earlier, portamento can be added to a performance on a monosynth by deliberately playing a low note, followed by a higher note to emphasize the glide effect. The portamento effect is less noticeable when subsequent notes are played close together. In performance, this two-handed 'leap' is sometimes replaced by a variation where one hand spans an octave width with the low note initially held down, and then the hand pivots to play the note one octave up and emphasize the portamento. In this way, the audible effect is similar to the portamento being controlled by a foot switch, but requires less co-ordination of hands and feet. Later instruments sometimes provided 'fingered' portamento where staccato notes were unchanged, but legato playing would add in portamento – an acknowledgement of the early performance technique.

5.25.4 Controllers

Sustain pedal or foot switch

Organs lack a sustain pedal, and the effect of a sustain pedal on a real piano is different from the sustain pedal or foot switch on a synthesizer. Sustain on a

Table 5.25.1 Keyboard Features

	Velocity Sensitive	After-Touch	Piano Action
Organ	No	No	No
Piano	Yes	No	Yes
Monosynth	Yes	Yes	No
Polysynth	Yes	Yes	For wide keyboard version

synthesizer is equivalent to holding notes down, and so can be used as part of playing technique for producing drone notes or held chords, whilst the hands move to another keyboard to provide accompanying notes to the held note or chord. On a darkened stage, sustain pedals and foot switches can be difficult to find, and therefore many keyboard players will adjust some sounds so that they have long release times, thus removing the need to use a sustain pedal to lengthen notes. This programming technique is particularly effective on pad sounds with long attack times, but spectacularly ineffective for percussive sounds where the sustain is used to add legato to specific transitions between notes for effect.

Pitch bend

The pitch-bend control as a performance control first appeared on analogue monosynths suchas the MiniMoog in 1969. Although it is possible to bend the pitch of some hammered mechanical instruments where the key remains in contact with the string (the clavinet is a notable example), pitch control through a wheel was not part of pre-existing piano or organ playing technique back in the late 1960s. As with most monosynth performance techniques, the approach that evolved used both hands and consists of the right hand playing the keyboard, whilst the index and middle finger of the left hand are used to control the modulation and pitch-bend wheels. The amount of pitch bend applied to notes and the direction of pitch change were initially derived from listening and observing guitar players. The general rules are as follows:

- Normal setting of pitch bend is a semitone.
- Pull the pitch down by a semitone, then play the note as you restore pitch back to normal.
- When you hold a note in the middle of a phrase, bend the pitch up and down again by a semitone or less.
- When you hold a note at the end of a phrase, bend the pitch down and up again by a semitone or less.

Pitch bend is often applied in place of a grace note, especially in percussive sounds where retriggering the note would cause an undesired repeat of the start of the note.

Although the pitch-bend wheel has become almost a standard, there are still some manufacturers who have replaced it or augmented it in various ways. The Multi Moog used a ribbon controller, whilst two-axis joystick controllers have been used by Korg.

Modulation

The modulation wheel is often used to apply vibrato or tremolo to a sound, and the normal point of application is when a note is held in the middle of a phrase, at the same time as an upwards pitch bend is being applied. Although the modulation wheel is almost always assigned so that it produces vibrato or tremolo, it can also be used to control parameters like filter cut-off or other timbral changes, or even effects mixing, pan position or LFO speed. Keyboard players use very specific frequencies for vibrato and tremolo, and tend to set them to fade in automatically at about the same time. Assigning the modulation wheel to LFO speed with an auto-fade modulation setting allows the modulation wheel to adjust the speed of the vibrato or tremolo instead. A small change of the rate of LFO modulation can be very effective, and is also used in Leslie speaker emulations, where it simulates the non-instantaneous change in speed of the motor as it changes between the slow and fast settings.

Front panel controls are there for two reasons. One is for programming sounds. The other is for adjusting sounds whilst playing. Using front panel controls as controllers can be a very effective way of adding extra expression or variation into a performance. Changing the detune of oscillators can change the mood of a bass sound, and following the mood of the music by adjusting or 'riding' the filter cut-off can produce very flowing lead-lines. Unlike the regular repetition of an LFO modulation, human-generated changes to front panel controls can be much more irregular, or restricted to bar or phrase divisions.

5.26 Examples of digital synthesis instruments

5.26.1 Casio CZ-101 – waveshaping (1985)

The Casio CZ-series of synthesizers are one example of a commercial use of a full waveshaping implementation to produce sounds.

Although it is called 'phase distortion', it uses waveshaping, but presents it in a way which is intended to emulate the operation of an analogue synthesizer. Two DCO oscillators provide the raw pitched sound source, and two parallel sets of modifiers follow. Each DCO has a separate EG for controlling its pitch, although vibrato is provided by a single LFO. The DCO output passes through the digitally controlled waveshaper (DCW), again with an associated EG, and finally through a digital VCA or digitally controlled amplifier (DCA). Ring modulation and noise can also be added. By using just one of the two sets of DCO and modifiers, the polyphony is doubled.

The DCW or waveshaper is designed to behave and sound much the same as the VCF found in an analogue synthesizer. As the control value increases, harmonics are gradually added to the sine wave, so that it changes into one of the eight waveforms, and this can be controlled by an EG as well as tracking the keyboard note. This implies that the transfer function is changing dynamically, which would suggest that a great deal of complex processing is being carried out. However, by working backwards from the waveform, it is possible to work out what is really happening. Because waveshapers tend to add harmonics, not take them away, then the only way that a sine wave can be produced is if the basic waveform at the input to the waveshaper is a sine wave. The waveshape selection is thus used to change the transfer function of the waveshaper, not the waveform produced by the DCO. The waveshapes shown represent the final output of the waveshaper when the full range of the transfer function is being used. The waveshapes that are provided reinforce this the sawtooth, square and pulse shapes are joined by a 'double sine', 'saw pulse' and three 'resonant' waveshapes (Figure 5.26.1).

5.26.2 Roland JD-800 (1991)

The JD-800 is a 24-note polyphonic S&S synthesizer with CD-quality samples, and an intriguing user interface: nearly 60 sliders and nearly 60 buttons, lots of LEDs, two LCDs and one LED display. Each slider is dedicated to a single function, reminiscent of early analogue synthesizers.

5.26.3 Yamaha SY99 and SY77 (1991, 1990)

The RCM synthesizers incorporate advanced versions of Yamaha's FM and Sampling (AWM) technologies, as well as a way of using samples inside FM called RCM. Both methods incorporate detailed control over the source and modifiers resonant filters can be used to process the samples and the FM, which makes the FM synthesis more powerful since dynamic timbral changes are not only controlled by the modulator envelopes. The built-in effects sections provide a wide range of chorus, reverb, EQ and echo effects. The SY99 also provides user RAM for storing samples that can be loaded from disk or through the MIDI Sample Dump Standard (SDS) (Figure 5.26.2).

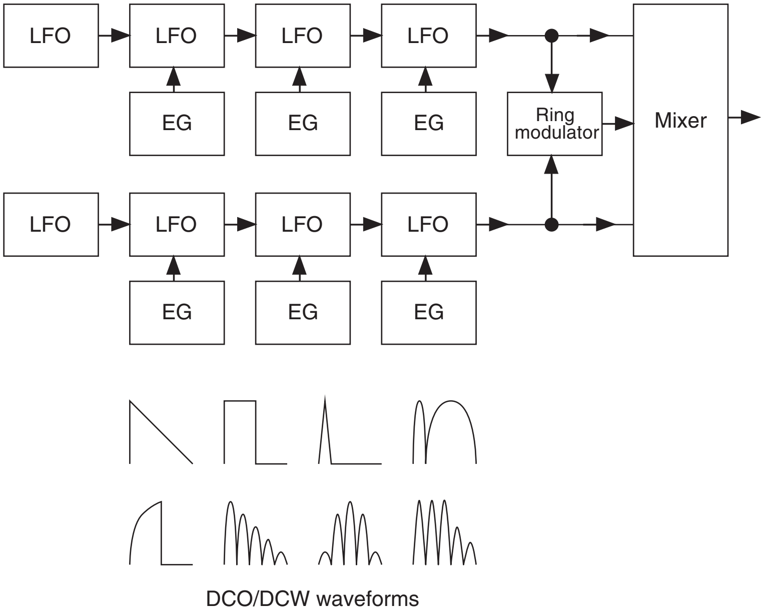


FIGURE 5.26.1 The Casio CZ-101 uses waveshaping, but presented to the user as a DCO followed by a DCW (a digitally controlled waveshaper). The waveshapes provided include the sawtooth, square and pulse waves of conventional synthesis, plus five more unusual ones.

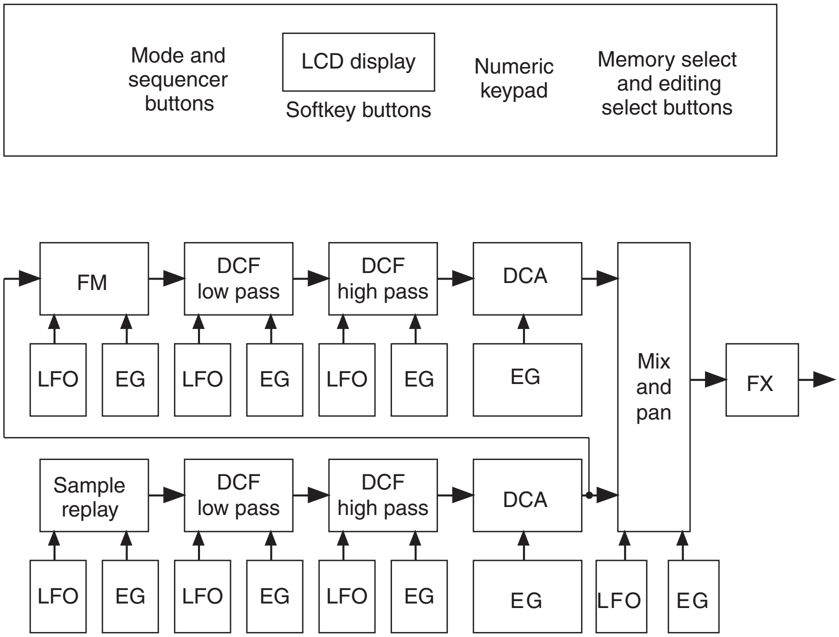


FIGURE 5.26.2 The Yamaha SY99 gives a comprehensively equipped set of FM and sample-replay synthesizers, but allows the samples to be reprocessed through the FM.

Note that the sample processing capabilities had advanced considerably since the early S&S instrument like the Roland D50 (see Figure 4.6.2).

5.26.4 Yamaha VL1 (1994)

Subsequent VL-series instrument, notably the VL70m, allowed editing of the parameters through computer. But this was non-intuitive and arguably harder to understand than FM.

The VL1 is designed as a performance instrument and provides duophonic sounds. It uses preset models of instruments (both real and imaginary) and allows them to be controlled through instrument controls; no user editing of the models is allowed. Although it uses a conventional keyboard with velocity and pressure sensing, as well as pitch-bend, dual modulation wheels, pedals and breath controller inputs, these can be mapped to a large number of instrument controls, including the following:

- pressure (or bow speed)
- embouchure (tightness of lips or bow pressure on the string)
- pitch (the length of the tube or string)
- vibrato (affects pitch or embouchure through an LFO)
- tonguing (simulates half-tonguing damping of saxophone reed)
- amplitude (controls volume without changing the timbre)
- scream (drives the whole instrument into chaotic oscillation)
- breath noise (adds widely variable breath sound)
- growl (affects pressure via an LFO)
- throat formant (simulates the players lungs, throat and mouth)
- dynamic filter (controls the cut-off frequency of the modifier filter)
- harmonic enhancer (changes the harmonic structure of the sound)

- damping (simulates air friction in the tube or on the string)
- absorption (simulates high-frequency loss at the end of the tube or string).

As you can see, most of the controllers are very specific to real instruments, although the parameter that is being controlled may be one that does not and cannot exist! There are individual scaling curves and offsets for each controlled parameter, and therefore you can adjust the effect of a controller like breath control to do exactly what you want with great precision.

The outputs of two separate instrument models can be combined and then processed through the user-programmable modifiers:

- harmonic enhancer
- resonant dynamic filter (low-pass, high-pass, band-pass and notch)
- five-band parametric equalizer
- impulse expander
- resonator.

Although the VL1's self-oscillating virtual acoustic synthesis (S/VA) physical modeling synthesis is designed to synthesize real monophonic instruments, the companion VP1 was intended to produce polyphonic synthetic timbres, and uses a different variation of physical modeling called free-oscillation virtual acoustics (F/VA) and this was probably based on something like the Karplus–Strong algorithm for producing plucked and struck sounds. The VP1 never saw a commercial release (Figure 5.26.3).

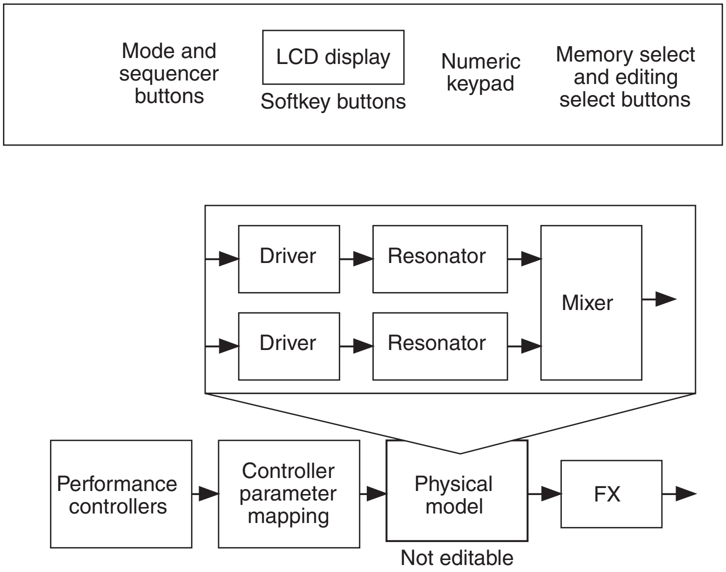


FIGURE 5.26.3 The Yamaha VL1 uses a fixed physical model to produce the sounds, but the user model via MIDI controllers is very sophisticated.

5.26.5 Technics WSA1 (1995)

The WSA1 takes the best parts of physical modeling and combines them with the familiar parts of S&S. Rather than model a complete instrument, it takes the driver and resonator split, provides preset driver 'samples', connects them to a programmable resonator and includes physical model-like interaction control of the resonator. The output of the resonator passes through a conventional digitally controlled filter (DCF) and amplifier (DCA) synthesis section.

The driver 'samples' in the WSA1 are not really equivalent to the samples you find in S&S synthesizers. They do not have the same emphasis on length/ time and multi-sampling that conventional samples have because of the use of resonators to modify the sound of the driver 'sample' in a way that would normally require multi-samples. The driver/resonator model works extremely well – the bass and snare drums are an excellent example where although the basic driver sounds usable on its own, putting it through a resonator suddenly makes it more 'drum-like' (Figure 5.26.4).

5.26.6 Korg Z1 (1997)

Korg's Z1 is a 12-note polyphonic modeling synthesizer derived from Korg's OASYS development computer. The oscillator section provides 13 different types of combinable modeling module including analogue modeling, FM and

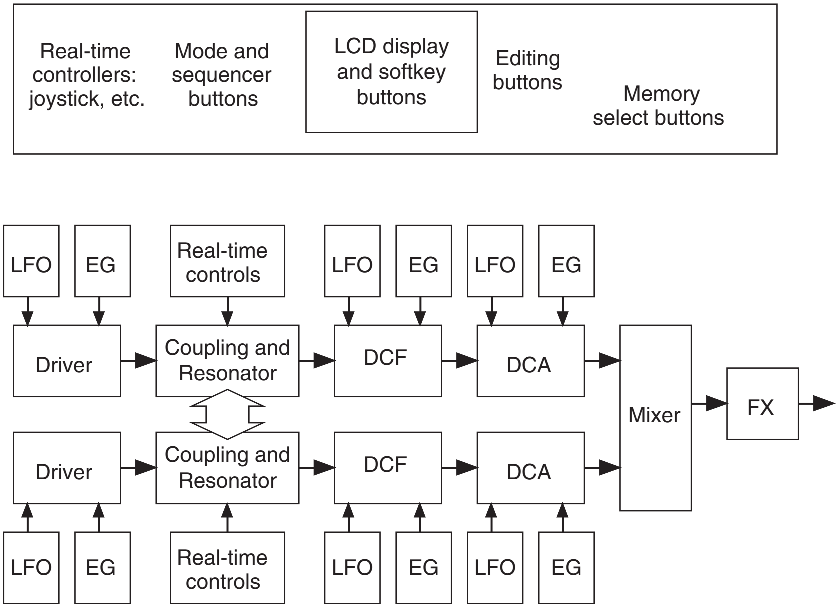


FIGURE 5.26.4 The Technics WSA1 uses source-filter synthesis to provide 'physical model' like capabilities. Two of the four synthesis sections are shown, and the resonators in each of these can be coupled to provide more complex resonators. The two front panel real-time controllers provide 'live' user control over the timbre: a conventional joystick and tracker ball.

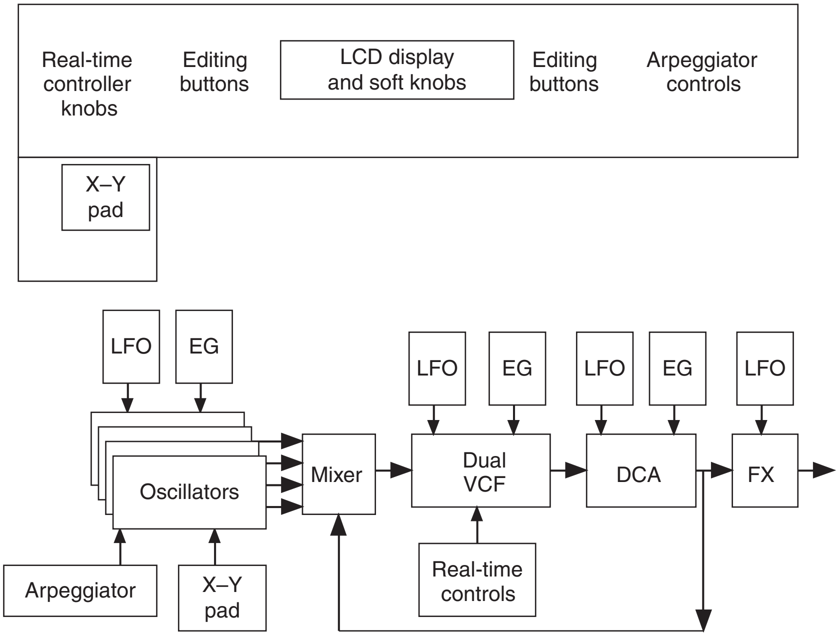


FIGURE 5.26.5
Korg's Z1 is a 12-note polyphonic modeling synthesizer.

physical modeling. The dual multimode filters provide sophisticated control over the timbre and are controlled by dedicated control knobs; also have digital multi-effects and a polyphonic arpeggiator; storage through a PCMCIA (PC card) flash memory card. The display has five assignable knobs, and there is an X–Y controller pad above the compact Prophecy-style pitch-bend and modulation wheels (Figure 5.26.5).

5.27 Examples of sampling equipment

5.27.1 Ensoniq Mirage (1985)

The Ensoniq Mirage was the first affordable commercial sampler (Figure 5.27.1). Although only monophonic and with a grainy 8 bits of sample resolution, with 8-note polyphony and a very restricted memory (2 seconds total sample time at 15 kHz), this instrument changed synthesis and ushered in S&S instruments and samplers. In contrast to the basic sample replay that might be expected from a first instrument from a new company, the Mirage has LFO modulation and separate filter/VCA envelopes, with a velocity-sensitive keyboard. The user interface is minimalistic, with a two-digit LED display and yet there are plenty of features. Up to 16 multi-samples can be assigned across the keyboard, the low-pass filters have a resonance control and keyboard tracking and samples can be looped. There is also a simple sequencer too!

FIGURE 5.27.1
Ensoniq Mirage.

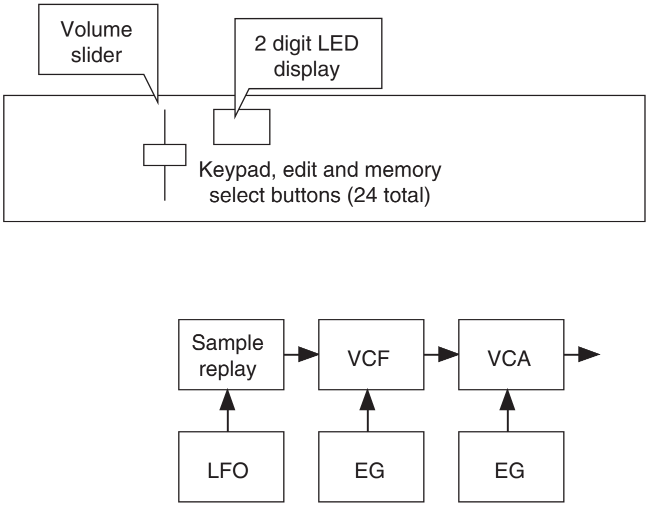
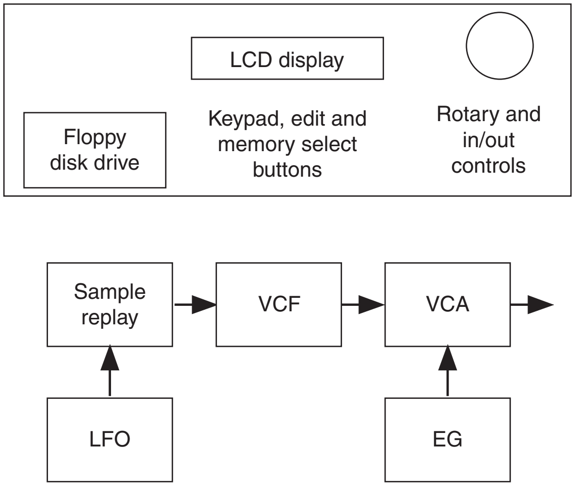


FIGURE 5.27.2
Akai S900.


LCD display
Floppy disk drive
Keypad, edit and memory select buttons
Rotary and in/out controls
Sample replay
VCF
VCA
LFO
EG

5.27.2 Akai S900 (1986)

The Akai S900 (Figure 5.27.2) was probably the first serious rack-mount professional-quality sampler. Although only 12-bit and 8-note polyphonic had the facilities (like eight individual outputs) and software to make it almost a 'de facto' standard for sampling for several years, with the floppy disk format used for the samples also acquiring the status of a common exchange medium. The S900 had a maximum sample rate of 40 kHz giving 12 seconds of high-quality monophonic sampling. Up to 32 samples could be assigned across the keyboard, and it provided facilities such as velocity switching/cross-fading and cross-fade looping. Complete setups of the sampler can be saved onto disk, with 32 of these available at any one time.

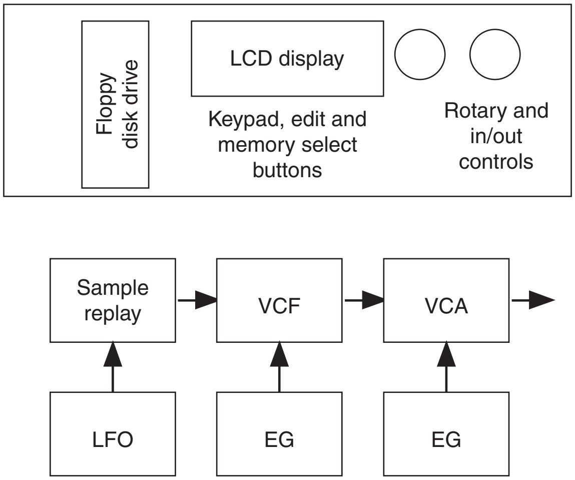

Floppy disk drive
LCD display
Keypad, edit and memory select buttons
Rotary and in/out controls
Sample replay
VCF
VCA
LFO
EG
EG

FIGURE 5.27.3
Akai S1000.

5.27.3 Akai S1000 (1988)

The Akai S1000 (Figure 5.27.3) introduced 16-bit stereo sampling with 16-note polyphony, eight separate outputs, increased memory size, additional controls on the front panel and sample compatibility with the S900, as well as an optional SCSI hard disk interface. Sampling time on the standard model was almost 50 seconds at 22 kHz for monophonic samples. The modifier section looks much like an analogue synthesizer, with LFO modulation, and separate filter VCA envelopes. The sample playback control includes all the cross-fading/velocity switching, looping points and forwards/backwards/alternating loop modes (and more) that you would expect on a second-generation professional sampler.

5.27.4 Akai CD3000 (1993)

The Akai CD3000 was a 16-bit stereo sampler, which allowed sampling from the built-in CD/CD-ROM drive. This was a reflection of the growth in popularity of sample CDs.

5.27.5 E-mu Emulator Four (1994–1999)

A sampler with very sophisticated synthesis capabilities, including a z-plane filter similar to the one found in the same company's Morpheus synthesizer. One of the last of the high-end hardware samplers, and very software oriented. E-mu's subsequent samplers chased the low-cost market until it vanished as computers took over sample replay.

5.27.6 Roland VP-9000 (2000)

The Roland Variphrase VP-9000 provided almost complete independence of time and pitch, with very sophisticated (and patented) signal processing giving huge control and flexibility of working with samples. Computer-based software has since acquired much of the functionality.

5.28 Questions on digital synthesis

1. Compare and contrast the major features of analogue and digital methods of synthesis.
2. What are the two common artifacts that can result from digital synthesis?
3. What happens to the output of the carrier oscillator as the level of the modulator oscillator is increased in an FM synthesis system?
4. What are the three basic parameters that define a static FM timbre?
5. What is the difference between waveshaping and a guitar 'distortion' pedal?
6. What is important about the relationship between the input frequency and the additional frequencies that are produced by a waveshaper?
7. Why is an audio signal always sampled at a rate of at least twice the highest frequency component?
8. Taking one sample of each note on an instrument is one approach to obtaining maximum realism from a sample-replay instrument. Suggest an alternative way of using several samples to enable accurate reproduction of the dynamics of an instrument.
9. Describe one way of splitting a musical instrument into separate parts that may be useful in producing a physical model of the instrument.
10. What is the connection between FOF, VOSIM and human speech?

5.29 Questions on sampling

1. What are the differences between a sampler and an S&S synthesizer?
2. What is the relationship between the speed of a tape and the playback pitch? What happens when the tape is played backwards?
3. What three electronic devices form the basis of a sampler?
4. What do anti-aliasing filters do?
5. What do reconstruction filters do?
6. What criteria need to be considered when looping a sample?
7. What editing functions would you expect to find in a sampler?
8. Outline the convergence of samplers with S&S synthesis.
9. Describe the limitations of analogue sampling techniques using optical and magnetic storage. Then show how these limitations were then overcome by the use of digital techniques.
10. What electronic devices use analogue sampling? What audio applications have they been used for?

5.30 Questions on environment

1. When would you use a stack?
2. How has the role of the keyboard player changed since the 1950s?
3. Compare and contrast three examples of electromechanical instruments, with three electronic equivalents.
4. What drum patterns would you expect to find in a typical drum machine from the 1960s, 1970s, 1980s, 1990s and the 2000s?
5. How would you go about composing the drum patterns for a medley of songs over the last half-century?
6. How would you use a twenty-first century sequencer to emulate a 1970s' 16-step analogue step sequencer?
7. Who would find a twenty-first century workstation the most familiar: a 1950s organist or a 1950s pianist?
8. How would you use a workstation to produce the live accompaniment for a solo singer?
9. How would you use a groove box in combination with two turntables in a DJ set?
10. Compare and contrast the live performances of two performers: one using mostly hardware and the other software?

5.31 Timeline

Date	Name	Event	Notes
1600	Gottfried Leibnitz	Developed the mathematical theories of logic and binary numbers.	
1642	Blaise Pascal	First mechanical calculator.	Addition or subtraction only.
1694	Gottfried Leibnitz	Devised a mechanical calculator that could multiply and divide.	
1815–1862	George Boole	Father of Boolean algebra, which was used to describe the computations inside a computer.	Symbolic logic-based algebra based on 'true' and 'false' values.
1833	Charles Babbage	Invented the computer – intended for producing log tables.	The electronic calculator eventually made log tables obsolete!
1837	Samuel Morse	Invented Morse Code.	
1943	Colossus	The world's first electronic calculator.	Built to crack codes and ciphers.

(*Continued*)

Timeline *(Continued)*

Date	Name	Event	Notes
1949	C. E. Shannon	Published book 'The Mathematical Theory of Communications', which is basis for subject of Information theory.	Shannon's sampling theorem is basis of sampling theory.
1969	Philips	Digital master oscillator and divider system.	
1971	Hiller and Ruiz	Published 'Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects'	Used mathematical approximations to solve the wave equations for physical modeling.
1973	John Chowning	Published paper: 'The synthesis of complex audio spectra by means of FM', the definitive work of FM.	FM introduced by Yamaha in the DX series of synthesizers 10 years later.
1973	Oberheim	First digital sequencer.	The first of many.
1975	New England Digital	Synclavier was launched. First 'portable' all-digital synthesizer.	Expensive and bulky.
1977	Roland	MC8 microcomposer – a small digital sequencer intended to control modular synthesizers.	Enabled use of timecode for synchronization.
1977	Roland	MC8 microcomposer launched: the first 'computer music composer' – essentially a sophisticated digital sequencer.	Cassette storage – this was 1977!
1977	Roland	MC8 microcomposer – a small digital sequencer intended to control modular synthesizers.	Enabled use of timecode for synchronization.
1977	Samson Box	CCRMA, Stanford. Peter Samson designed the Systems Concepts Digital Synthesizer: additive, subtractive, waveshaping and FM synthesis techniques were supported.	256 oscillators, 128 modifiers (filters, VCAs) and a delay-line effects module for echo and reverb, and output through four audio channels.
1979	Fairlight	Fairlight CMI was announced. Sophisticated sampler and synthesizer.	The start of the dominance of computers in popular music.
1980	Electronic Dream Plant	Spider Sequencer for Wasp Synthesizer. One of the first low-cost digital sequencers.	252-note memory, and used the Wasp DIN plug interface.
1980	E-mu	Emulator – first dedicated sampler.	
1981	Casio	VL-Tone. Rhythm, drums, chords and monophonic synthesizer in a low-cost 'overgrown calculator'.	Electronic music for the masses!
1981	Roger Linn	The Linn LM-1: world's first programmable digital drum machine.	Replays samples held in EPROMs.
1982	Philips/Sony	Sony launched CDs in Japan.	First domestic digital audio playback device.

(*Continued*)

Timeline *(Continued)*

Date	Name	Event	Notes
1983	Philips/Sony	Philips launched CDs in Europe.	Limited catalog of CDs rapidly expanded.
1983	Yamaha	'Clavinova' electronic piano launched.	
1983	Yamaha	MSX Music Computer: CX-5 launched.	The MSX standard failed to make any real impression in a market already full of 8-bit microprocessors.
1983	Yamaha	Yamaha DX7 was released. First all-digital synthesizer to enjoy huge commercial success. Based on FM synthesis work of John Chowning.	First public test of MIDI is Prophet 600 connected to DX7 at the NAMM show – and it worked (partially!).
1984	Kurzweil	Kurzweil 250 provides 2 Mbytes of ROM sample playback.	
1985	Korg	Korg announced the DDM-110, the first low-cost digital drum machine.	The beginning of a large number of digital drum machines...
1985	Yamaha	DX100 (four operator mini-key) FM synthesizer launched.	
1985	Yamaha	DX21 (four operator full size keyboard) FM synthesizer launched.	
1986	Yamaha	Electone HX series organ launched.	Mixture of FM and AWM (sampling).
1987	Casio	Introduced the Casio CZ-101, probably the first low-cost multi-timbral digital synthesizer.	Used phase distortion, a variant of waveshaping.
1987	DAT	DAT (Digital Audio Tape) was launched. The first digital audio recording system intended for domestic use.	Worries over piracy severely prevented its mass marketing.
1987	Julius O. Smith	Published 'Music Applications of Digital Waveguides'.	One of the early practical descriptions of 'waveguide' physical modeling synthesis.
1987	Karplus and Strong	Published 'Digital Synthesis of Plucked String and Drum Timbres'.	The roots of waveguide physical modeling.
1987	Kawai	K5 digital additive synthesizer was launched.	Powerful and not overly complex.
1987	Roland	MT-32 brings multi-timbral S&S synthesis in a module.	The start of the 'keyboard' and 'module' duality.
1987	Roland	Roland D-50 combined sample technology with synthesis in a low-cost mass-produced instrument.	S&S synthesis (Sample & Synthesis)

(*Continued*)

Timeline *(Continued)*

Date	Name	Event	Notes
1987	Yamaha	Yamaha DX7II centennial model – second generation DX7, but with extended keyboard (88 notes) and gold plating everywhere.	Limited edition.
1988	Korg	Korg M1 was launched. Probably the first true music workstation. Uses digital S&S techniques with an excellent set of ROM sounds.	A runaway best seller. Filter has no resonance.
1989	Akai	XR10 Drum Machine was launched.	A digital drum machine using sampled drum sounds.
1990	Korg	Wavestation was launched. An updated 'Vector' synth, using S&S, wavecycle and wavetable techniques.	Powerful and under-rated.
1990	Technos	French-Canadian company Technos announced the Axcel – first resynthesizer.	There was no follow up to the announcement.
1990	Yamaha	SY77, a digital FM/AWM hybrid synthesizer/workstation, mixed FM and sampling technology.	Followed in 1991 by the larger and more powerful SY99.
1991	Roland	JD-800, a polyphonic digital S&S synthesizer.	Notable for its front panel – controls for everything!
1992	Kurzweil	The K2000 was launched. A complex S&S instrument, which mixed sampling technology with powerful synthesis capability.	
1993	E-mu	Morpheus synthesizer module was launched. Used real-time interpolating filter morphs to change sounds.	Sophisticated DSP.
1993	Korg	Oasys prototype was launched.	Very much a prototype, but followed by the Trinity, then the Z1, and then the Triton.
1994	Waldorf	Wave, a powerful hybrid synth with an amazingly large front panel.	Wavetables on steroids.
1995	Clavia	Nord Lead – programmable digital analogue emulation synthesizer with a 'subtractive synthesis' metaphor.	DSPs were used to emulate the sound of an analogue synthesizer.
1995	Roland	VG8 Virtual Guitar System – not a guitar synth and not a guitar controller.	A physical modeling guitar sound processor…
1995	Yamaha	VL1, world's first Physical Modeling instrument was launched.	Duophonic and very expensive.
1999	Korg	Triton, S&S sampler with 62-note polyphony and comprehensive effects.	Six separate audio outputs, plus a Mac/PC serial interface.

(*Continued*)

Timeline *(Continued)*

Date	Name	Event	Notes
2003	Analogue Solutions	Vostok, a briefcase analogue synthesizer with two VCOs and one wavetable oscillator.	Has pin-matrix patch panel harking back to the EMS VCS3 and AKS.
2003	Creamware	Noah, hardware modeling synth with a bias towards analogue, plus a B3 organ.	Discontinued in 2005. Hardware expanders do not seem to be very long-lived…
2003	Dave Smith	Evolver, a hybrid monosynth module/expander with downloadable 128 point wavetables.	Digital oscillators, but analogue filters and lots of modulation facilities.
2003	Roland V-Synth	Roland continued their exploration of the creative potential of sample technology that thinks it is synthesis.	No sequencer! This is a synthesizer merged with a sampler.
2004	Access	Virus TI, a DSP-based modeled analogue synthesizer.	Has nine parallel sawtooth oscillators for a fat 'hypersaw' sound.
2004	Korg	Triton Extreme, HyperIntegrated (HI) S&S workstation.	Has a front panel flap for the sample RAM SIMMs.
2005	Korg	OASYS, a cut-down version of the Oasys system in a keyboard instrument.	Linux Operating System, expandable, open architecture. Could this be the future?
2006	Creamware	Minimax ASB, hardware emulation expander of the MiniMoog.	A plug-in, in hardware!
2006	Dave Smith	MEK, the Mono Evolver Keyboard adds a keyboard to the Mono Evolver.	Also a poly version, the PEK. Do plug-ins make expanders obsolete?
2006	Korg	Radias, powerful mixture of modeling techniques: analogue, S&S, FM, formants and vocoder, in a radical case design.	The case design enables a rack-mount expander to be used as a keyboard synthesizer.
2007	Korg	R3, a sophisticated mixture of synthesizer and vocoder.	A modern revisiting of the 1970s vocoder mixed with powerful modeling technology.
2007	Roland	SH-201, a modeling analogue synthesizer that looks and sounds like something from the 1970s mixed with the 2000s.	Entry-level analogue retro.
2008	Arturia Origin	A prolific plug-in manufacturer produces a hardware synthesizer that allows mixing and matching of their plug-ins.	The start of using plug-ins as atomic units of sound-making, but not the end.
2008	Dave Smith	Prophet 08, a reworking of the classic Prophet 5 for a new century.	It is rare indeed for a synthesizer to get a second edition.

This page intentionally left blank

CHAPTER 6

Making Sounds with Computer Software

CONTENTS

Computer History

Computer Synthesis

Environment

6.1 Mainframes to calculators

Computers were initially used for number crunching in large corporate, education and government applications. Initial work concentrated on the connections between music and mathematics, and this ultimately led to the strong ties that still exist between music and computer science in some of the top universities around the world.

Although music has been made on computers from almost the very beginnings (often as part of demonstrations of processing power in terms ordinary people could understand), the change over the past 50 years has been startling. We have moved from the 1950s, when there were only a few tens of mainframe computers in the whole world, to a world where an ordinary home may contain more than 10 microprocessors, and probably more than one ‘personal’ computer. The concept of a ‘personal’ computer was so alien in the 1950s that it was used as part of the essential equipment of ‘B movie’ mad scientists bent on taking over the world.

Computers have thus moved from a small market to a mass market, and it is hard to imagine many activities without them. Whilst music and computers have been closely connected in academic applications, the ordinary musician had probably never considered a computer for musical purposes until the late 1970s, when the first computer-based sequencers began to appear from Roland. Based on, and perhaps influenced by, the cash register application for the Intel 8080 microprocessor chip, these early sequencers had calculator-style numeric keyboards and limited displays of numbers, but they moved computer-controlled music from consuming precious time on a shared and hugely expensive mainframe computer to something affordable and personal.

6.2 Personal computers

Marketing sometimes creates amazingly far-sighted ideas. Calling a computer a ‘personal’ computer is just such a landmark. Computers in the 1970s were shared resources, with their many users getting access to short time slices of the processor from many terminals, each one little more than a display and keyboard. The personal computer (PC) reversed things, so that a single person could monopolize a whole processor, although you could also connect the PC to a mainframe. This changed expectations away from an expensive shared resource that someone else looked after to a device that one person could own.

From the 1980s onwards, processing power moved from mainframe computers to PCs as businesses moved to a 'one per desk' micro-management mentality for 'personal' computers. The development of the World Wide Web and the browser started a reversal of this trend as the need for shared 'serving' of data over a network became increasingly important. Servers and mainframe computers now provide the power behind the vast processing needs for electronic commerce, banking and other computer functions using the Internet as a means of connection. PCs have, in many cases, reverted to being used almost solely as the equivalent of a terminal to a central processor again, albeit a browser accessing 'The Internet', but little different in functional terms from the mainframe and the terminals of the 1970s.

But some people do use PCs for activities other than surfing the Internet. Word processors and spreadsheets are still used, but the low cost and wide availability of computers has made them accessible for other uses. The expensive specialized computer sequencer and the PC collided in the 1980s, with the release of the musical instrument digital interface (MIDI) specification that standardized intercommunication between computers and musical instruments. Something which had been very difficult with analogue synthesizers now became very easy – connecting on instrument to another and being able to play something on one keyboard using the sound from the other instrument. Digital synthesizers also allowed a further innovation that was beyond almost all analogue synthesizers: program switching over MIDI. Control voltages and gate signals could be used to connect two analogue synthesizers together, provided that they had either the same linear or the exponential format, but controlling the selection of a patch was not accessible, and if it was, then it was by using proprietary cabling between just one manufacturer's equipment. MIDI changed that at a stroke and introduced the stack sound: 1 note producing more than one sound simultaneously.

MIDI also allowed computers to be easily used with synthesizers. Before MIDI, there were a number of ways to produce control voltages and even generate musical sounds, but these were often proprietary and expensive. MIDI allowed any computer to be connected to any synthesizer, just by adding a MIDI interface, and the MIDI interface was deliberately designed to use low-cost standard computer hardware. MIDI interfaces quickly appeared for the home computers of the day: 8-bit microprocessor-based game-playing machines with cassette storage and using TVs as monitors. PCs were still priced for the business market, and it was not until the 1990s that market forces made them affordable.

In the twenty-first century, the market for PCs has settled down to essentially one hardware platform (with an 'x86' processor from Intel, AMD…) with just three operating systems: Windows, Mac OSX or Linux. It seems to be fashionable to be passionate about one particular operating system, but they all have strengths and weaknesses, and you should use whatever one you prefer, since software to do just about anything you want is available for all the three. This could be compared to choosing a grand piano. There are a number of names to choose from: Steinway, Beckstein, Yamaha, Kawai, Young Chang, and so on, but to a member of the general public, they are all pianos, and they all sound like pianos.

6.3 The PC as integrator

PCs are interesting because of the way that they, and the microprocessor chips that they are closely related to, dominate the electronics in everyday life. PCs outsold televisions for the first time in the early part of the 2000s, and television-on-demand supplied by the Internet is increasingly the way that people want to watch television. MP3 audio files are loaded onto PCs from the Internet and then loaded onto small portable computers for playback.

There does seem to be a trend for almost anything that can to move onto a computer. The computer is an amazing general-purpose device, and factors such as convenience and ease of use are probably important, as is familiarity. Blu-Ray players tend to look like DVD, CD or video-tape players, and mobile/cell phones look like MP3 players, which look like the Walkman cassette players of old. Synthesizers, of course, do have a very specific look too.

Therefore not only is the computer capable of doing lots of things, plus it has become a familiar piece of technology, but it also has an interesting property that is described by one of those 'laws' which is actually an observation-Moore's law. Gordon E. Moore, one of the founders of Intel, the manufacturers of many of the computer chips we use, noticed that the number of transistors that could be put into a chip doubled every couple of years, and this reflected an ongoing doubling trend for a number of other important trends and such as processing power hard disk size. The trends have become a sort of goal for the computer industry and have been met or bettered for the last quarter century. It seems that computers are improving all the time: getting more powerful and relatively cheaper.

This is a powerful set of attributes. A toothbrush stubbornly refuses to get better with time, and I suspect it gets worse. A car is not twice as powerful as the previous one, nor does it go faster, or use less fuel. But computer-based devices do get better as a consequence. Digital synthesizers are more stable, have a broader range of sounds, more polyphony and better displays than their analogue ancestors.

But beyond sound-making, there is the environment in which synthesis and sampling are used, which is why this book concentrates on the differing environments in which different types of synthesizer technology were used. Having a computer that can emulate a synthesizer or play back samples is only part of the jigsaw, and it seems that computers are very good at integrating things too.

It could be called the 'Integration hypothesis'. Computers are a one-way street to integration.

Certainly, the effect of computers on electronic music-making has been a steady one of integration. MIDI made it easy to connect instruments together and to store their sounds on a computer. The sequencer on the computer was much easier to use than a multi-track recorder, and you could play back at any tempo without pitch changes. Editing samples is much easier on a large computer screen with a mouse, plus computers have lots of storage and can keep track of all those sample sets and synthesizer edits. The sequencer needs a mixer so that the musical events and the resulting audio can be edited in context, and sample playback means that a hardware sampler isn't needed. Effects units as plug-ins for the mixer mean that the outboard effects aren't needed, and trying to remember all of those effects programs was tricky. Plug-in synthesizers mean that MIDI cabling is no longer needed, and having all of the sounds immediately available without Sysex downloads and librarian software is far more convenient.

Computers have almost integrated music totally. Modern music-making software tends to use words such as 'Digital Audio Workstation (DAW)' or 'Music sequencer'. It is remarkably easy to buy and install a piece of software that has a sequencer, sampler, analogue and digital synthesizers, effects units, mixer, samples of real instruments and drum sounds, and tutorials on how to use it, and which costs a fraction of what exactly the same equipment in real physical form would have cost 10 years ago. This is an astonishing achievement.

6.4 Computers and audio

The early 8-bit computers of the 1980s had simple rectangular wave outputs, produced by setting an output port to one or zero repeatedly. In current audio terminology, this would be described as 1-bit audio. Telephone quality is 8 bits (or 12 bits if you take into account its non-linear dynamics), CDs are 16 bits and pro-audio interfaces will get you 20 bits or more of resolution. The 8-bit computers make sounds that can be described as varieties of 'beep'.

Later, 8-bit computers had more advanced sound chips, and one in particular, the SID (Sound Interface Device) chip found in Commodore computers was special. Devised by Robert Yannes, who would go on to found the Ensoniq digital synthesizer company, the SID chip was effectively a simple subtractive synthesizer chip, with three oscillators, multi-mode filter, ring modulation and envelopes. Not surprisingly, this was considerably more capable in terms of music-making than any other sound chip at the time, and the Commodore C64 became a best seller, in part due to the sound.

The BBC Micro used software synthesis, written in assembler code, to do similar things to a SID chip (but leaving little processing power to do anything else) and this was called 'The Music System'. More sophisticated MIDI control could be achieved by using the UMI sequencer, which was reviewed by the author at the time, and which was then the state-of-the-art.

The 8-bit computers usually had small loudspeakers intended for use more as alerts for errors, indicating the end of a process, or the need to acknowledge something. Audio input was not a standard feature, although as storage was usually on audio cassettes, it could be argued that an audio input of some sort was present. But in general, audio processing in the era of 8-bit computers was done mostly in stand-alone analogue hardware, not in computers.

Section 6.1 mentions how expensive early mainframe computers were used for musical applications, and it is interesting to note that even simple 8-bit home computers were also programmed to make music.

The 16-bit computers followed, and these could actually play back short audio samples with very limited polyphony. MOD files were the control files for these granular sample players, called Trackers, and they were widely used to create video game music. The Apple Macintosh with its, at the time, revolutionary graphical user interface rapidly became a popular MIDI music computer, although the high cost in Europe meant that the lower cost Atari ST computer enjoyed considerable success too.

Some 16-bit computers had audio inputs, usually at the microphone level, and the Atari ST had MIDI sockets – still a unique feature for a mass market uncustomized computer. Floppy disks became the new standard for desktop data storage and were the natural home for MIDI files, as well as becoming familiar on many hardware synthesizers.

The IBM-compatible PC computer had lots of interface slots, although these changed over time. But they did offer a readily accessible way to add audio features. Notable one was a sophisticated MIDI breakout box called the MPU-401. Designed by Roland, this had DIN Sync 24 as well as MIDI In, Out and Thru ports and was widely used and widely cloned. PCs did not come with audio input and output sockets until CD-ROMs became popular, and many companies allegedly resisted adding audio to the specification of their computers because of fears that they would be used to play music. PCs also had the code for a cassette interface left inside their BIOS for many years after cassettes had been abandoned for data storage. When sound cards were added to PCs, one of the popular early cards used a sound chip that was based on a Yamaha FM chip.

Modern 32- or 64-bit computers have CD quality audio input and output, as well as built-in support for MIDI, but MIDI ports do not come as standard. The loudspeakers have improved slightly, but still serve a utilitarian purpose rather than a music one. Separate audio breakout interfaces are recommended to give more bit resolution, line level inputs and better noise floors, and these can be connected through USB, FireWire or PCMCIA (PC-Bus) interfaces or through specialized interfaces that connect into the PCI, PCI-X or PCI-Express bus inside the computer for more demanding applications (more channels of audio). Audio inputs and outputs need not be just analogue: support for electrical and optical digital audio signals like S/PDIF over coax or TOSLINK can be found natively on some computers, or through special audio interface cards for interfaces such as AES/EBU, MADI, ADAT or TDIF. MIDI interfaces can be found on some sound cards, and separate MIDI breakout boxes can provide high-quality ports. USB ports are increasingly used to provide both audio and MIDI connections, although for more demanding purposes, FireWire is used (mLan is one example), and other hardware solutions are available for special purposes. GM-compatible sound sets are included as part of the Mac OSX and the Windows operating systems, and therefore a basic level of audio and music capability is available from a default installation.

The operating system, which provides the basic environment in which all other software runs on a computer, has also developed audio functionality over the years. Both Windows Vista and Mac OSX have (different) low-level audio features that are confusing given the same name: Core Audio. Mac OSX has comprehensive MIDI support through Core MIDI, and support for extending the audio processing: Audio Units. Windows has more basic MIDI support, but XP has the DirectX framework into which DirectShow filters can be placed for extending the audio processing, whilst Vista has the Media Foundation framework into which Media Foundation Transforms can be placed for extending the audio processing.

6.5 The plug-in

The plug-in is a simple idea to solve an immediate problem, but it has far-reaching consequences. Although the general concept had been in use for some years previously, HyperCard, the application toolkit for the Apple Macintosh, was one of the first pieces of PC software to implement plug-ins in a way that would be familiar to users today. HyperCard, released in 1987, allowed users to design applications by working with a card metaphor, rather like a programmable card index. Bill Atkinson, the designer, realized that they would not be able to know in advance all of the functionality that people might require, and therefore an interface was specified so that people could add their own software to augment the functions that came with HyperCard. One of the functions that was missing in HyperCard was support for MIDI, and this was subsequently added by other programmers using the interface.

The word ‘plug-in’ came a year later, in a program called SuperPaint, where additional painting facilities could be added. Up until this point, you used the facilities that were provided in a program and waited until the next update to see what had been added. The idea of a plug-in was that it would be possible for a really determined person to add in that one little missing bit of functionality that the programmers of the ‘host’ software had overlooked… What actually happened was that users did not feel that just a few minor bits were missing, but they would eagerly adopt any extras that were written.

Adobe Photoshop, the photo retouching tool, illustrates this perfectly. There are a great many plug-ins available, for a wide variety of purposes, and it would be impossible for Adobe to have known about all of these requirements, but by providing an interface for plug-ins, the program's functionality could be changed on demand.

One feature that frequently arises with plug-ins is version compatibility. Early plug-ins provided simple functionality, and therefore SuperPaint provided additional special effect brushes, and therefore had simple interfaces. Plug-in programmers tend to explore the limits of the interface provided, and plug-ins became one of the areas that programmers would receive feedback on requests for new interface features. Plug-in interfaces thus tend to increase in functionality with new versions of the parent software. Successful plug-in interfaces can also become standards, for example the Photoshop interface has now been adopted by other software.

In audio, Steinberg introduced their Virtual Studio Technology (VST) in 1996, into Cubase, their flagship MIDI sequencer. The following year, Steinberg released VST and the ASIO audio stream input/output interface as open standards and encouraged programmers to use them. VST 1 allowed the creation of audio processing units that could be added to the mixer in Cubase. Reverb and other effects units were typical early VST plug-ins.

VST 2 (note the increase in functionality as the plug-in interface develops) added MIDI processing ability, which meant that it became possible to take MIDI events and turn them into audio outputs – which allowed plug-in synthesizers and sample players. Steinberg calls this functionality VST Instruments because it allows programmers to make instrument plug-ins.

VST 3 was released in 2008 and is a complete rewrite of the VST code that also adds a number of new features: dynamic processing so that audio processing happens when audio is present, sample-accurate parameter automation, multiple MIDI ins and outs and deeper integration with the host software (Figure 6.5.1).

Other sequencer manufacturers added their own plug-in interface formats, and there are now several variants, usually specific to a particular manufacturer. Some formats are proprietary to their manufacturer and therefore there are no public developer resources, whilst others provide comprehensive developer support to anyone. But plug-ins are often provided in several interface formats, and wrappers allow plug-ins to be used in a different type of plug-in interface.

For example, Digidesign's FXpansion VST to RTAS Adapter allows VST plug-ins to be used in Digidesign and Avid products.

Some plug-in formats are as follows:

- VST from Steinberg (used in their Cubase sequencer).
- MAS from Mark of the Unicorn/MOTU (used in their Digital Performer sequencer).
- Audio Units from Apple (used in their Logic sequencer).
- DirectX from Microsoft (these are DirectShow filters, usable in several Windows-based sequencers).
- RTAS/Real Time AudioSuite from Digidesign (used in their ProTools sequencer).

Most plug-ins are specific to an operating system, therefore a Windows VST plug-in will not work on a Mac OSX computer, but there may well be a Mac

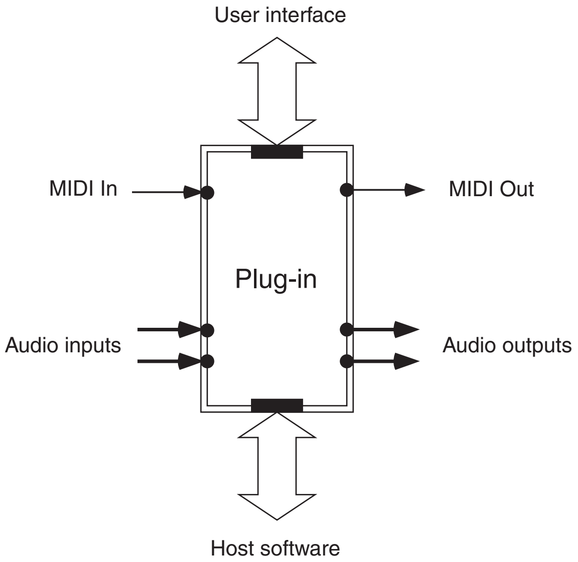


FIGURE 6.5.1 Plug-in overview. Plug-ins hook into the host software in a number of ways.

OSX version of the plug-in. The examples given earlier are not exclusives, and VST plug-ins are particularly widely adopted amongst many software manufacturers. On Mac OSX, Audio Units are increasingly popular and dominant, except for Digidesign products, where RTAS must be used. For Windows, VST is still popular.

You should always check that a plug-in is compatible with your computer, its processing chip (which is normally called the CPU, or Central Processing Unit, and is a terminology left from the days of mainframe computers), its operating system and the host software that you will be using. Some plug-ins come with several different versions to suit different operating systems, computers and host software, but this is not always the case.

Plug-in technology is always changing. In the Windows world, DirectX (DX) plug-ins are being replaced by DMO plug-ins, which are DirectX Media Objects and easier to write and are Microsoft's recommendation to write instead of DirectShow filters. Media Foundation Transforms are the next generation of DMOs.

There are many names for the combined 'audio and MIDI sequencing' software that provides hosting facilities for plug-ins: sequencer, audio and MIDI sequencer, DAW, audio workstations, music workstation, workstation software and more. The term DAW is often used as a synonym for all audio and MIDI sequencing software, but some interpretations specify that a DAW implies the availability of high-quality audio input and output facilities. But many audio and MIDI sequencers can produce MP3, WAV or other digital audio files as their output, and may have no specialized audio output facilities, which means that they are sometimes not classified as DAWs.

Since all of these provide host interfaces for plug-ins, they will be referred to here as 'host software'.

6.5.1 History overview

The first plug-ins were restricted to audio processing and therefore were simple mixer-familiar outboard add-ons such as EQ and reverb. Time-based effects such as echo and flanging followed. Once MIDI events could be processed then simple sample replay plug-ins could also be produced. FM and analogue modeling plug-ins started to move away from simple sample playback towards modeling technologies, and with increased processing power, the mid-2000s saw physical modeling plug-ins that can produce realistic sounding classic instruments like electric pianos and strings. Plug-ins are also available for arpeggiators, audio-to-MIDI converters, librarians and editors for external synths, MIDI parameter remappers, step sequencers, chord generators and more.

Whilst plug-ins appear to be a direct conversion of existing digital software to use on a computer, they share one very significant feature with computer software, and in many ways increase its importance-ongoing updates. Mechanical devices tend to stay in the form in which they are sold, although wear and tear may result in parts needing to be replaced from time to time, but the essential mechanism remains the same. Therefore hammers, bicycles and other items will keep working in the same way. Software, particularly PC software, has gradually acquired a different approach-often called 'continuous beta'. What this means is that the software often requires a lot of development time, and it is hard to remove all the errors and bugs, and therefore it is released in a partially completed state, and the users become the testers.

Testing of a nearly complete piece of software is called 'beta' testing, and in an ideal world, the developers would carry out the initial 'alpha' testing, then it is beta tested by internal testers, and then it is released. The complexity of modern software, plus time pressure, often results in early release of software that has only been partly beta tested, and with some uncertainty about exactly which features should be provided. It is very difficult to remove all of the bugs in a piece of software, and additional features that are needed will frequently only be revealed when lots of people use the software in real-world applications. This is very different to a hammer or a bicycle, where there is no way to change their functionality, and if you want a different type of hammer or a bicycle with a different number of wheels, then you purchase one.

But software is, almost by its very nature, adaptable. Early computers put software into ROMs, and therefore they could not be changed, but modern software is stored on hard drives or in flash memory and can be changed easily. As a result, software is released after enough testing to enable it to be used reliably, but with the expectation that features will be added, and bugs fixed, whilst the software is supported. In synthesizers and samplers, this support often lasts after the hardware has stopped production, and sometimes is taken over by a third party after the original manufacturer has stopped support. Some designers

A replacement operating system for the Yamaha TX16W sampler was produced by a Swedish software company between 1994 and 2000.

have put only essential operating system functions into ROM, and then provide the remainder of the operating system on a removable storage such as floppy disk or CD-ROM. The Ensoniq Mirage sampler (and many other synthesizers and samplers, as well as other electronic devices like the Sony PlayStation One) deliberately put some of the operating system onto removeable storage in this way. This allows the operating system to be improved over time.

In terms of plug-ins and host software to run them inside, continuous beta means that all software comes with a version number, and that, in general, the latest version will have the most features and bug fixes. By partitioning the functionality into host and plug-ins, updates to specific features can be made more quickly because only that feature needs to be changed. Plug-in interfaces thus provide a way for host software manufacturers to enable updates of the features that are provided by plug-ins, without the host software manufacturer needing to do anything at all to the host of software.

But the availability of updated plug-ins is not normally automated, and therefore it is often left to the end user to check that they have the latest version of their plug-ins. With a large number of plug-ins, this can become a large management overhead. One approach is to use updates to host software as the trigger for checking plug-ins. Host software often includes automatic checking for updates, and so whenever the host updates, this is a good time to check for updates to plug-ins manually. Automatic updating of plug-ins may become more widespread in the future.

6.5.2 User interface

Plug-in interfaces provide programmers with access to audio input and output streams, MIDI input and output streams and user interface graphics. Early plug-ins had simple controls, but the graphical sophistication has increased with time.

Most sequencers and DAWs base their user interface on multi-track tape recorders, MIDI sequencersor audio sequencers. Time normally runs horizontally, and the mixer is shown as a vertical set of channels. But there are exceptions.

Propellerhead software, from Sweden, released their Reason DAW in 2001, describing it as a 'virtual studio rack'. Reason's on-screen appearance is a studio effects rack, complete with a rear view showing patch cables. The front view shows a mixer, a step sequencer and lots of spaces for plug-ins. Plug-ins appear as effects units, synthesizer expanders or samplers in the rack and can be patched in as if they were physical hardware.

Ableton software, from Germany, have two views in their Live DAW: a time-based horizontal view and a mixed phrase sequencing and mixer view with vertical channels. But below this mixer is a horizontal area where time editing, enveloping and looping of samples can be carried out, or sequences of effects can be produced by dragging in a series of effects, which then become the audio path for the selected channel of the mixer. Each channel can thus have its own chain of effects. Similar chains of synthesizer and sample plug-ins can be assembled to create complex instruments as well.

Reason's appearance is direct and realistic: it looks like a studio rack and invites you to use it. Live's look is very abstracted and minimalistic on the surface, but it hides great power and complexity underneath. But they both provide enormous flexibility in how you can work with audio and MIDI, and much of that comes from the use of plug-ins as an intrinsic part of the way that the software is designed to be used.

6.5.3 Consequences

With provision for inputting and outputting both audio and MIDI, as well as allowing access to the screen display for the user interface, the software manufacturers have provided more than just a simple flexibility point to allow for oversights in functions, sounds of effects. Instead, they have allowed personal customization by selection of the sounds, functions and effects that people want, rather than what the host software provides.

When synthesizers provided removable cartridges or floppy disks that could store sounds, a marketplace appears where people who could create sounds would sell them, often through intermediaries, to people who did not know how to program sounds, but who wanted different sounds to the factory presets. Plug-ins did the same thing, but this time the effects were more significant because the plug-in interface allowed programmers to get much deeper into the host device. Programming sounds is a very constrained activity, even when you are trying to explore the boundaries and limits to get better or different sounds, but a plug-in interface allows much greater freedom.

The result is that the initial factory plug-ins were rapidly improved upon by third-party plug-ins, and competition between plug-in programmers began to explore the limits of what was possible. The manufacturers of the host software, the sequencers and DAWs wanted to differentiate their products from others, and therefore bundled third-party plug-ins with their software.

Ongoing increases in available processing power, as well as improvements in the facilities offered in the plug-in interface, mean that plug-ins can always do new things, and this sells upgrades of memory, processor power and even new computers or new host software. One interesting physical variation on plug-ins happened in synthesizers in the 1990s, and it shows how rapid the technology was developing. Manufacturers such as E-mu, Roland and Yamaha realized that they had expertize in producing sounds or sound generating modules to put inside their hardware sample players, and that by selling hardware expanders with slots or sockets for those sound ROMs or modules, they could then sell the sounds and modules to people to increase the sound generating possibilities of their expanders. Unfortunately, whilst these sold well, the slots and the sockets were proprietary, and it can be difficult for third-party companies to make the ROMs or modules to go in them. As a result, the majority of sounds were just more factory presets from the manufacturers, and the market did not grow very rapidly.

Creating software for a plug-in interface might require time and effort, but most host software manufacturers publish the details of the interface, and actively encourage programmers to use it. Therefore the software plug-ins, which started out with effects but soon included various types of synthesizers and sample players, quickly offered a large range of sounds and effects that could be used in just about any host software. Computers and host software were sold to people who wanted convenience and lots of flexibility, and hardware was sold to people who wanted restricted choice, incompatibility and lots of MIDI cables and audio wiring.

The author used to be one of the latter.

The Korg MicroX is one example of this type of 'zero-cpu load' plug-in sound source.

One notable consequence of plug-ins is that synthesizer expanders, the keyboard-less modules controlled through MIDI, are losing popularity and many manufacturers are adding keyboard to them so that they can be used as keyboard controllers for live use and computer use. Some manufacturers are integrating these keyboard sound sources with plug-ins so that they appear inside host software like conventional synthesizer soft-synth plug-ins.

6.5.4 Significance of plug-ins

Plug-ins might appear to be just a way of customizing software, but their importance is huge because of what it does to synthesis, sampling and sound-making. Before plug-ins, software did specific things, and intercommunication between different pieces of software was often difficult and had to be done by saving an output manually and feeding it into another piece of software manually. This could be seen as being analogous to sound-making before synthesizers, or even during the heyday of analogue synthesis. Whilst it might be possible to achieve a specific series of functions on an audio signal, or to chain processing functions together to make a sound, this would require considerable time and effort. The triple-Grammy winning classical album, *Switched on Bach*, by Wendy Carlos, is a startling example of what can be done with a multi-track tape recorder and a modular synthesizer, particularly in terms of the amount of time and effort that it must have required.

After plug-ins had been introduced, then things changed in a way that is somewhat analogous to the effects of MIDI becoming widely adopted. Connecting things together became easy, and chains of processing or assemblies of separate devices producing a composite output became possible. Instead of a small number of devices with limited connectivity, plug-ins made it possible to utilize a large number of devices with very broad connectivity. In many ways, plug-ins are the synthesizers of the twenty-first century, and whilst owning half a dozen plug-ins is not unusual, owning half a dozen analogue or MIDI synthesizers would be the mark of a serious performer or recording artist.

But beyond just owning plug-ins, the host software that enables the plug-ins to do useful things is rather like a well-equipped synthesizer-oriented recording studio – it has lots of cables, converters, adapters, mixers and other general-purpose useful audio stuff – and allows just about anything to be connected to anything else in the quest for making sounds. Therefore if you want to have two different reverbs, you get two plug-ins, and access them through the mixer section of the host software of your choice, thus allowing you to send some of the tracks to one reverb, and some to another. Depending on how the software works, it may be possible to arrange the plug-in reverbs in series, so that some sounds can be sent to one reverb, some to be sent to the other reverb and some to both (Figure 6.5.2).

Therefore if plug-ins are the synthesizers, sample-replay devices and effects processors (and more) of the twenty-first century, then the host software is the enabling environment that enables plug-ins to be used in a sound-making way. This is why the main chapters of this book have a section on environment to illustrate the changes with each successive generation of synthesizer or sampler.

Plug-ins can have a very wide range of functions, and any list is going to be incomplete. There are many times more plug-ins now available than models of analogue and MIDI synthesizer and sampler! But the breath of types can be informative:

- reverb and other time delay with feedback-based effects
- chorus and other modulated time delay effects
- analogue synthesizers using modeling technology
- physical modeling-based synthesizers
- sample, DLS/SoundFont and other players
- sample rate converters
- granular, FOF, formant, FM and many other types of synthesis
- arpeggiators, step sequencers and other accompaniment functions.

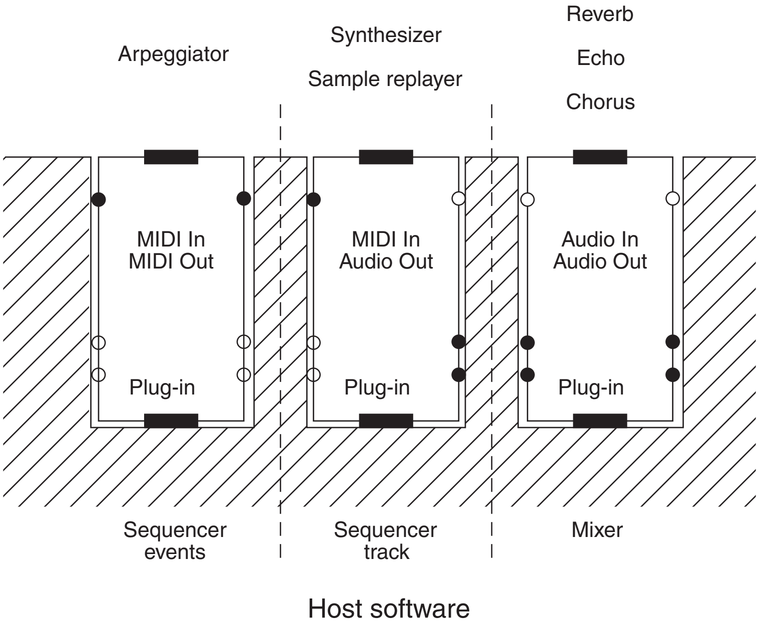


FIGURE 6.5.2 Plug-in functionality. Plug-ins can provide functionality at several places in host software.

Some host software provides a higher level of abstraction of plug-ins by allowing plug-ins to be assembled into chains of effects, stacks of synthesizers or samplers, or even complex mappings of samples and synthesis across keyboard note ranges and velocity. The possibilities for making virtual instruments and effects offered by some host software are way beyond anything that you could realistically assemble in hardware. Best of all, their very virtuality means that they can be saved and recalled whenever they are needed, something which hardware is very weak at accomplishing.

In terms of synthesis and sampling, then plug-ins are available that will provide all of the techniques described in this book, and more. Many of the example instruments are available in virtual form as plug-ins, and if they are not, then there are DIY plug-ins that will allow them to be created, so be creative! In terms of stretching what is possible, assembling plug-ins into larger and more complex abstractions can achieve this, and modular toolkits like Reaktor provide more scope than most people will ever explore thoroughly, which means that those who do are in unexplored territory.

Analogue synthesis was hard to keep in tune, hard to interface to, and hard to turn into something really novel. MIDI and digital made it easier to keep in tune, easier to interface to, but in many ways, made it harder to be novel – The Roland D-50s *Digital Native Dance* is a good example of something that everyone thinks is novel and interesting, and therefore it almost instantly becomes an unusable cliche. Computers are just as free of tuning issues as digital, do not need any interfacing because everything can be in one place or you have MIDI and breakout audio boxes if you need them, and novelty is easier to attain because of the broad available resources and the depth of support for exploration.

6.5.5 Programming plug-ins

In much the same way that describing how to build an analogue synthesizer is beyond the scope of this book (but completely within the scope of other books from Focal Press), programming plug-ins requires know-how and experience and is outside of the scope of this book. But plug-ins are just software versions of the techniques described in the earlier chapters. Analogue synthesizer plug-ins use analogue modeling techniques, either sample replay of samples of analogue synthesizer waveforms or more detailed models of VCOs, VCFs, etc. FM synthesizer plug-ins just express the FM formulas in software, although the more detailed versions also model any peculiarities of the sine wave generators (like limited bit resolution). Wavetable synthesizers are literally counters that access areas of memory holding the wave tables.

Some do-it-yourself plug-in utilities are available, which allow you to make your own plug-ins, either by specifying what the plug-in does in terms of processing MIDI or audio, or by connecting together smaller pre-defined snippets of audio or MIDI functionality. The most advanced versions of this type of tool allow you to create plug-ins that are just like commercial plug-ins.

As discussed in Chapter 5 on modeling, modeling requires detailed knowledge of how the sound is made, as well as what the significant features of the sound are, and can consume considerable amounts of time and effort. But once you know how to model a sound, synthesize a sound or replay a sample, then wrapping that in a plug-in is very straightforward, and the plug-in then works inside any suitable host software.

You do not need to know how to program plug-ins to make the most of them. In fact, unless you want to learn how to program them, learning how to get the best out of plug-ins and make different and interesting sounds can be just as challenging as programming them, and it is far more useful in terms of actually making sounds and music.

An alternative interpretation of 'programming plug-ins' is also important. Plug-ins do not operate in a closed, isolated environment, although some early ones did have very limited user controls. Modern plug-ins provide not only a user interface for the user to change their settings, but these settings can often be assigned to controllers and therefore so can be altered live. Some host software allows the control of plug-in settings to be from other plug-ins, which makes the environment rather like a hardware modular synthesizer, but expanded in capability to a modular audio workstation in software.

Whatever the details of the implementation of the host software and the specific plug-ins that you use, the same principles apply as with any synthesizer or sampler. They are:

- Understand what the device does: a good mental model is ideal.
- Explore the limits of what the device can do: go beyond the manual.
- Understand how the device works in its environment: so that you can exploit it effectively.
- Use and misuse the device and the environment: if you are told that you cannot do something, find a way to do it.
- Do not just collect: compare, contrast and choose the one that suits you, and then learn about it in depth.
- Do not bloat: if you try a plug-in and do not like it, remove it.

6.6 Ongoing integration of the audio cycle

The 'Produce, Mix, Record, Reproduce' sound cycle was introduced in Section 1.7. Now that all of the technology development is reaching the end, the cycle can be presented in the context of how computers are used to make sounds. The following list starts with the oldest technology and least computer integration and ends with the most recent technology and greatest amount of computer integration.

Live performance or jamming is the starting point for many music productions, with the mix as a result of musician interaction. The record of this could be a recording, or it could be sheet music capturing the performance. Reproduction could be through the recording, published sheet music or live performance.

Multi-track recording allows either one musician to control everything or a number of musicians to clone themselves for the purpose of performance. Tracks can be produced singly or several at a time, and they are then mixed down to a final version, which can be reproduced and distributed. At this point, computers are a limited replacement for a real tape-based multi-track recorder, because they have limited ability to record several tracks at once, and storage of removable hard disks or data tape cartridges is awkward.

MIDI sequencers provide computer-based record and playback facilities for musical events on electronic musical instruments. To integrate this with acoustic instruments or vocals requires the capture of the MIDI sequenced audio by a multi-track tape recorder, or the playback of the MIDI sequence at the same time as the playback of the multi-track audio during mix-down. The computer is thus very much a peripheral device that is solely producing the electronic instrument audio part of the final sound.

Sample playback by the computer allows pre-prepared samples of acoustic instruments or vocals to be incorporated into the MIDI playback. This simplifies some of the integration of the computer-controlled audio with the multi-track, but requires preparation of samples. MIDI's ability to deal with samples using MIDI sample dumps (MSDS) was acceptable when samplers were 8 bits and samples were small, but long 16-bit samples and MIDI's slow data transfer speed made using editors slow and cumbersome. Hybrid mixtures of MIDI and SCSI were tried, but they were not as convenient as a MIDI cable.

When the computer can record multiple tracks of audio and then replay them as samples alongside the audio, the computer is moving from a peripheral towards the core part of the recording process. There is no need here for a multi-track recorder, but all of the audio outputs from the electronic instruments and the samples need to be mixed down manually during mix-down. Large MIDI rigs eat up audio mixers, particularly when MIDI synthesizers and samplers have stereo audio out, plus separate individual outputs from internal busses intended as effects sends. They also require multi-port MIDI interfaces to drive all the MIDI devices. The author found that a 16 × 16 MIDI interface and routing matrix was not adequate and resorted to using manual switch boxes and putting simple devices at the end of a Thru box.

Since the computer can also automate the mix of the audio samples, the external mixer is required only to mix the computer output and the electronic instruments. This explains why some manufacturers in the 1980s and 1990s made very simple line level mixers with channels often arranged in pairs with no pan controls, no EQ and lots of effects sends – the author has over 40 channels of exactly this type of sub-mixing.

Adding plug-ins to replace the external MIDI instruments and any outboard effects means that there is no need for any hardware other than the computer and perhaps an audio interface. The computer is acting as the multi-track audio and MIDI recorder, plus the mix-down automation. The plug-in instruments are providing the MIDI electronic instrumentation; the sampled audio tracks are providing the acoustic instruments and the vocals. Trying to work with this type of integrated computer-based system with external hardware requires careful setting of latency compensation inside the software, but it also emphasizes how convenient it is to have everything available in a small computer. No sysex dumps to load sounds, less sample loading from CD-ROMs, no bundles of audio and MIDI cabling connecting everything together and getting tangled and impossible to alter without undoing everything and starting all over again.

In other words, once you have moved onto a computer, analogue becomes difficult and doing everything on a computer becomes easy. Therefore the natural consequence is that audio hardware (effects, synths, samplers) is replaced by software.

Figure 6.6.1 shows a comparison of hardware versus software. The hardware has three MIDI cables, six audio cables (12 for stereo) and requires wiring

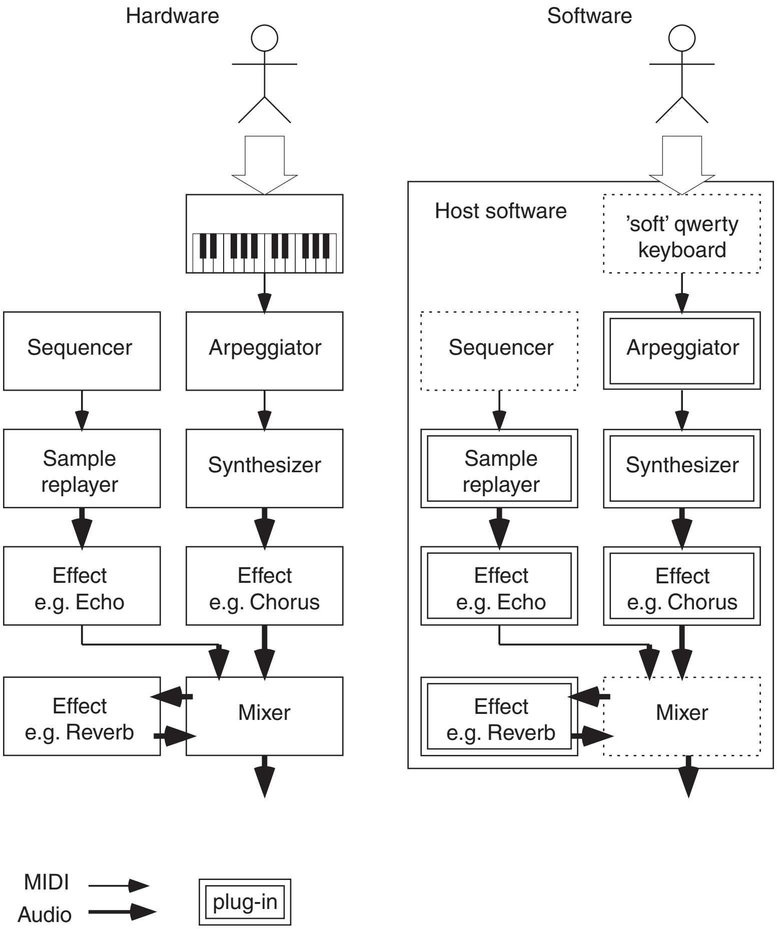


FIGURE 6.6.1 Hardware and software environments.

up and the parameters for the various devices setting (sounds, effects, mix) before it can be used. The software has six plug-ins, and the host software deals with all of the routing of audio and MIDI signals between them. The whole configuration can be stored in the host software and recalled very quickly.

6.6.1 Latency

Latency is the word for the time delay between initiating an action and the action actually happening.

In a keyboard synthesizer, it would be the time that it takes from the key being pressed on the keyboard to the sound appearing at the audio output, and the delay would be caused by the time it takes to determine that a key has been pressed, the time to pass the 'key number n pressed' information to the oscillators and then the 'key pressed' information to start the envelope.

In a MIDI expander or sound module, the MIDI messages are made from blocks of bits arranged in series, and it takes a finite time for the messages to be received, converted into a digital number (you have to wait for the whole message to arrive before you can convert it) and interpreted. Therefore there is a built-in minimum time delay just to receive the message. Any subsequent processing of a MIDI message, like telling a synthesizer chip to start producing an output, also takes time. Sound modules are also frequently designed with 'adequate' processing power, which can slow down the response time, particularly when lots of other messages are being processed (after-touch, controllers).

Effects units need to convert from audio signals into digital, which takes time, then process the digital numbers, which takes more time, and then convert the output digital numbers into audio again, which takes more time.

In a computer sequencer, the major cause of delay is not MIDI, since the messages to a plug-in to initiate a note are already inside the computer and therefore have a short fixed time delay that can be compensated for. Similarly, it is not audio conversions from analogue to digital and vice versa, since the audio signals to effects plug-ins are also already in digital form, and therefore again have a short fixed time delay that can be compensated for. The main cause of delay is with the audio output. The host software communicates to the audio output through a software driver that provides a standardized programming interface to software on the computer, whilst also providing a specialized interface to the on-board audio sockets, PCI-bus sound card or external USB/Firewire audio interface. The driver is thus a critical component, but it behaves in a very predictable way. In order to move the audio from the host software to the audio output, it uses some memory as a buffer for the audio samples. Samples are placed in the buffer by the host software and removed by the driver software as it sends them to the audio output.

When the buffer is large, then the computer processor (the CPU) does not need to move the samples to the audio output very often, and when they are moved, then lots of them can be moved efficiently as one large block. Unfortunately, setting a large buffer size means that the latency is large, which translates to a long time delay between the host software producing the sound and it appearing at the audio output. When the buffer is small, then the CPU will need to move small numbers of samples to the audio output frequently, which is not efficient, and increases the loading. But this has the advantage that the time delay is very low, since samples are sent to the output very quickly.

Therefore, a small buffer size gives low latency but high CPU loading, whilst a large buffer size gives high latency but low CPU loading. Aiming for a zero latency is obviously not a good idea because of the CPU loading, and no matter how hard you push the processor, it is always going to take some time to move samples through buffers and out through the sound card. Setting a compromise value is the answer, and host software documentation will describe a number of techniques that can be used to find the optimum value for your specific application. All computer host software has a CPU load indicator, with either a bar, a percentage or some other metering indication, and this should be something that you monitor whilst working in the host software, in much the same way that one might keep glancing at the speedometer in a car whilst driving along a road with a speed limit. High CPU loads are a bad idea because most host software is set to continue the audio output if it can, whilst sacrificing things like timing accuracy or maybe dropping some sounds.

As a comparison, few hardware synthesizers or samplers provide indication of the processing load, and therefore it is much harder to know if or when you are approaching the limits. When working with external hardware instruments, then the latency inside the host software is likely to be shorter because it has no MIDI input delay, and only an output audio buffer. In contrast, the external instruments have MIDI input delay and an audio input conversion delay as the audio is converted to digital. From that point on, the output audio buffer delay is the same for external or internal audio – effects of instruments. But the MIDI input delay and audio conversion delay can be compensated for by sending the MIDI messages early in time. A similar compensation can be made for external effects units, where the audio can be sent to them early in time, so that it then matches up with the internal host software audio. Some host software has facilities to make this delay compensation process easy and even semi-automatic.

Latency is a natural effect and can always be worked around. It is worth remembering that even when you play an acoustic instrument where you interact directly with the vibrating, sound-making part, then it will still take time for the sound to travel to a listener.

6.6.2 CPU load

Once you are in the habit of monitoring CPU load indicators, and you have thoroughly learnt how to use your host software, and your plug-ins, then it should become very apparent that CPU loading is the major limitation on what can be done with a computer sound-making environment. It provides a practical limit to how many virtual instruments or effects that can be used at any time, and so familiarity with how it is affected by the plug-ins is important. CPU efficiency can be improved by purchasing more RAM, or by replacing a slow 5400 rpm hard disk drive with a 7200 or 10,000 rpm drive, and maybe by replacing USB or Firewire hard disk drives with eSata connected alternatives, but these only provide limited improvements, and once done, there is little more than can be done in terms of local hardware.

Two techniques can be used to maximize your utilization of a computer-based sound-making solution: CPU load optimization and distributed processing.

CPU load optimization requires detailed familiarity with how plug-ins affect CPU loading. You should test the host software to the limits for each plug-in. For effects plug-ins, this could mean adding reverb to lots of separate channels, with long reverb times, until the CPU load reaches its limit. For sample replay, then playing lots of notes simultaneously will stress the polyphony of the sample replay plug-in. Loading very large samples is also a good way of stressing host software, and there are a number of techniques that can be employed to deal with large samples. It is possible to load some samples into RAM, or to optimize the access of files of hard disks by setting buffers in advance with a technique called sample or disk streaming. Virtual instruments that use modeling techniques can present very large processing loads because of the large number of calculations that they are carrying out, and these can be targets for what is known as track freezing, where the track is recorded and therefore becomes a sample, and is then played back using sample-replay facilities, which are normally much less demanding of processor power.

Stereo samples are another area to consider. Panned mono samples can be used instead of stereo samples in many circumstances, and this saves CPU processing power that can then be used for instruments that are more forward and more important in the mix. On a similar theme, reducing the polyphony of accompaniment parts can reduce the CPU load, and often helps to prevent the mix becoming too muddy. With hardware expanders and a MIDI sequencer, it is very easy to double notes and use huge multi-layered stacked sounds, and then to put reverb and chorus on everything, which typically results in a thick muddy sound. Thinking about CPU loading should mean that this will only be done when it is necessary, and not on everything.

Some host software, and some plug-ins, have the facility to stop loading the CPU when they are not needed, or to turn off functionality that is not being used in complex modular plug-ins, or even to be prioritized so that they are muted if CPU loading gets too high. One similar facility is to allow prioritization of timing for tracks that require it, so that drum tracks get priority handling whilst string pads get less tight timing accuracy. This is not a new technology: it was one of the features of some Atari ST sequencing software back in the 1980s. A more modern technique might be to deliberately increase the audio latency to get some more CPU processing power. The most important thing to make the most of the available CPU processing power is to know and understand the host software, and the plug-ins that you use. Familiarity with what the limits are, and the ways of getting around those limits, is what differentiates a professional from an amateur.

Distributed processing takes the opposite approach: rather than to try to maximize what the local CPU is doing, it spreads the load over more CPUs instead. Computer technology is doing this already for local, on-board CPUs by providing multiple processor cores in the same chip, but this chip is still normally fixed inside that local PC. But by connecting to another computer, through either an Ethernet network or a specialized internal or external connection, the other CPUs can be used to share the processing load for lots of virtual instrument plug-ins or demanding effects plug-ins. There are a number of hardware and software solutions to achieve distributed software, and it is a technique that has been used in other industries for many years: the rendering of computer animation being one example.

Using distributed processing requires some pre-planning and access to the processor power, but it is surprising how many computers are not used in some locations, particularly out-of-hours. When the author looked after the computers at a PCB design facility, the network of powerful graphics workstations was used to design circuits by day, but at night, it was used for mathematical problem solving.

One mixture of CPU optimization and distributed processing concerns hard disk drives. Fast 10,000 rpm drives with eSATA connectors can be used as a way of improving a single local computer's ability to pull samples off disk very quickly, but by creating copies of all the samples, then each processor in a distributed system can have rapid access. It may even be possible to create specific hard drive contents with just the sample files required by each of the distributed processors. For large sample sets, RAID (Redundant Architecture of Inexpensive (!) Drives) techniques allow multiple drives to be used simultaneously to get samples very quickly by effectively pulling the samples off in parallel.

This is the sort of technique that is used for video editing, but it can also be used for sample hungry host software, and can be used for sample serving, or for individual processors. On a smaller scale, having a fast external hard drive with just samples on it, and the hard drive in the computer with just the bare operating system and the host software (no browser, no email, no screensavers, and so on, can also not only optimize the CPU availability, but also reduce the chances of virus or malware problems.

Most host software have sample management software built-in or as part of the suite of software, and whilst this may appear to be an unnecessary overhead, it is worthwhile to spend some time optimizing sample sets, their locations and their naming. Logical organization of large numbers of files can help a lot with small projects, but with large projects, and particularly when returning to a large project after a few months, it can become essential.

6.6.3 Changeover

The author finally moved to software between the second and the third editions of this book (which equates to huge latency), and has retained the hardware for sentimental reasons and now has more virtual hardware than ever existed previously in real hardware, and it is all usable! More than that, it is far more flexible, faster to set up and recalls setups better than the hardware. All of which makes it a much more creative process, although it can take just as long, or longer, to explore possibilities.

The switch from hardware to computers has been rapid and leaves behind controllers, live performance, workstations and retro instruments as the remainders of the hardware electronics of the first two generations of synthesizer: analogue and digital. The computer's role has thus moved from being a minor peripheral, to being the major hardware component in the cycle. The synthesist's role has changed from being a technician, cable-plugger and musician to being a conductor, arranger and aspiring computer expert.

6.7 Studios on computers: the integrated sequencer

The computer was described earlier as a general-purpose device, and this makes it adaptable for a number of tasks, which leads to the integration of those tasks. This ongoing integration role has been very strong in the electronic music field, where the computer has moved from a minor peripheral to the host of complete studio functionality. The computer as a complete integration of DAW functionality has been reached in the twenty-first century, and it joins the workstation keyboard as the main ways to produce sound.

Making sounds on a computer allows synthesis and sampling to be controlled in detail, whilst still seeing them in context. Having all the information about a musical performance on a computer allows the view and the detail to be controlled and filtered as required. If a physical synthesizer is being used to make a special effect sound, and the focus is on making the sound, then the relationship of that sound to other parts of the music will not be apparent because of the difficulty of hearing the final sound. But if the synthesizer is a plug-in in a DAW, then the synthesizer can be adjusted live, in context.

Making sounds on a computer also gives freedom from physical constraints. Multi-track recording of a single physical monophonic synthesizer requires accurate timing and tuning, and consumes lots of time. An analogue modeled monosynth plug-in allows copies to be made, and the recording can be made much more quickly and with more opportunities to adjust the timing or performance to achieve the correct feel and emotion in the performance.

Being able to store complex chains of effects, and to name them for instant recall later, can speed up music-making. Having the same ability for synthesizers and samplers can provide even more time-savings or allow much more detailed examinations of alternative possibilities and options. Re-wiring complex sets of physical instruments is not practical in many circumstances, but integrated computer sequencers make it unnecessary.

6.7.1 Evolution

In the 1970s, computers were big, heavy and expensive. In the early 1980s, the home computer made low-cost computing available to all, but with limited power and only a few pieces of music software. In the later, 1980s, the Apple Macintosh was the professional musician's computer of choice, although the Atari ST was a far cheaper alternative that saw huge success from hobbyists to professional use. The first musical applications were simple, but they rapidly developed into sophisticated sequencers that made the most of the large screens and graphical operating systems that differentiated them so strongly from the tiny LED and LCD displays of the hardware sequencers of the time.

As time went by, software sequencers began to converge on a common feature set, with the only differentiators being the details of the user interface, and particularly the metaphors used to represent the music. Sequencers were MIDI only for all but the most powerful computer setups until the early 1990s, when storage and processing began to make audio recording and playback viable. By the end of the 1990s, sequencers were able to work with MIDI and audio with almost equal ease, and some people were moving to working only with audio.

The incremental addition of features to match the developing processing power of the computers meant that the major sequencer packages were large and complicated, and new specialist manufacturers appeared with simpler, dedicated sequencers devoted to sample replay (e.g., ACID) or a complete replacement for a rack of equipment, a mixer, effects and a sequencer (e.g., Reason).

Once audio is integrated into a sequencer, then it can also be used as a virtual sampler. Software samplers based on computer storage and high-performance audio input/output cards have removed some of the dependence on hardware samplers, but computers are not well suited to road usage. Laptop computers are more suited in some ways to live performance (size and battery power) and are increasingly used for, live performances.

Beyond audio, sequencers next acquired mixing functions, so that the samples could be mixed together before being converted into audio for the final mix alongside any MIDI instrumentation and live instruments. Plug-in architectures allow effects to be added into these mixers, and this gradually developed into plug-in modeling instruments to augment the sample replay. Eventually, the whole of the sequencing, instrumentation, effects and mixing could be carried out in software on the computer, with only the final output being converted to audio. Even vocals can be sampled and incorporated as part of the virtual studio on the computer.

The addition of plug-in capability to sequencers allowed the functionality to be extended only where the user required it, but the extra processing power required can be extensive. Magazine reviews have increasingly been quoting figures based on the number of channels of specific plug-ins as a measure of a computer's capability. Sequencers now typically have a meter that displays the processing power being used, and by implication, how much power is left. This is something which hardware rarely, if ever, revealed (Figure 6.7.1).

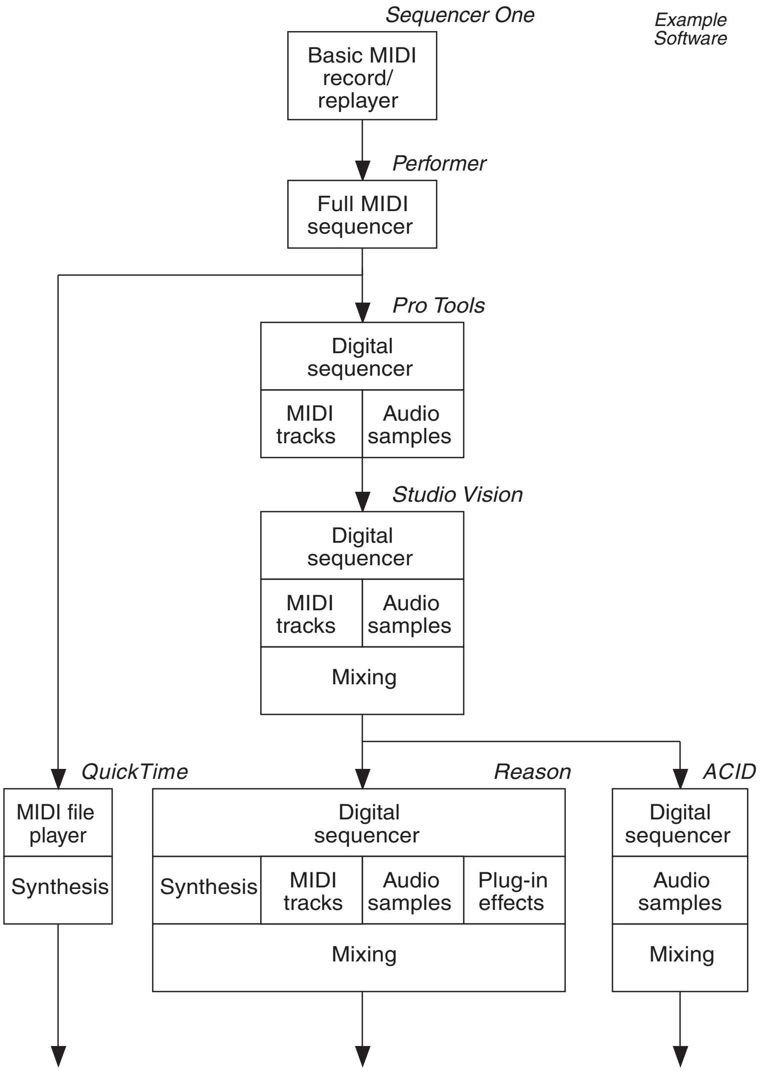


FIGURE 6.7.1 Software evolution.

The brief history of the software sequencer is thus:

- MIDI
- audio + MIDI
- audio + MIDI + mixing
- audio + MIDI + mixing + effects
- audio + MIDI + synthesis + mixing + effects

At some point in this history, the sequencer makes the transition into a DAW, although this term has many definitions and interpretations of what it implies.

One final thought about the way that MIDI has become an accepted part of the way that computers work is to consider using a computer to play a MIDI file. On a Windows PC, the Media Player will use a built-in software GM synthesizer to replay the file, just as if it was an audio file. On an Apple Macintosh, the QuickTime Player does the same replay function through a GM soft synthesizer. For many computer users, standard MIDI files appear to be audio files, because when you 'play' them, they make music. The abstraction that a MIDI file represents is not apparent.

On a wider level, the abstraction that the computer represents has also been accepted: a computer-based DAW can produce a complete musical performance, and yet no one questions how it was done. It has become accepted that computers can be used by composers and musicians to produce complex musical performances.

6.8 The rise of the abstract controller and the fall of MIDI

The first edition of this book dealt with the controllers that appeared on synthesizers. The second edition looked at 2D controllers like the Alesis Kaoss Pad that could be used in applications outside of their intended club usage. DJs are increasingly viewed as performers or conductors, and it has become increasingly difficult to know when a DJ stops and a musician begins. This is actually the same sort of question that arose in the 1970s and 1980s, when the boundaries between record producers and musicians became blurred because of the technological developments of the time, and producers like Rupert Hine or Alan Parsons became well known for innovation and music-making, rather than just organizing its production.H

Live control of parameters has been a strong feature of the end of the 1990s and the twenty-first century. It just so happened to suit analogue synthesizer control for a particular style of music, but the process of adapting and producing new controllers has made it applicable to other types of music, with the result that there are now a wide range of 'abstract' controllers, where a rotary control might be controlling detune, next to a linear slider that is controlling a filter's cut-off frequency, near a cross-fader that is mixing between two arrangements. Abstract controllers are particularly useful when controlling synthesis plug-ins inside a computer, since the arrangement of the plug-ins can change very quickly, and the external controllers need to be able to change to match quickly too.

Most of these abstract controllers work either through MIDI or increasingly by using MIDI hidden inside a USB connection. USB has become a very popular and ubiquitous computers peripheral interface, and there are many devices that use it to connect to a computer. By providing a USB connection to a device that is a MIDI conversion box, it is possible to input MIDI controller messages into the computer. But the same conversion box could also be a DJ controller that has rotary controls, sliders, cross-faders, 2D or 3D gestural controllers and perhaps even a music keyboard, all producing MIDI controller messages, and the box will transmit those MIDI messages over the USB connection to the computer without even providing any MIDI sockets on the outside of the device. The twenty-first century has seen an increasing number of controllers that hide their use of MIDI, and it may be that these are the first examples of the more generic synthesizer, sampler and sequencer controller of the future.

6.9 Dance, clubs and DJs

Dance music is interesting not only because it is music with a very clear purpose, music to dance, but also because it needs to react to its audience. An effective dance set is not all 120 bpm 4:4 beats with driving rhythms and strong melody lines. Truly effective performances have dynamics, emotions and contrast, and follow the audience's moods and swings rather than impose on them. To do this requires the right controllers so that the correct elements are adjusted, and therefore the music becomes a live performance rather than just sequencing from one item in a playlist to another.

Controllers allow the direct use of computers to be hidden or obscured. CD decks are replacing or augmenting LP decks in some installations, and if mixers are joined by a number of abstract controllers, then the sound-making could be a computer, a groove box or a keyboard workstation. It is still not clear where the end point of dance music technology is going to be.

But there are many other forms of music, although there seem to be fewer controllers arising from other performance environments. There are other users of synthesizers, samplers and electronic music technology, and is the computer well suited to their usage? Suppose you do not want a computer? Have we ended up with a musical environment that is tailored to a few specific styles and performance approaches and lost the genericism of old sequencers? What is going to happen over the next decade as more synthesis and sampler hardware fails and computers are the only answer?

6.10 Sequencing

What sequencer? The 1990s saw the rise of the workstation keyboard and the gradual disappearance of the synthesizer and sampler as separate, stand-alone devices. The integration of a sequencer into a keyboard with synthesis, sample replay, effects and maybe even CD-R burning removes some (or most?) of the need for hardware synths, sequencers, effects boxes, MIDI. The stand-alone hardware sequencer has been absorbed into a broader music-making device.

Computers used for music are just a virtualization of workstations, but with bigger screens, no music keyboards, more portability and more plug-ins offering a wide choice of synthesis techniques, sounds and effects. Some workstation manufacturers have recognized this, and have moved to make their workstations much more like computers with music keyboards by adding plug-in compatibility.

Sequencer skills are still required when using workstations or DAWs (host software for plug-ins), but they provide additional complexity because they need to be integrated into the synthesis, sampling, effects and mix-down environment which the sequencer is part of. One example of this is hocketing (see Section 7.4) where conventional descriptions talk about varying sounds according to various parameters like pitch, velocity or time. But once you have an integrated environment for creating music, then you can add additional variations that have little to do with the sounds, but more to do with the sound-making environment. Some examples include:

- changing the pan position with velocity (mixer function)
- changing the reverb with pitch (effects functionality)
- changing the echo feedback with 'beat in the bar' position (effects functionality).

6.11 Recording

Recording onto tape in an analogue environment required careful tuning and planning of performance, balancing volumes of equipment that was being recorded onto a limited number of tracks, perfect playing. Pressing the 'Record' button was a major event. In contrast, the 'Record' button is much less threatening in a world where many tracks are available, editing is easy and everything is in one place, with complex effects, routing, mixing, synthesis stacks and multi-everything samples, and in context-monitoring. This hopefully promotes experimentation, but it may be that it merely increases pressure for instant results.

Recording on computers does have some issues. The major requirement is for lots of computing power, especially when lots of plug-ins are being used, unless you are using lots of legacy analogue equipment to save processing power. Computers and quiet environments do not necessarily go together, which means that the computer loses some of its advantages.

6.12 Performing

Synthesis in performance is a wide topic and is the one where the ongoing developments cut across any attempt to categorize and neaten them. Accordingly, this book has noted, many times, the reuse of devices, the way that critical success can follow commercial failure, and an ongoing evolution towards integrated devices and away from traditional hardware. The new hardware seems to be computers, but these can have a relatively high initial cost, issues with stability of operating systems and frequent software update costs. This makes the true cost of ownership much greater than the initial purchase price, and therefore there still seems to be a role for hardware in some circumstances. It remains to be seen in what form the traditional hardware manufacturers will survive in an increasingly software-led world. The purchase of music software companies by some of the hardware companies (Apple and Emagic, Yamaha and Steinberg.) might indicate the likely outcome.

The integration of many devices into single integrated performance stations is also significant. In the 1980s, apart from the amplifiers and speakers, there would have been little musical equipment in common between a performance by an experimental electro-acoustic band, a pop band on tour and a DJ in a nightclub. The first years of the twenty-first century have seen much more reuse of equipment, and therefore the unsigned experimenter, turntablist and band might all use a groove box for part of the set, plus two turntables, plus a laptop running a sequencer or something more exotic.

6.12.1 Synthesis live

The live reworking of sound has evolved from the playback of simple prerecorded sequences, through 'scratching' LPs, to sophisticated performances. In the process, sound synthesis and sampling have moved from a 'back-room' activity that happened slowly and meticulously to an interactive improvised performance. Both extremes still exist-the 'studio album' may still take years of careful, painstaking, detailed assembly; but when used in live performance, the technology has increasingly been used for much more transitory and immediate material.

6.12.2 The rise and fall of keyboard stacks

When synthesizers were room sized, the synthesist went to the synthesizer. This is analogous to the conductor going to the orchestra: the orchestra requires many resources, takes time to set up and is expensive to run. The orchestra and the synthesizer both require considerable effort to be expended by the conductor and the synthesist in order to realize the music.

This situation only changed when more portable instruments like the Minimoog and Electronic Music Studios (EMS) VCS-3 were produced. These were more suited to live performance and were affordable and accessible to ordinary musicians, in contrast to the small number of pioneers who had been performing live with modular synthesizers up until this point.

Perceptions and expectations change with time and context. In the late 1960s, audiences were familiar with seeing keyboard players playing a single instrument that made just one sound, either a piano or an organ. Keyboards were either solo instruments (accompaniment to a soloist) or in a band context, they were typically used as backing instruments, providing chordal and rhythmic backing. The 1970s saw the introduction of additional keyboards and a gradual change to 'stacks' of keyboards, where one would be arranged above another. Unfortunately, the physical design of most synthesizers or keyboards is not well suited to being placed on top of another, and over the next 10 years, rapid development of keyboard 'stands' provided ample scope for the raising and angling of keyboards to suit the demands of performers.

Although they were both released in 1969, the design approaches of the Minimoog and the EMS VCS-3 were very different. The Minimoog was more performance oriented and therefore was used for melodies and bass lines, whilst the VCS-3's patch-panel flexibility was more suited to sound effects – especially with the patch memory plug-in cartridges. (Note how a 'plug-in' cartridge from the 1970s sounds retro in that context, but up to date in 2000s computer-speak.)

The first 'stacks' were more modest: often just a Minimoog on top of a Fender Rhodes or Wurlitzer piano, and in fact, the rounded top of the Fender Rhodes was eventually replaced with a flatter, more stack-friendly version. By the end of the 1970s, the well-equipped performer might typically have had a 'keyboard stack' containing a dedicated string section emulation called a string machine, a string synthesizer or a string synth, an organ, an electric piano and a monophonic synthesizer (frequently a Minimoog). Serious professionals would also have a portable electric replacement for a real piano: Yamaha's CP70 'electric' piano was a ruggedized piano where the contact-microphone-equipped harp section could be removed from the keys, action and hammers, and therefore was easier to transport.

Some monosynths may have been intended to be used in conjunction with other keyboards, but the physical design did not match this.

Even with a flat top keyboard underneath, the front-to-back depth of a Minimoog or an ARP Odyssey made it almost impossible to place in a good and stable playing position.

The early 1980s saw a gradual replacement of the string machine and the organ with polyphonic synthesizers, but the biggest two changes came with the Yamaha DX7, and also became the first instrument that most people encountered which had MIDI as standard. The Sequential Prophet 600 may have had MIDI, but it was just another polyphonic synthesizer, whilst the DX7 changed the keyboard stack forever.

Performers such as Rick Wakeman and Herbie Hancock were probably almost as famous in the 1970s for their use of multiple stacked keyboards on stage, as they were for their music.

The DX7 had one preset sound that triggered a sudden change in the established stack, as well as the electric piano industry: preset 11, called 'E. Piano 1'. Although not a perfect emulation of an electric piano, this single preset changed the keyboard stack and introduced a whole new category of musicians to electronic musical instruments. Musicians who had been using big and heavy electric pianos replaced them with the light and slim DX7, and hence hastened the rapid development of a variety of keyboard stands. But the many players of home organs finally had access to a single keyboard that could sit on top of an electronic organ and supply polyphonic versions of a wide variety of additional instruments with a much higher fidelity than many of the organs at the time. In particular, the DX7 excelled at sounds that were both reasonably realistic, and difficult to achieve with analogue instruments: vibes, glockenspiels, harpsichords and bells. It was also very affordable and offered 16-note polyphony, which was more than generous at the time in comparison with the more usual 4-, 6- and 8-note polyphony of the analogue polysynth alternatives. As a consequence, the DX7 sold in large numbers to both markets, and became a runaway best seller for Yamaha. But the DX7 also created a solid user base for MIDI, and changed the synthesizer from an expensive, specialist rarity into an affordable, general-purpose, mass-market workhorse.

As MIDI became established, there were other casualties in the late 1980s. String machines and organs were gradually replaced by polyphonic synthesizers, and even keyboards themselves disappeared to produce sound modules designed to be driven exclusively through MIDI from a 'master' controller keyboard. The 'live performance' keyboard stack quickly became just one or two master keyboards connected to a 19-inch rack of sound generating modules connected to a mixer. In the studio the playability of physical keyboards meant that the ratio of keyboards to modules was higher, but the 'one keyboard:one sound' era was gone forever. The word 'stack' does live on, it has a become common usage to refer to a number of sounds all triggered from one keyboard as a 'stack' (Figure 6.12.1).

In the 1990s, the keyboard stack and the keyboard player moved away from the limelight on stage as guitars, vocalists and the ability to dance became fashion essentials. Keyboards and synthesizers moved backwards into the darker part of the stage, as vocalists and dancers became the visible 'stage presence'. Increasingly powerful synthesizer modules meant that the 19-inch rack required less and less space for individual modules. By the end of the 1990s, the laptop computer had become a viable alternative for some types of music, and it joined the DJ's pair of decks as a familiar live performance setup.

In the first years of the twenty-first century, the trend away from keyboards and hardware increased, with phrase sequencers consolidating their position as a reworking of the hardware sequencers of old, but this time with drums, bass, accompaniment and even vocal backing through samplers all on board, and with ever more powerful laptop computers as an alternative.

On stage, the keyboard has moved in and out of the limelight many times over the past 50 years or so. The vocalist has always been the center of attention, except for the guitar solo, where the lead guitarist gets the spotlight. Bass guitar and drums may get solos, but the rhythm section is almost always at the back of the stage. The 1970s and 1980s saw some performers and bands that used just keyboards and drums: Rick Wakeman, Tangerine Dream, The Human League, Ultravox.

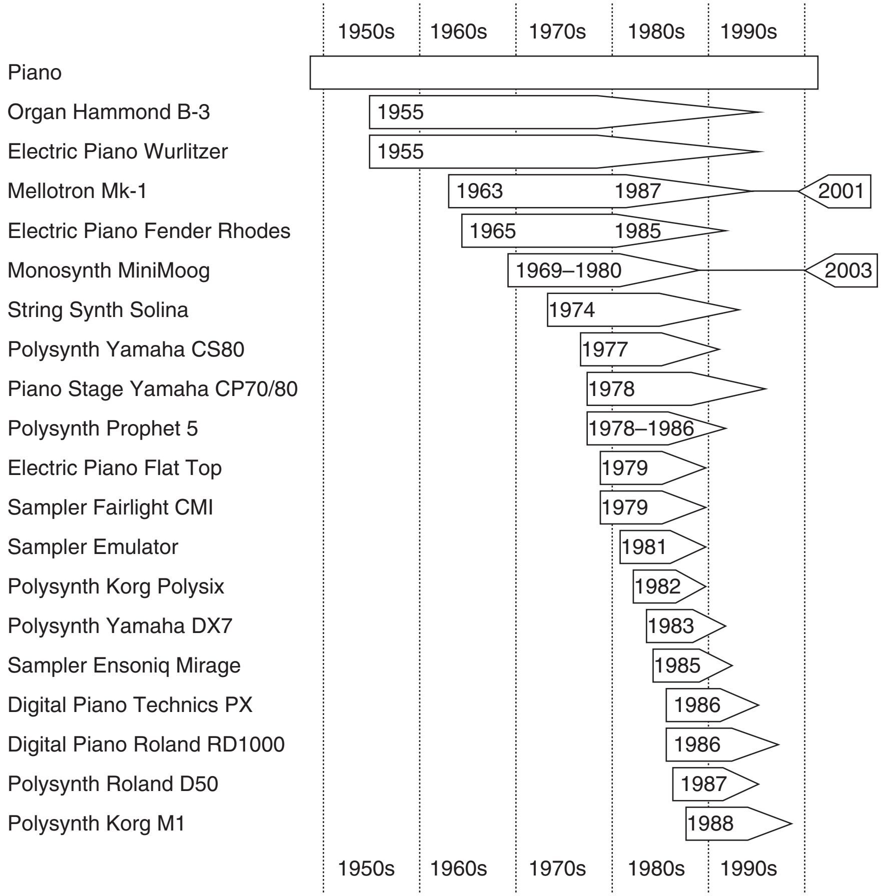


FIGURE 6.12.1 Stack evolution.

By the mid-2000s, hardware was being sold off as software synthesis and modeling became solidly established and the laptop computer with a MIDI over USB controller keyboard or just a MIDI over USB controller became dominant. Instead of a roomfull of hardware, people now had a disk drive full of modeling software instead. In less than 10 years, the market had completely changed, from MIDI and hardware to USB and software.

The timeline for Chapter 2 covers live, physical performance.

6.13 Examples

6.13.1 The Music System (1984)

The Music System was music-making software for the BBC B microcomputer that provided a simple step sequencer and a simple waveform oscillator with envelope sound generation. It provided state-of-the-art 8-bit computer sound generation in 1987 – a proof that 8-bit microprocessors could be used to make music.

6.13.2 Ample (1987)

Ample was the music-oriented programming language used by the Hybrid Music System for the BBC B and not connected to The Music System at all. The main part of the system was the Music 5000 (or earlier 500) unit, which contained a 16-channel wavetable synthesizer with stereo outputs and was programmed using Ample, an advanced music programming language that provided control over music events and musical environment events. Other units included the Music 2000 MIDI interface.

6.13.3 Max (1986)

Max was a music control software program written in the C language and developed at IRCAM by Miller Puckette. It was programmed by using the mouse to connect rectangular functional blocks together and could be used to make MIDI message processors, arpeggiators, step sequencers and lots more. Max and the audio processing and video processing variants continue to be used today in various forms, and have changed the way that programming is presented to programmers and non-programmers.

6.13.4 Studio Vision (1989)

Opcode was one of the first companies to produce a MIDI sequencer for the Macintosh and an early casualty of changes in the music business in the late 1990s. Studio Vision was their flagship sequencer and was the first commercially available music sequencer for a PC (the Apple Macintosh) that integrated MIDI and digital audio recording, editing and playback. Innovations that it introduced included sequence queues controlled by the qwerty keyboard (type 'abacab'), monophonic audio-to-MIDI conversion (in 1998) and tight integration with MIDI interfaces like the Studio 5LX, which allowed sophisticated MIDI processing. It was popular amongst professional musicians and widely used in the music business.

6.13.5 Sonic Foundry ACID (1998)

ACID is a sequencer that plays samples. It comes complete with sample library and has seen continuous development since its release: rather like a hardware sequencer plus hardware sampler, but without the hardware. Subsequently bought by Sony from Sonic Foundry, and currently at version Acid Pro 6, ACID was the first sign that the hardware sampler's days were numbered, and one of the first sample-replay sequencers to become popular at a time when MIDI sequencers were dominant.

6.13.6 Reason (2001)

Propellerhead Software's Reason took the idea of a DAW being a rack of synthesizers, samplers, effects and mixer and made it look like one. Complete with a rear view that has all the patch cables, Reason looks like an outboard rack, and the design has a sort of 1970s retro feel to it, even for equipment like multi-sample replay that you could not get back then. Reason is a great tool for immersing yourself in and experimenting.

6.13.7 Ableton Live (2001)

Live is exactly what the name suggests – a DAW that you do not need to stop. You can do almost everything without stopping it – and just about the only time you get a dialog box is when you get a warning that the audio will stop. Combining step sequencing with phrase sequencing with time and pitch flexible multi-samples and chainable effects, Live is a synthesizer that has swallowed a sequencer, and vice versa. Amazingly capable and flexible, Live looks stark and clean but this hides depth and sophistication. It could be called: 'the Studio Vision of the twenty-first century'.

6.13.8 Vocaloid

Yamaha's Vocaloid singing software continues to develop, and can hopefully blossom in the long term, which is where the true potential of this technology will be realized. In the last edition of this book, a descendant of Vocaloid was predicted to have a number one hit in the future, and there seems to be no reason to revise this, although the date will probably be wrong, and the likelihood of it being on a television talent show remains low.

6.13.9 Reaktor

Often compared to a modular synthesizer, Reaktor from Native Instruments is much more like an audio synthesis and processing toolkit. It provides vast resources for making and processing sounds, as well as for making further things that make and process sounds. If Reaktor has a weakness: it is that is has such depth and capability that it can occupy all your attention and effort.

6.14 Questions

1. What might have happened if PCs had been called 'work-group' computers instead?
2. What can MIDI do that analogue finds much more difficult?
3. Is Moore's Law a 'Law'? And what does it mean?
4. Name two pieces of host software.
5. Can a plug-in be used with any computer operating system?
6. What happens next in DAWs on computers? Has the end point been reached?
7. What would you take on stage for a live performance: a workstation keyboard or a computer running DAW software?
8. Analogue synthesizers are still used. Digital synthesizers are still used. Given the rapid obsolescence of computer hardware and software, will today's software be usable in 10 years time? In 20 years? Will it be possible to migrate proprietary files that describe complex assemblies of sequencer and plug-in functionality to other software? Is a time-bomb already ticking?
9. Answer the rhetorical questions in Section 6.9.
10. If synthesis and sampling have become totally integrated into DAWs, will this hinder their further development, or open up possibilities?

6.15 Timeline

Date	Name	Event	Notes
1933	IBM	First commercial electric typewriter.	
1936	Alan Turing	Described a Turing Machine, a general-purpose computing machine.	Derived from Kurt Gödel's work on the limits of mathematical proofs.
1949	Manchester University	The Manchester Mark 1, the first true stored-program computer.	It used a variation of a TV tube for storage.
1950	Alan Turing	Proposes the 'Turing Test' for artificial intelligence.	A bit like someone listening in to Instant Messaging and trying to judge which of the senders is intelligent.
1955	Transistor	Nobel prize for the transistor invention.	Shared by John Bardeen, Walter Brattain, and William Shockley.
1958	Chip	The Integrated Circuit (IC) is the basis of all chips since.	Jack Kilby and Robert Noyce.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1963	Mouse	The first computer mouse is invented.	X–Y positional device.
1969	Arpanet	Arpanet links two computers at Stanford.	The Internet started here.
1971	Floppy Disk	Called ‘floppy’ because early versions were. Later versions added harder casings.	First ones were 8 inches or more across!
1971	Hiller and Ruiz	Published ‘Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects’.	Used mathematical approximations to solve the wave equations for physical modeling.
1972	C	Dennis Ritchie writes C, an intermediate level programming language that became very popular and influential.	C + + the power user language is a revised and an improved version of C.
1976	Apple	Apple II is launched, first home computer with color graphics.	Prototype is in a wooden box, now at the Smithsonian in Washington.
1978	IBM	The personal computer is announced.	The start of the modern era of desktop computers.
1978	VisiCalc	The first spreadsheet program.	Before this, people used pen and paper.
1979	WordStar	One of the first word processors.	Before this, people used typewriters.
1984	Macintosh	Apple launched the Macintosh, with a leading edge graphical user interface.	Influenced by Xerox Parc’s Star demonstrator, but turned into a consumer device.
1987	Julius O Smith	Published ‘Music Applications of Digital Waveguides’.	One of the early practical descriptions of ‘waveguide’ physical modeling synthesis.
1987	Karplus and Strong	Published ‘Digital Synthesis of Plucked String and Drum Timbres’.	The roots of waveguide physical modeling.
1989	Tim Berners-Lee	Invents the WWW (World Wide Web) as a way to distribute academic papers and documentation.	The WWW takes off rather faster than anyone expected!
1990	Windows	Windows 3 announced.	GUI for everyone.
1991	CD-R	CD-Recordable is launched.	CD would never be the same again, ever.
1995	Java	Sun releases the Java programming language.	Java is now used in many web software projects.
1996	Google	Just another university search engine project.	Google has grown just a bit since then.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1996	Steinberg	VST, Virtual Studio Technology, allows audio plug-ins in sequencers.	Followed by VST2, which allowed MIDI processing and thus allows synthesizer plug-ins.
1997	Deep Blue	Big Blue, the computer, beats Kasparov, then the reigning chess world champion.	Big Blue was a highly optimized IBM computer.
1997	DVD	The DVD format is launched.	DVD's copy protection held up the release of DVD for longer than it took to be cracked.
1998	Google	The search engine goes public.	7 September, 1998.
1998	Nemesys	GigaSampler is one of the first software samplers for PCs. Initially aimed at professional users.	Turned a PC into a MIDI-triggered sampler.
1998	Sonic Foundry	ACID, loop-based sample sequencing software.	A low-cost alternative to hardware samplers…
2000	Traktor	Traktor, a software DJ solution is developed.	Later licensed to Native Instruments.
2001	Apple	Mac OSX 10.0.0 Cheetah is launched.	Mac moves to Unix.
2001	Apple	iPod is launched.	Not a runaway success at first: a slow start.
2002	Gartner	1 billion PCs sold since the 1970s, according to research film Gartner.	
2003	Native Instruments	Traktor DJ Studio 2.5 DJ software is launched.	Adds time-stretching, OSC (Open Sound Control) support and skins.
2003	VirSyn	Cube, a software additive synthesizer.	Uses sound morphing to simplify the user interface.
2004	VirSyn	Cube 1.5 added analysis to the Cube additive synth to allow for resynthesis of sounds.	A neat extension to the morphing technology used to simplify the user interface.
2004	VirSyn	Cantor, a software vocal synthesizer.	Provides access to the phoneme transitions: one of the keys to vocal realism.
2004	Native Instruments	Guitar Rig, guitar audio path modeling software.	Models effects, amplifiers, speaker cabinets and even microphones.
2004	Korg	Legacy Collection, PolySix, MS20, and Wavestation plug-ins.	Original release had an MS20 hardware controller. Later release had just software.
2005	Muse	Receptor, dedicated VST hardware box to run VST plug-ins.	Also runs as a 'zero-cpu load' virtual instrument expander.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
2005	YouTube	Video sharing site starts up.	Runaway success.
2006	Korg	MicroX, a HyperIntegrated (HI) S&S sound source designed to be used with a plug-in.	Provides a zero-cpu load sound source (or virtual instrument expander). An experiment to see if the expander can be re-invented for a plug-in world?
2006	Sony	Blu-Ray is announced at CES Show.	HD-DVD loses battle against Blu-Ray in 2008.
2007	Apple	iPhone announced.	A rather different look at what a mobile phone should be – a computer!
2008	Digidesign	Transfuser, a loop-based groove utility for ProTools.	Launched as a preview first, then the full software later.
2008	Steinberg	VST 3 improves parameter automation and timing.	Adds deeper integration with the host software.
2008	VirSyn	Matrix, a software vocoder.	One of many late 2000s vocoder examples.

PART 3

Applications

This page intentionally left blank

CHAPTER 7

Sound-Making Techniques

CONTENTS

Techniques

Environment

The chapters so far have concentrated on the theory behind synthesis and sampling. This chapter deals with the use of synthesis and sampling to make music and other sounds, whilst Chapter 8 looks at how synthesizers and samplers are controlled.

7.1 Arranging

At the simplest level, the user interacts with the synthesizer to produce sounds. This often means directly controlling the synthesizer by using a keyboard or other controller. The synthesist is thus carrying out a role which is analogous to that of a player in an orchestra.

But when the control is indirect, as in remotely controlling a synthesizer by using musical instrument digital interface (MIDI) or other computer-based means, the role is much closer to the conductor of an orchestra but with the option of simultaneously being able to act as an individual performer as well. This dual role actually provides the synthesist with something much closer to the detailed control and freedom that is available to an arranger, rather than the decreasing degrees of freedom that are available to the conductor and the performer.

Synthesis thus provides the ability to control three layers of a performance:

1. Choice and control over the use of timbres (arranger).
2. Precise control of the timing and dynamics of individual sounds in context, as they are produced (conductor).
3. Very low-level detailed control over the production of timbre (player).

For an arranger, working with a conductor and an orchestra requires a transfer of instruction through the score, demonstration or the spoken word.

To simplify the text, the words 'synthesizer' and 'synthesist' have been used, but these should be taken as referring to both synthesizer and sampler, and the users of them.

Also, this chapter uses the word 'keyboard' as a way of referring to the generation of note events, but these could also come from any other controller such as a wind controller or a guitar controller. This avoids the continuous use of phrases such as 'a note produced by a keyboard key press, wind controller articulation or guitar controller string pluck'.

Note that whilst MIDI control is described in this chapter, the techniques also apply to control voltage (CV) and gates, or other interfacing methods found in analogue synthesizers.

But for a synthesist, changes to the performance or the conducting require only a change of role. (This can make the task of a synthesist working as part of a larger orchestra a difficult set of compromises regarding control over the synthesizer.) Many multi-timbral synthesizers provide this 'three-layered' level of control within one instrument, since they can produce several different independent timbres simultaneously. The skills of arranging are thus applicable from single synthesizers up to large 'synthesizer orchestras', or even a combination of real and synthetic instrumentation. In this section, the emphasis will often be focused onto individual sounds, but the same principles equally apply to larger structures.

The sales literature for many synthesizers concentrates on the production of sounds, and not the wider aspects of arranging. This chapter is not intended to be a guide to arranging, since that is a complete topic of study, but instead to show some of the tools and techniques that are available to the synthesist. There are three major divisions:

1. *Arranging*: This covers topics such as stacking, layering and hocketing.
2. *Timbres*: This covers topics such as multi-timbrality and polyphony, general MIDI (GM) and the use of effects.
3. *Control*: This covers topics such as the use of performance controllers and editing.

Many of the terms that are frequently used in connection with synthesizers are only loosely defined or even redefined for this specialist usage. Stacking and layering are often confused and given different or overlapping definitions, and they are often used interchangeably in manufacturer literature. In this book, stacking will refer to a single composite sound produced from two or more timbres, whilst layering implies a composite sound that changes or evolves with time, and thus exhibits dynamic changes of separate 'layers' of sounds (Figure 7.1.1).

There are two different methods that can be used to produce stacking or layering: several separate synthesizers can be used, or alternatively, a single multi-timbral synthesizer can be used. These two ways of achieving stacking or layering are rather more than just differences in scale or equipment:

1. Physically wiring synthesizers together with MIDI cables requires a MIDI patch bay and specific synthesizers, which can make it difficult to store or move from one location to another unless the same equipment is available in both places. This is thus a large-scale structure, typically within the MIDI network, and stored in many different locations: each of the individual sounds within the synthesizers, plus the MIDI setup held in the patch bay or a sequencer setup.

2. Multi-timbrality is implemented as part of the operating system functions of a synthesizer and so is achieved using software. Stacks or layers produced by utilizing multi-timbral features can thus be stored

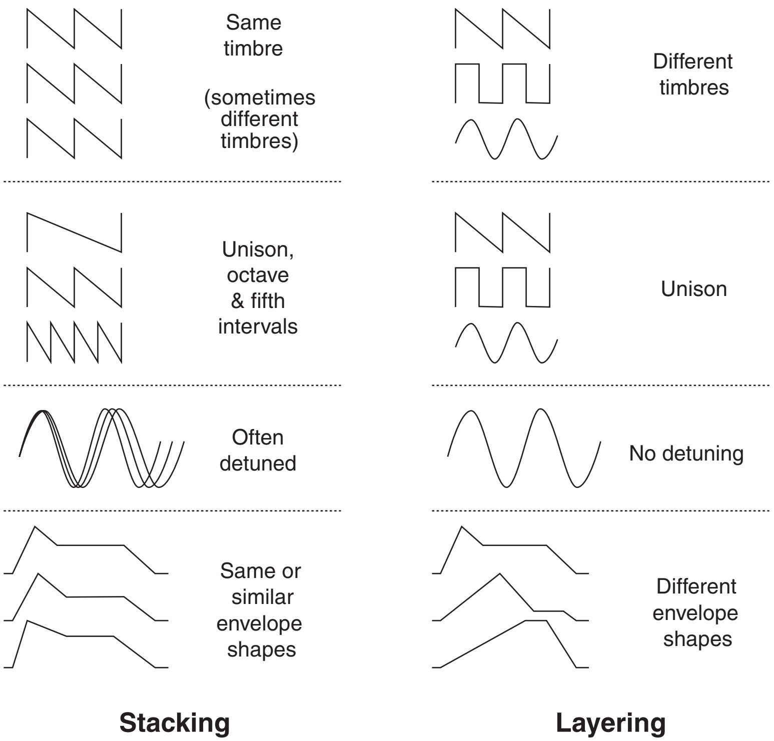

Same timbre
(sometimes different timbres)
Different timbres
Unison, octave & fifth intervals
Unison
Often detuned
No detuning
Same or similar envelope shapes
Different envelope shapes
Stacking
Layering

FIGURE 7.1.1
Stack versus layer. Stacking is a single composite sound produced from two or more timbres. Layering is a composite sound which changes or evolves with time.

as part of a performance memory inside the synthesizer, which makes transportation simple: everything required for the sound is contained within one instrument.

In this chapter, this distinction between the sources of sounds will be largely ignored so that the details of the techniques are not obscured. The phrase 'synthesizer sound source' or the word 'part' will be used whenever a sound can be produced by a separate synthesizer or a section of a multi-timbral synthesizer. But the practicalities of employing the methods described earlier should still be considered when they are being used in practice.

7.2 Stacking

Stacks can be divided into two types: composites and doubles. Composites are concerned with multiple sounds or timbres, whilst doubles take one sound and use multiple pitches derived from it.

7.2.1 Composites

Composites are combinations of more than one sound happening at once. Using more than two parts can waste the available polyphony, and can also produce sounds which are over-complicated. The best sounds are often simple but distinctive, which is harder to achieve than you might think. There are several methods of producing composite sounds.

Additive

Simply combining two or more sounds at random rarely produces useful results. By choosing simple sounds, it is possible to use the same abstraction as additive synthesis to produce composite sounds that are made up from much simpler sounds. This technique is particularly useful when the synthesis techniques used are limited in their timbral possibilities: for example, subtractive synthesis often provides low-pass filtering only, but by adding together two sounds produced by subtractive synthesis, it is possible to produce more complex timbres which do not have the limitation of a single low-pass filter.

Hybrid

Hybrid sounds are produced using contrasting or complementary synthesis techniques. Although this often implies the use of physically different synthesizers, some multi-timbral instruments do allow different synthesis techniques to be employed for each part. The range of 'contrasting and complementary' techniques is large, but some possibilities include the following:

- Analogue with digital, where the 'natural' sound, variations in timbre and slight tuning often attributed to analogue synthesis can be used to complement the more precise and controlled 'digital' sound.
- Imitative with synthetic, where the resulting sound has some of the characteristics of the real instrument, but with enough artificiality to ensure that it is not mistaken for a purely imitative sound.
- Familiar with alien, where the final sound has some elements that are familiar to the listener, but which also includes additional elements which are unfamiliar. This can be useful for avoiding the overuse of a lush string sound as a generic pad or backing timbre. Examples include mixing violin samples with slightly pitch-enveloped sawtooth waveforms, or adding munchkinized vocal sounds to conventional choral sounds.
- Additive with frequency modulation (FM), where FM sounds are used to counter the inharmonic weakness of the additive synthesis technique.
- Sample with imitative, where the basic sample is enhanced with additional imitative sounds to make it sound more like 'the real thing'. Curiously, many people prefer sounds which are made hyper-real in this way rather than exact copies.

Splitting

The process of splitting a sound into component sounds, then producing those sounds using separate sound sources and finally combining them together to produce the final sound is related to the analysis–synthesis methods of digital

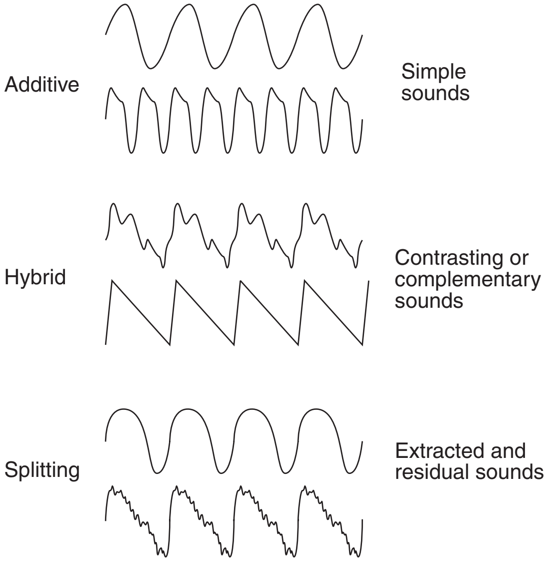

Additive
Simple sounds
Hybrid
Contrasting or complementary sounds
Splitting
Extracted and residual sounds

FIGURE 7.2.1
Composite stacks.

synthesis. But in terms of stacking, the usual method is to choose sounds that approximate to the required components and then iteratively combine them, with any analysis being done intuitively by the synthesist. Techniques such as residual synthesis, where a sound with a similar spectrum is 'removed' from a source sample, and where the residual source sample is then used as the basis for further iterative extraction of sounds, do exist and may be used in future synthesizers (Figure 7.2.1).

7.2.2 Doubling

'Doubling' is a hi-tech musical term, which is used to describe the reuse of the same musical information. Doubling implies either a transposition or a tuning change, with both the original and the doubled parts then playing together. Doubling using transpositions thus produces fixed parallel intervals.

The word 'double' is used in classical music to mean a variation.

Detuning

Detuning the two parts produces a 'richer' sound because of the chorusing providing additional 'movement' and interest in the timbre. There are two approaches for detuning parts: detune both away from 'in-tune' by opposite amounts and detune just one. These two methods have different results when used in combination with other sounds, and the optimum is best chosen by experiment.

Octaving

Transposing one of two parts up or down one or more octaves can be used to 'thicken' up a sound, although as the interval increases the parts will tend to be heard as two separate sounds. Octaving can also change the harmonization of the music.

Intervals

Transposing a part up or down by an interval other than an octave produces parallel pairs of notes. Fifths are commonly used, although this can change the harmonization of the music.

Chording

Transposing two parts away from a third part produces parallel chords. Although useful as a special effect, the constant chording can completely upset the harmonization of the original music.

Alternatives

Although not strictly doubling, it is possible to stack two very similar sounds from different synthesizer sound sources. This is used in the same manner as audio 'double-tracking', where two different performances of the same material are combined so that the result incorporates the slight imperfections and differences from each and so produces a more interesting and sonically 'rich' sound. For example, this variation on doubling can be used to combine the string sounds from two GM modules so that the sound is 'thicker' and not as readily identifiable as coming from a standard GM source (Figure 7.2.2).

Guitarists who use hex pickups as the interface to guitar synthesizers have additional stacking capabilities. The guitar sound from the conventional pickup can be combined with the synthesizer sound and the sound from the hex pickup, and all three of these sound sources can be independently processed and placed at different locations in a surround sound-field. This can provide a versatile way of creating sounds that can exploit the depth of control possible from the close interaction of the guitarist's fingers with the strings and the fretboard.

Keyboard synthesizers tend to send the same CV and gate or MIDI messages to one or more expanders or samplers to produce the sounds, and so the stacking is very precise. Additional controllers can be used to provide variations to the stacked sounds, and this is particularly easy in computer-based host software.

7.3 Layering

The principle of layering has already been discussed in the context of the two parts of many samples and synthesis (S&S) sounds, but the same principle can also be extended to using more than one complete sound – individual parts can be layered, and then these composite sounds themselves combined in layers. This is particularly useful when equipment that is full of preset sounds needs to be made more personal. By layering customized sounds with the presets, new textures can be created with the minimum effort by making hybrid sounds. For this type of use, conventional sounds are not as useful as the more unusual ones that might be rejected as being unusable in normal circumstances (Figure 7.3.1).

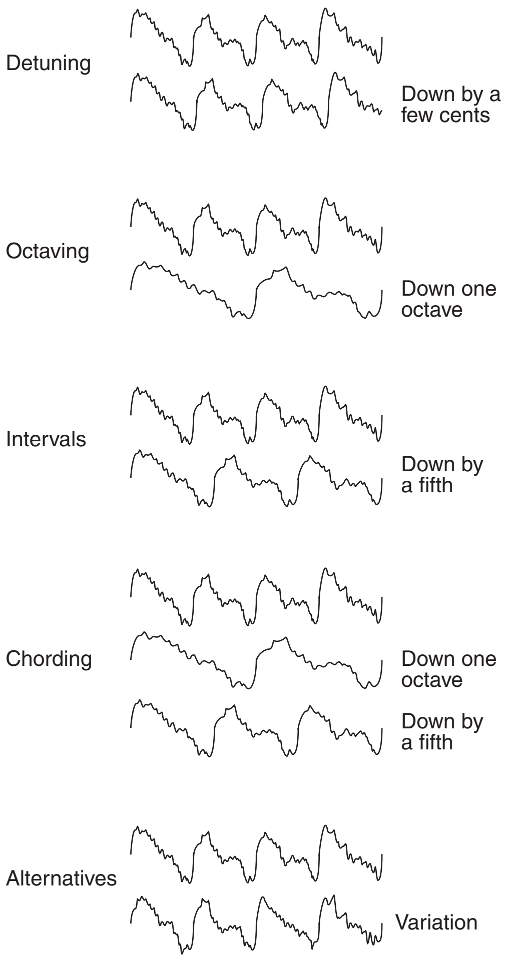


FIGURE 7.2.2
Doubled stacks.

Layering normally implies a time separation of the sounds. This conventionally means that one sound is used to provide an initial attack sound, whilst the other is used for the sustain portion of the sound. But this is just one approach to using two sounds that are independent of each other in time. Some other possibilities include:

- *Percussive and pad*: The combination of a fast attack/rapid decay sound with a **slow attack**/slow release pad sound can be very useful for providing accompaniment timbres from a single performance keyboard. By altering the playing from staccato to legato, or by using the sustain pedal, the level of the pad sound can be controlled.
- *Opposites*: By providing the two sounds with opposite envelopes, the result is a complex sound which dynamically changes between the two timbres. This is particularly effective with slowly evolving sounds,

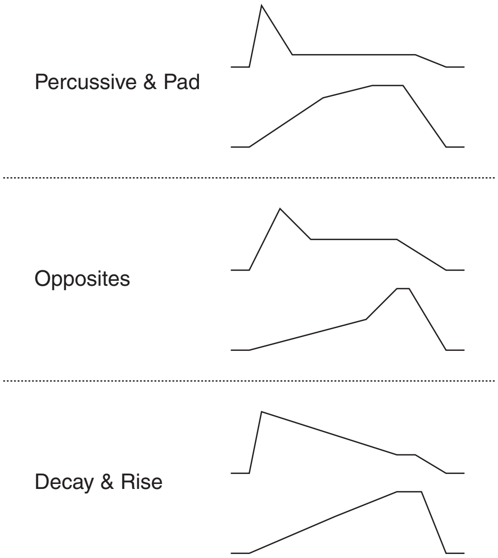


FIGURE 7.3.1
Layering is concerned with changes in envelopes over time.

or sounds that have some common elements and some contrasting elements.

- *Echo/reverb and dry*: By layering two sounds that have different acoustic 'spaces', it is possible to dynamically change the apparent 'position' of a sound. Dry sounds tend to be perceived as being close to the listener, whilst echoed or reverberant sounds are interpreted as being further away. Two sounds that have different timbres and different echo timings can be very useful for creating poly-rhythmic textures.
- *Pan position*: Layering sounds that have contrasting pan positions can be used to produce composite sounds which change their stereo position dynamically.
- *Slow decay and slow rise*: By using a sound that decays slowly and another sound that rises slowly whilst the first is decaying, the composite sound can have a 'sustain' sound which is not static, but which changes as the relative balance between the two layers changes.

Loops

Repeating or looped sounds can be played against pads, percussive sounds or even other loops. This provides continuous variation in the overall sound when it is held for a long period.

7.3.1 Splits

Although layering can be achieved by assigning both sounds to the same key, many synthesizers provide the ability to 'split' the keyboard, which allows different sounds to be used in different areas of the keyboard range. Atypical application

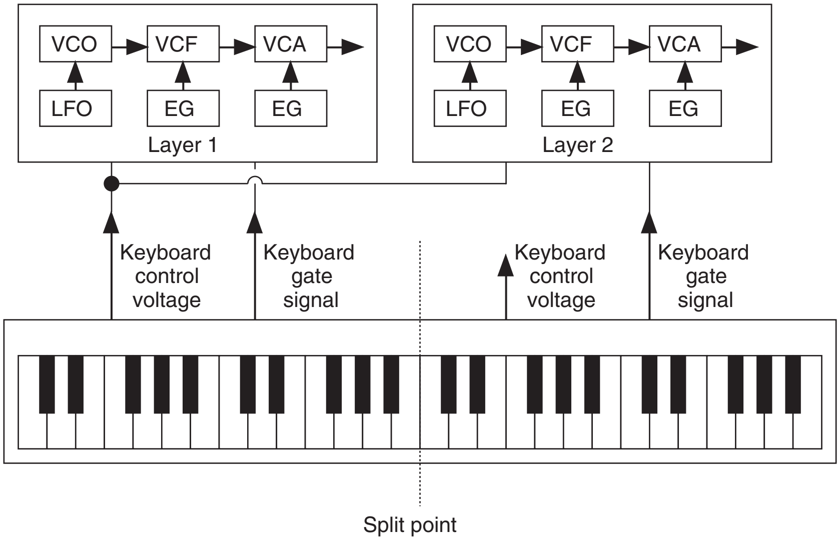


FIGURE 7.3.2
By splitting a keyboard and sending the keyboard CV to two 'layers' of a sound, it is possible to use the separate gates to control the second layer independently.

for jazz use might place a piano sound on the upper half of the keyboard, with a string bass sound on the lower half, perhaps with an additional brush sound. Less typically, the parts of a split can be used to produce different layers of a sound instead of different notes – this is easy to set up on an analogue modular synthesizer, although it can also be produced using MIDI equipment with specialized software. This allows control over which layers of a composite sound are produced on a note-by-note basis, which is more akin to the sort of control available to an orchestra and not a keyboard-based synthesist (Figure 7.3.2).

7.4 Hocketing

Hocketing is the name given to the technique of sending successive notes from the same musical part to different instruments rather to the same instrument (Figure 7.4.1). With the use of different (complementary and/or contrasting) timbres, or different pan positions, contrasting effects settings, or even slightly different amounts of detune, this can produce very complex sounding arpeggios and accompaniments and, combined with doubling and layering, can give the impression of a very detailed arrangement, when in fact all that has happened is that a few notes have been moved from one track to another, and a few tracks have been copied and pasted.

The general process to produce hocketing in a sequencer is to:

- listen to the track with one sound source;
- choose a hocketing criteria (see later);
- select the notes to be hocketed;
- filter the notes according to order, number, velocity, beat, time …;
- copy the filtered notes to another track;

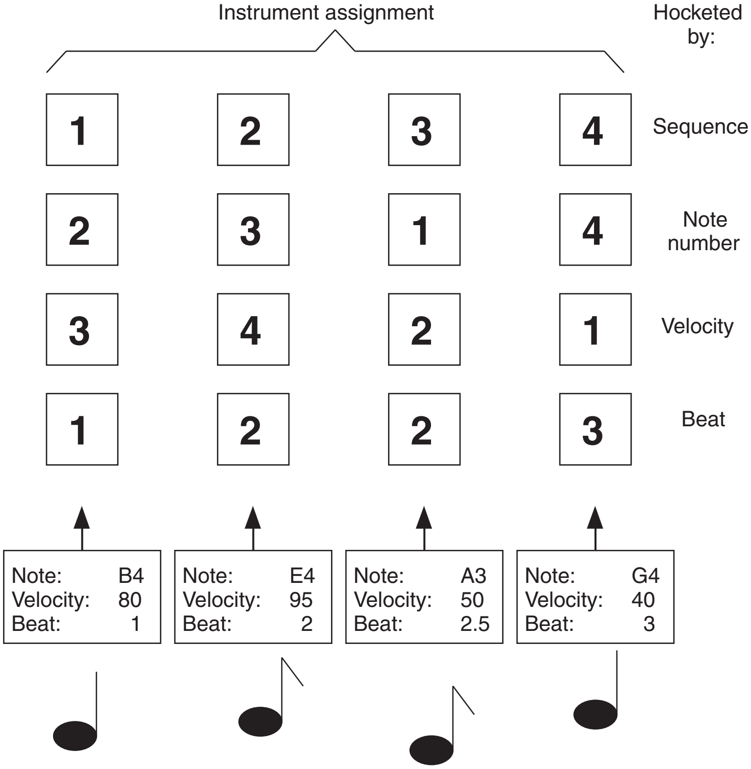


FIGURE 7.4.1 Hocketing involves changing the instrument that plays a note based on criteria like the MIDI note number, or the MIDI velocity, or the position of the note in the bar or in time. In this example, the 4 notes are hocketed to two instruments using the note number.

- assign the hocketed notes to a different sound source;
- set the two sound sources to contrasting or complementary timbres.

The sophistication of modern computer sequencers allows many variations of this process, and in general, they reduce the number of operations, whilst restricting the flexibility. This is often a useful compromise. In particular, hocketing of more than two instruments is becoming increasingly popular, and facilities to make this easy to set up are under ongoing development. Many of these variations of hocketing are given other names.

Hocketing can be achieved using several criteria:

- by note sequence order;
- by note number;
- by velocity;
- by beat;
- by time;
- by a controller;
- by a note event.

By note sequence order

Hocketing by note sequence order is the obvious one: you put the first note of the arpeggio or melody to the first instrument, the next note to a second instrument, and so on, with perhaps the third or the fourth note being allocated to the first instrument again. Some assignment algorithms provide more complex allocations, including random allocation of notes to instruments. This hocketing method is sometimes called MIDI note cycling or MIDI note randomizing.

With similar timbres for each of the hocketed instruments, the effect is quite subtle, and by using slight detuning, the instruments can produce very 'non-synthetic' ensemble effects. Using different pan positions for the hocketed instruments can be used to provide movement in a sound which would otherwise be static. With contrasting timbres, the effect of the hocketing is more like splitting the notes into separate parts.

By note number

By using the note number to hocket, there are a number of effects that can be produced. Hocketing with a fixed note number as a split-point will send all the notes above the split to one instrument, with those below the split going to the other instrument. Odd and even note number hocketing can break up chords into what sound like pseudo-inversions or complex arrangements (especially if the two instruments are transposed by one or more octaves relative to each other). Pairs of numbers can be used if the odd/even hocket results in too much spread of the notes in a chord. Chord splitting involves taking each chord and assigning hockets split around the middle note, with notes above the middle going to one instrument, and notes below going to the other. This can also be very effective if the assignment alternates so that the notes above go to one instrument for the first chord, then to the other instrument for the second chord.

By velocity

Using velocity to determine which instrument plays the sound involves selecting only those notes that have a velocity value above a specific value (half-way, 64, is a good starting point) and then allocating those notes to a different instrument. This is like the velocity splitting available on some samplers and S&S instruments, although the split-point can be edited as part of the score or sequencer information instead of as part of the sound. Hocketing by velocity can be particularly effective on sounds which move between accompaniment and melody roles. By using alternative sound sources for the velocity-hocketed notes, it is possible to have different snare sounds selected from different drum machines or samplers merely by editing the velocity. Reducing the velocity sensitivity of the sounds allows the velocity to be used as a mixing control rather than a dynamics control.

By beat or time

It is also possible to use the position in the bar to determine which instrument a note is hocketed to, or even the absolute time, in which case the hocketing is not related to the bar position at all. The hocketed instruments can have different timbres, in which case the timbre is reflected by the location in the bar, or by pan position, in which case the sounds can be made to move around in the stereo field in time with the music. This is much more effective and controllable than the 'random pan position' option which is available on some commercial synthesizers.

By a controller

By using a controller to determine how the notes are allocated to instruments, it is possible to change the hocketing by using any suitable controller. DJ cross-fader type controllers are good for hocketing between two instruments, whilst 3D or surround sound joystick controllers can be used for more than two instruments. 'Legato mode' uses a controller to select between the playing of two loops of note events and so allows hocketing between two rhythms of notes, where each pattern carries on from where the last one stopped, hence the use of the word 'legato'. This can produce very complex patterns that are still under the direct control of the performer.

By a note event

'Key switching' is the term for hocketing that is controlled by note events. The note events can be generated by a keyboard, either using a split or very low- or high-pitched notes, and they control hocketing of instruments played with the remainder of the keyboard. Key switching is often used for providing direct control over different samples of a single instrument. One hand is used to produce the controlling key events (usually the left hand), whilst the right hand plays the sound using the controlled instrument. This allows very detailed control over the use of instruments and can be used rather like velocity switching, but without the requirement for precise playing and the corresponding change of volume.

Key switching can be used as a user-controlled form of note sequence hocketing, where the performer directly controls the allocation of notes rather than the time sequence. It is often used to provide extra realism by providing variations of timbre for successive notes or similar timbres for successive notes of the same pitch, but different timbres for different pitches. The two-handed playing technique looks similar to monophonic synthesizer top-note priority playing.

A variation in using a special part of the keyboard, or a split keyboard, is to use the timing of the playing on the keyboard itself. By using the repetition rate of the playing to produce a controller parameter, this can be used to control hocketing of instruments. This is often called 'Trigger on playing speed', and allows effects like an instrument that is played only when fast runs of notes are played, or different instruments played when notes are played at different rates. This type of control is particularly effective when using a guitar controller, since it allows the performer control over what sound is used for specific runs of notes just by using their hands. A keyboard player would need to have very precise velocity control and use sounds that lacked velocity sensitivity, or use a foot or breath controller to achieve the same effect.

With a little practice, all of these 'hocketing' edits are relatively quick and easy to make on a computer sequencer, and yet they can greatly improve the detail and quality of the finished music. More advanced use of hocketing, combined with arpeggiation and Legato mode note event generation, can be found in a very flexible form in the Karma technology that is found in some Korg synthesizers.

7.5 Multi-timbrality and polyphony

Multi-timbrality is the name given to the ability of a single synthesizer to produce several different timbres at once. This has the effect of turning a single physical synthesizer or expander module into several 'virtual' ones, although some of the functions remain common: normally the effects, the overall control and the management of the sounds.

Multi-timbrality is an extension of the concept of 'stacking'. If an instrument can produce two different timbres at once when a key is played, then it is logical to extend this capability so that the two different timbres can be played independently. The history of multi-timbrality is closely connected to the development of polyphonic synthesizers and the differences between analogue and digital synthesis.

Many sounds and instruments are naturally monophonic, that is, one sound at once. Most people can only sing one note at once, and many acoustic instruments will only produce one note: flute, tuba and triangle are some examples of naturally monophonic instruments. The availability, price and complexity of early modular analogue synthesizers meant that they tended to be used as monophonic instruments too. Recording onto tape allowed the single notes to be combined into complete 'polyphonic' performances. Even when the synthesis resources were available, trying to keep two or more notes in tune and with the same timbre could be harder than using tape.

The first true polyphonic synthesizers were based around simplified analogue monophonic synthesizer voltage-controlled oscillator/filter/amplifier (VCO/VCF/VCA) circuits to provide the notes. These cards are often called 'voice' cards, since each card provides a single 'voice', rather like independent singers in a choir. These synthesizers could be operated in two modes: either with common control of timbre and independent control of pitch or with independent control of timbre and pitch. The mode with common control of timbre allowed true polyphonic operation, where several notes the same or similar timbre could be played simultaneously. The mode that provided the independent control of timbre allowed stacking and layering of sounds, which reduced the number of different notes that could be produced simultaneously. Because of the problems of tuning and controlling voice cards, there was a practical limit of about 8-note polyphony. Even with eight voice cards, the minor variations in tuning and timbre can produce a distinctive open 'feel' to the timbres produced, rather like the ensemble playing of a violin section in an orchestra. Eight-note polyphony is very restricting for producing multi-timbral music, and so analogue polyphonic synthesizers were used more for their polyphony than for their multi-timbrality.

Polyphonic digital synthesizers such as the Yamaha DX7 started out as monotimbral in 1983, although digital technology provided larger polyphony than many of the analogue mono- or multi-timbral polyphonic synthesizers: 16-note instead of the 4-, 5-, 6- or 8-note polyphony found in the analogue instruments of the time. But the precise tuning and identical timbres gave a very different 'feel' to these early digital instruments, and it was not until the early 1990s that the technology for emulating the imperfections of analogue polyphonic synthesizers began to be implemented in digital synthesizers. But the availability of larger polyphony meant that utilizing the multi-timbrality was easier and more effective. The introduction of 8-part multi-timbrality in 16-note polyphonic synthesizers was quickly followed by 16-part multi-timbrality, which has formed the basis of most synthesizer specifications ever since.

By the beginning of the twenty-first century, 64- or even 128-note polyphony had become more common, particularly in S&S synthesizers, and this allowed a single synthesizer to produce many polyphonic parts simultaneously. The need for more than one rackmounted synthesizer expander module was increasingly driven by sounds rather than polyphony. The fundamental limit of 16 channels on one MIDI port has been overcome by some manufacturers by providing more than one MIDI-in socket. Some synthesizers and workstations arrange for the built-in keyboard and sequencer to access one set of 16-part multi-timbrality, whilst one or more MIDI-in sockets provide access to the remaining sets of 16 parts of multi-timbrality.

7.5.1 Definitions

- Polyphony is the total number of different pitches or sounds that an instrument can make at any one time. For a single sound, the polyphony is the number of different pitched notes that the instrument can play simultaneously (Figure 7.5.1).
- Multi-timbrality is concerned with the number of different sounds or timbres that can all happen at once and played by at least one note each.

The word 'part' is frequently used to indicate a separate timbre, and although it can be used in the same sense as a musical 'part', the two terms are not necessarily synonymous.

The word 'voice' is also sometimes used to mean a single sound generating section: a monophonic part.

Thus, a 16-note polyphonic instrument can play a sound using up to 16 different pitches at any one time, whilst a 16-'part' multi-timbral instrument can play up to 16 different sounds or timbres at any one time. The two different numbers are independent, although the specifications for synthesizers frequently

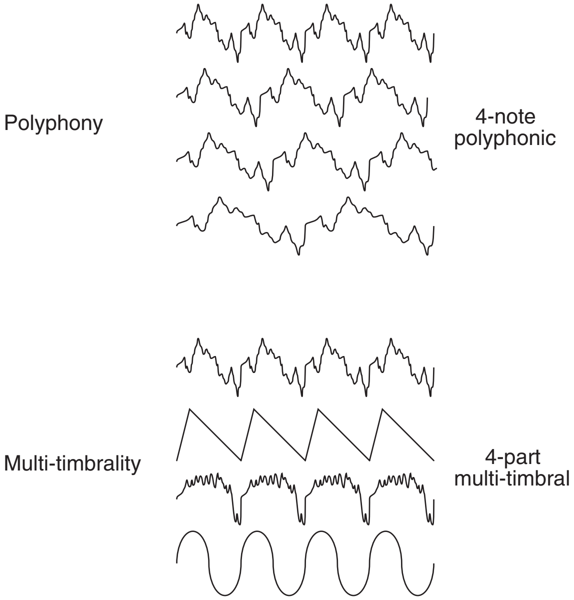


FIGURE 7.5.1
Polyphony is the total number of different pitches or sounds that an instrument can play simultaneously. Multi-timbrality is the number of different sounds or timbres that can happen simultaneously.

quote the same number for both the polyphony and the multi-timbrality. This has continued even when the multi-timbral parts have exceeded the 16 parts which can be carried by a single MIDI cable. Instruments with 32-note polyphony are frequently quoted as having 'up to' 32-part multi-timbrality, even though using all 32 parts requires the use of the keyboard and on-board sequencer as well as one MIDI-in socket, or two separate MIDI-in sockets and a multi-port MIDI interface on a sequencer or computer.

As a general rule, the polyphony should always equal or exceed the multi-timbrality, so a 16-note polyphonic, 16-part multi-timbral instrument is feasible, whilst a 2-note polyphonic, 16-part multi-timbral instrument is not possible: it can only ever make the sounds for two parts. Actually utilizing the 16 independent monophonic parts from a 16-note, 16-part multi-timbral instrument can be harder than it might initially appear because of the need for true monophonic sequences of notes for each part: any overlap can increase the polyphony, which automatically reduces the available multi-timbrality.

- 'Maximum polyphony' is a term used where the polyphony is dependent on the synthesis resources inside the synthesizer. In such a case, the use of simple sounds will result in the greatest polyphony, whilst complex sounds will reduce the effective polyphony.
- The allocation of synthesis resources is very dependent on the synthesis technique used, but a simple example based on oscillators will illustrate the general principles. If the synthesizer has 16 'oscillators', then the maximum polyphony is 16 notes, provided that each note only requires only one oscillator.

- The 'maximum' polyphony is therefore an exceptional case where only single oscillator sounds are used for each note. If the sound for a note is made up of two or more oscillators, then the polyphony reduces accordingly: 8-note polyphony when two oscillators are required per note, and 4-note polyphony when four oscillators are required for each note. The 16-note polyphony is thus a 'best case' rather than a typical value, and this should be taken into account when reading the specifications of synthesizers.
- In general, the best sounds will tend to use the full capabilities of the instrument, and so from the example above, this would be four oscillators per note, giving a 'minimum' polyphony of 4 notes. Typically, not all sounds will utilize the full synthesis capabilities and so the 'typical' polyphony would be between 4 and 8 notes.

The important element of the definition of multi-timbrality is the simultaneity. With a monophonic synthesizer, it is possible to make a large number of different timbres, but it is always mono-timbral, that is, one timbre at once. If a synthesizer has two separate sound generating sections, then it has two-part multi-timbrality, thus it is two-part multi-timbral. But by adding in a third sound generating section, each separate part that can happen simultaneously with other parts only adds one to the multi-timbrality. The number of available timbres does not affect the multi-timbrality; there may be a large number of ways of combining two timbres, but the multi-timbrality is fixed by the polyphony and the number of simultaneous parts, and not by the timbres.

7.5.2 Notes per part

Using multi-timbrality can require a surprisingly large polyphony: even a 64-note polyphonic instrument can only play four simultaneous pitches on 16 multi-timbral parts. Trying to produce 16 multi-timbral parts using only 16-note polyphony can result in only 1 note per part if everything plays at once, and many synthesizers have limitations as they approach their polyphony or part limits. Producing music using only one note per part also has problems; although orchestral arrangements use monophonic melody lines played on 'one note at once' instruments such as oboes, flutes and trombones, they use several players to get the required polyphony (and volume, of course). And an orchestra also has fixed limits to polyphony: if there are four violin players, then you can only have four separate violin parts.

The transitions between the notes are the cause of many of the problems when a synthesizer reaches its polyphony or part limits. This is easiest to understand by considering the case of making music using only one note per part on a polyphonic keyboard. Staccato playing, with gaps between the notes, gives controlled use of polyphony. But legato playing, tied notes, or overlapping notes can all cause short overlaps. Each overlap uses up 2 notes of the available polyphony, and because of the way that music tends to concentrate events around bar or beat intervals, the overlaps 'cluster' across parts. So it is likely that an overlap between 2 notes within one part will happen at the same time as overlaps in other parts. The required polyphony can thus be much higher than the apparent polyphony of the music: up to double in exceptional cases. Apiece of music written for a quartet of four monophonic instruments can have a peak polyphony requirement of 8 notes.

The release time of sounds can be a major contributing factor in causing problems with overlapping notes. Although the start of a note is when the key is pressed (or the MIDI note on message is received), the end of a note can last after the note has been released (or the MIDI 'note off' message has been received), because the synthesis resources are still being used to produce the sound as it fades away during the release segment of the envelope. 'Staccato' playing thus has a subtle but significant difference when playing electronic keyboards, whilst the notes written on the score may be separated by rests, the actual synthesizer may actually be playing legato notes because of long release segment times. The result is that an apparently low-polyphony piece of music may well require up to double the polyphony to avoid problems (Figure 7.5.2).

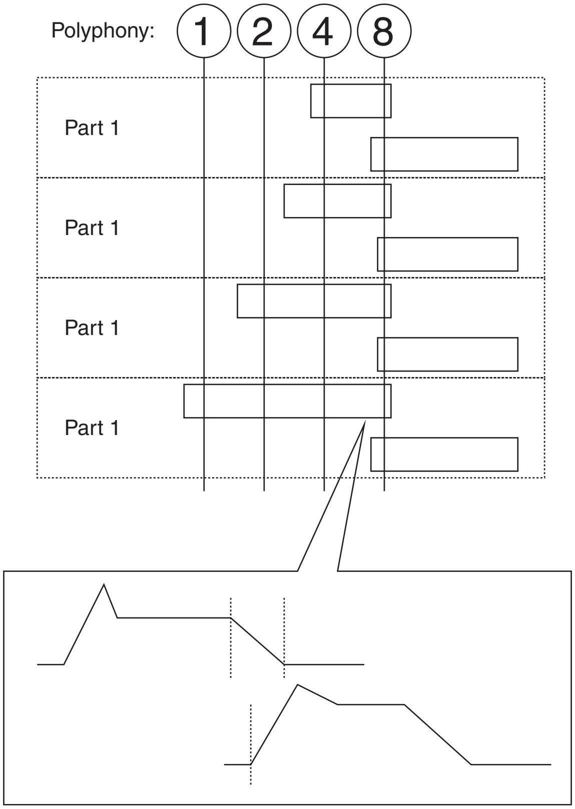


FIGURE 7.5.2

The peak polyphony depends on the overlaps between notes. Even if there appears to
be a gap between the 'gate' signals, the release of one sound can carry on past the start of the attack of the next note.

Here, 'old' means notes that are already making sounds, whilst 'new' means notes that have been pressed and which are about to make sounds.

When the polyphony limit is reached, notes are inevitably lost. If there are not enough resources to play all the required notes at once, then some that are already playing will have to stop to enable the new notes to be heard. This produces the effect known as 'note stealing', where 'old' notes make way for 'new' ones. The behavior of synthesizers when note stealing is taking place varies: it depends on the method used to assign the notes to the available sound-making resources inside the instrument, in an analogue synthesizer these would be the voice cards, and so these resources are often called 'voices'. A 'voice' is thus an imaginary part of a synthesizer which is capable of playing only one note, rather like a virtual polyphonic synthesizer voice card. In digital instruments, the distinction of physical hardware does not usually exist, and so 'voices' are more flexible (see Section 7.5.1).

7.5.3 Note allocation

The way that the required notes are assigned to the available 'voices' in a synthesizer is called note allocation or voice assignment. There are a number of techniques for doing this, and they all extend the basic idea of reusing the voice resources on demand. This underlying process is called cyclic assignment, and it assigns the incoming notes to voices in order: the first note is assigned to the first voice, the second note to the second voice, etc. The notes that are played could come from either the instrument's keyboard, from an internal sequencer, or via MIDI from an external keyboard, instrument or sequencer (Figure 7.5.3).

An 8-note polyphonic synthesizer would contain eight of these 'voice' resources, and so after all eight voices had been assigned, then the next available voice would be the first one again. The ninth note would thus cause the first voice to stop playing the first note, and this voice would be used to play the ninth note. Exactly the same 'stealing' would occur with the seventeenth note – the first voice would again be used.

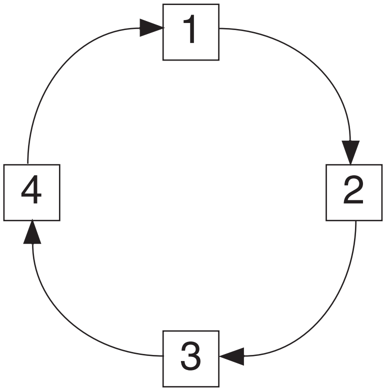


FIGURE 7.5.3 Cyclic assignment assigns incoming notes to available voices in sequence. In this example, there are four voices available, and the first 4 notes are assigned to these. The fifth note will replace the first note on voice 1.

Cyclic assignment does not take into account the status of the voices. If the first note is sustained, whilst the next seven are short staccato notes, then seven of the eight voices will be 'available' for the ninth note since they are not actually producing a sound. A pure cyclic note assignment strategy would ignore this and assign the ninth note to the first voice, thus stopping the sustained note. This approach thus wastes the voices, since the assignment is arbitrary and does not use the available resources efficiently.

There are many ways to improve the assignment strategy by making it responsive to the incoming required notes and the availability of voice resources. Some of these 'dynamic voice allocation (DVA)' approaches include the following:

- note reserving;
- part priority;
- voice status;
- envelope status;
- volume status;
- repeated note detection;
- sustain pedal detection.

Note reserving

Sometimes a particular part needs to always have a specific polyphony. For example, a note stolen from a sustained melody line or a solo passage will be very noticeable to the listener whilst a note stolen from an accompaniment pad sound will be less apparent. 'Note reserving' allows a fixed allocation of polyphony to timbre where a specific part will always have a given polyphony. So a melody part might be allocated a polyphony of 2 notes so that any overlaps will not cause assignment problems. The disadvantage is that when the melody part is not being used then the voices are still not available, and so the utilization of voices is not very efficient.

By using fixed polyphony, it is possible to make 8-note polyphony sound like 12 notes. Two contrasting timbres are allocated to two parts, and assigned to the same pitch control channel (or MIDI channel). Note reserving is then used to allocate the voice resources asymmetrically: 6 and 2 notes, for example. The result is like a 6-note polyphonic version of the two-timbre stack, because the listener will tend to hear the last 2 notes played, which are always played with the two timbres, and not the remaining 4 notes which are played on only one timbre. This is very effective when the two timbres are detuned relative to each other, since the result sounds like 12 detuned notes when only 8 are actually being used.

Part priority

Assigning priorities to parts allows a melody or a drum part to be relatively immune from note stealing, without the need for the permanent allocation of note reserving. By assigning a priority number to parts, it is possible to force notes to be stolen from the 'less important' voices or parts which have lower priorities.

The highest priority parts are only stolen when all other resources are in use. Time priority also needs to be considered: a higher priority needs to be given to the most recent notes. Despite the complexity of allocating priorities, this method gives a more efficient use of the available polyphony than fixed reservation.

Voice status

By monitoring the status of the voices, it is possible to alter the cyclic assignment so that incoming notes are only allocated to voices which are not playing sounds. This means that some mechanism is needed to keep track of the 'playing/not playing' status of each voice. Note stealing only needs to occur when there are no 'not playing' voices available.

Envelope status

Some portions of the envelope of a sound are less important than others. Stealing a note from a sound that is in the release segment is probably less audible than stealing from a sound that is in the attack segment. By monitoring the envelope status of each voice, it is possible to arrange to only steal sounds that are in the appropriate segments: normally the release or sustain segments.

Volume status

Volume, controlled by the note velocity, overall volume or envelope position, can also be used to determine whether a voice is a suitable candidate for stealing. Lower volumes or audio levels will be less audible if they are stolen by a voice assigner.

Repeated note detection

If the same note is allocated using a cyclic assignment, then the allocation will run through each of the available voices in turn, even though the same pitch is being played each time. By detecting the repeated note, and reassigning it to the same voice, perhaps with a retriggering of the envelope, the remaining voices will not be disturbed.

Sustain pedal detection

The sustain pedal on a synthesizer or MIDI device causes the notes that are being played when it is activated, or played whilst it is active, to be held at the sustain segment of their envelope. If the number of incoming notes exceeds the available polyphony, then notes will be stolen, and because of the time-based nature of the sustain pedal event, the note stealing decision should probably be on a cyclic 'oldest-note' basis.

In order to allow the reservation, priority, status and other parameters to be tracked, the synthesizer needs to maintain running lists of all the parameters which might affect the note allocation. This changes the 'note assigner' circuitry, from the simple counter that is required for a cyclic assignment technique to one which has to maintain detailed records for several separate voices.

Such an 'intelligent' assigner can add a considerable software overhead, which may mean slower response times from the synthesizer to incoming notes. There is thus a compromise between the complexity of the note allocation scheme and the response of the synthesizer.

7.5.4 Using polyphony

Multi-timbrality and polyphony tend to encourage the use of the layering, splitting and hocketing techniques described in this chapter. Polyphony can be either overused or underused: too much doubling or detuning can over-thicken a sound and instead make it dense and obscure the harmonic structure; too little available polyphony can be improved by reserving or prioritizing parts, and perhaps even thinning out the chords used for accompaniment or reducing the number of simultaneous drum events.

There are a wealth of low-cost synthesizer expander modules which are available, both new and second-hand which can be used to increase the available synthesis polyphony. As has been mentioned several times, a hybrid or mix of several instrumental sounds using different synthesis techniques can be very powerful.

7.6 GM

General MIDI, (GM) (White, 1993) was introduced as an extension to the MIDI specification in 1991 and is a shared set of specifications and guidelines agreed by the MIDI Manufacturer's Association and the Japan MIDI Standards Committee.

GM uses the idea that most music can be reproduced using a small set of commonly used instrumental timbres. In effect, the concept provides a minimalistic 'electronic' orchestra. Many of the sounds chosen are orchestral or 'popular' in origin, although some additional synthetic sounds are also included. There is a distinctive GM logo which can be found on compliant instruments.

There are 128 sounds, plus a drum kit mapping, in the basic GM sound set. The sounds are divided into 16 categories or 'groupings', each containing 8 sounds:

- keyboards;
- chromatic percussion;
- organ;
- guitar;
- bass;
- strings;
- ensemble;
- brass;
- reeds;
- pipes;

- synth lead;
- synth pad;
- synth effects;
- ethnic;
- percussion;
- sound effects.

The numbering and names of the groupings and sounds are given in the GM specification. The drum kit (percussion) receives MIDI messages on channel 10, whilst the remaining channels can be used for any purpose: the allocation of channels to parts is left to the user. The GM specification requires that the polyphony of the GM playback device should be a minimum of '24 fully dynamically allocated' notes, 16 'dynamically allocated' notes for instrumental sounds and 8 notes for percussion. GM modules cannot always meet this requirement. GM playback devices should also be 16-part multi-timbral.

The many GM sound sets that are produced by manufacturers are all intended to sound as similar as possible, so that a MIDI file created using one GM device will sound much the same when played back using another GM expander. This extends to how they combine together, how they respond to MIDI performance controllers such as pitch bend, velocity and volume. The result is a largely uniform range of instruments that can reproduce the same sounds in a predictable way.

The actual sounds that are included in the GM sound set are not intended as anything more than a means to an end: making a form of 'paint by numbers' music, where the MIDI file contains the information for both the musical events and the musical sounds. With minor differences, a given MIDI file will produce very similar results on any GM playback device. For use as a synthesis medium, GM offers very little. The sounds are chosen for their ubiquity of application, and many GM playback devices do not offer any editing of the sounds – sample replay is frequently used for producing the GM sounds. GM playback devices can be used in combination with synthesizers to produce composite sounds.

GM playback devices can range from professional synthesizers, through small expander modules, to personal computer (PC) sound cards and even software-only versions. Two manufacturers have introduced their own additional extensions to the basic GM specification, and these allow extra functionality to be provided by specifying additional sounds and controls: Yamaha's XG and Roland's GS systems are similar in many respects, but are not completely cross-compatible.

In 1999, an enhanced superset of the original GM specification was released. GM2 was designed to enable better exploitation of the powerful facilities of the synthesizers of the time, and the GM specification was renamed GM1. In contrast, 2001 saw the publishing of GM Lite, a reduced specification intended for use in devices such as mobile phones. These new specifications do not change the basic concept behind GM: define a standard set of synthesis capabilities across sound generating devices in order to maximize the chances of getting acceptable and consistent playback of music files.

7.6.1 Music files

Standard MIDI files (often suffixed with the file extension .SMF) need not use GM, but in practice, most .SMF or .MID files will use the GM1 sound and drum mappings. GM Lite is a pragmatic solution to the limited synthesis features of some devices such as mobile phones and so is usually used only as part of a process of converting conventional GM files to suit the specific devices. GM2 is backwards compatible with GM1, but has necessarily limited compatibility in the other direction.

The idea of producing music files which will play on a wide variety of different end devices is not new. These fall into three main types:

1. Waveform.
2. Waveform and representational.
3. Representational.

MIDI files are representational. They describe the notes are played, the sounds to be used and how the notes should be played. MIDI files can thus be seen as a sophisticated form of abstracted music notation, since the performance information is explicit, and can be edited. Changing the sound of a bass guitar requires only a change of the mapping of the sound: either a different sound or a change to the settings of the synthesizer producing the sound. The flexibility of representational files can result in the substitution of a very different sound: a whistle to replace the bass guitar or drum sounds. These changes are playback-only modifications: the file itself normally remains unaltered.

In contrast, MP3, WAV, AIFF and other recorded files are actually just audio files, where the actual waveform of the sound is stored, albeit sometimes in highly compressed formats. Audio files do not store anything about the notes are played, the sounds used or how the notes should be played, other than by capturing the actual sound.

Audio files are just records of the sound, and performance information is implicit and cannot be easily edited. If you do not like the sound of the bass guitar in a recorded audio file, then it is very hard to change it. If a technique is available for replacing the bass guitar with another sound, then this would require the creation of an altered audio file.

MOD files and MIDI DLS files are mixtures of the waveform and representational types. Both mix notation systems with samples of sounds. The notation is used to control the playback of sounds which are just samples of the required sounds. These files can produce very good reproductions of recorded music, but with very small file sizes. Mixed representations of music are good when synthesis techniques are limited. Vocal or spoken performances are one example where MIDI files used to be severely limited by the lack of vocal synthesis capability, but Yamaha's Vocaloid vocal synthesizer software, released in 2003, showed that even singing could be synthesized, albeit by a technique which mixes waveform and representational approaches.

Some music programming languages such as CSOUND can replay audio, or control sample replay or synthesis from notation, and so can utilize waveform, mixed or representational types of music file.

The future of music files seems to be the mixed type. Audio file compression is moving increasingly towards an approach where the coding technique analyzes the content to derive notation-like control signals that are used to control sample-like fragments of audio. MIDI files are increasingly using samples rather than synthesis for part of their sound generation.

7.7 On-board effects

In the 1970s built-in effects processors were the exception rather than the rule. Reverb, chorus and echo were added to the output of the synthesizer in the studio by using 'out-board' effects units. Built-in effects processors first began to appear in polyphonic hybrid synthesizers in the 1980s as a low-cost method of disguising the use of single digitally controlled oscillators (DCOs) instead of dual VCOs. String and piano sounds were significantly improved by the use of small amounts of chorus effect, and in fact, this spelled the end of the 'string machine' as a separate instrument. Whilst simple effects processors were commonly added to hybrid and S&S polyphonic synthesizers throughout the 1980s, it was not until the end of that decade that digital FM synthesizers were fitted with built-in effects processors. The 1990s have seen the effects unit change from being a simple 'afterthought' to an integral part of the synthesizer by the start of the twenty-first century.

7.7.1 Effects history

Although effects processors have been used to process most electronic (and amplified) musical instruments for many years, the addition of effects processors to synthesizers has been a gradual evolution.

The effects types which follow are organized into a rough chronological order of introduction.

- Reverb.
- Chorus.
- Echo.
- Automatic double tracking (ADT).
- Phasers and flangers.
- Ring modulation.
- Exciters, compressors and auto-wah.
- Pitch-shifters.
- Distortion.

Reverb

Reverberation is the effect produced by almost any acoustic environment: a short delay (the 'pre-delay') is followed by a series of echoes from the boundaries (the 'early reflections') and then echoes of those echoes (reverberation) which decay in volume (the 'reverb time') as energy is transferred from the audio to the environment. Small containers have short delays whilst large rooms have long delays. The timbral quality of the reflections and reverberation are determined by the boundaries of the environment: smooth, shiny concrete walls will give long reverberation times and only slight high-frequency attenuation, whereas sound absorbent material such as curtains will give short reverberation times and significant high-frequency attenuation.

Sounds without reverberation sound empty and synthetic – the human ear expects to hear sounds played in real spaces, and removing the effect of the space can sound very 'wrong'. Synthesizers without reverberation emphasize this perception, so adding reverberation is almost essential unless the synthesizer will be heard in a naturally very reverberant acoustic environment. Many commercial synthesizers suffer from having too much reverberation on their preset sounds – based on the assumptions that 'more is better' and that the synthesizer will be auditioned using headphones. (It is easy to show that large amounts of reverberation can detract rather than enhance a sound: listening to a piano being played in a ballet or dance rehearsal room with lots of mirrors to reflect the sound is an excellent tutorial exercise.)

In the 1970s, reverb usually meant either a spring-line reverberation unit or a tape-based fedback flutter echo unit. The spring-line was prone to mechanical interference (they were microphonic and knocking the casing was audible in the audio output), whilst the tape machines were prone to mechanical failure (tape loop splice failures or wearing out of components because of long periods of continuous use). In the 1980s, analogue bucket-brigade delay lines were replaced by digital effects, and the quality of reverberation improved markedly. The 1990s saw sophisticated digital signal processing (DSP)-based reverberation algorithms, with features such as early reflections, pre-delays, and room size simulations. Reverberation has now become so low in cost to implement that it has become largely ubiquitous: it is provided even on low-cost minimalistic GM modules.

On a synthesizer, reverberation can also be approximated by using an envelope release curve which starts out decaying rapidly, but then lengthens the release time as the level or volume of the audio signal decreases.

Chorus

The chorus effect is a cyclic detuning of the sound, mixed with the original sound. This cause the same type of phase cancellations that are associated with several performers playing the same notes on acoustic instruments: the violin section in an orchestra is a good example. Chorus is normally achieved by delaying the audio signal slightly in time by a few tens of milliseconds, and then changing the time delay dynamically. This has the effect of changing the pitch as the delay time is altered, and so produces the detuned chorusing effect. Chorus can also be produced by deliberately detuning two VCOs, oscillators or sounds.

Echo

Echo is a repetition of the original audio signal after a specific time delay, usually hundreds of milliseconds or more, and at a lower volume. It simulates the effect of having a remote large object from which audio sounds will be reflected. Rapid echoes are often called 'flutter echoes', whilst the timbral quality of echoes is largely determined by the object from which the audio sounds are reflected.

Echo is produced by time delaying the audio signal, and feeding back the output to the input through an attenuator to produce additional repeat echoes. Echo can also be simulated by retriggering envelopes using a low-frequency oscillator (LFO).

ADT

ADT is just echoes with a very short delay time: tens of milliseconds. It can be used to 'thicken' some sounds, although because it produces a fixed set of cancellation notches in the output spectrum, it also produces timbral changes that sound 'metallic' in timbre. Turning off the LFO modulation in a chorus effect produces ADT. If the delay is exactly one bar long, and the music being played is very repetitive, then ADT can produce a smearing or blurring effect when the music changes.

Phasers and flangers

Phasers and flangers are variations on the chorus effect, with a mixing of an undelayed with a delayed audio signal, but with feedback from the output to the input. Phasers use a phase shift circuit, whilst flangers use a time delay circuit. In both cases, cancellations occur when the delayed and undelayed audio signals are out of phase, and so a series of narrow cancellation 'notches' are formed in the audio spectrum. The spectrum looks like a comb, and these filters are sometimes known as 'comb' filters. As the phase shift or time delay is changed by an LFO, the notches move up and down in frequency. Phasers produce notches that are harmonically related because they are related to the phase of the audio signal, whilst flangers produce notches that have a constant frequency difference because they are related to the time delay.

Ring modulation

Ring modulation takes two sounds and produces sum and difference frequencies based on the frequency content of the input sounds (see Section 2.4.3). In effects processors, ring modulation normally uses an LFO or audio oscillator for one source sound, and the signal to be processed as the other. Ring modulation can make major changes to the timbre of a sound, normally making it metallic or robotic in sound. It is particularly useful for processing drum and percussion sounds, whilst for special effects it can provide sounds which are often suitable only for science fiction genres.

Exciters, compressors and auto-wah

By the 1990s, the effects processors that were being incorporated into leading edge commercial synthesizers were similar to the top models of stand-alone effects processors from the same manufacturers. The range of effects programs expanded to include additional effects which were not based on time delays, such as exciters (small amounts of frequency-dependent harmonic distortion), compressors (dynamic range reduction) and auto-wah (envelope following/triggered filters). By the middle of the 1990s the first effects processors designed specifically for use with 'multi-timbral' synthesizers began to appear.

Pitch-shifters

Pitch-shifting is a variation on the detuning that takes place in a chorus unit, and is sometimes called harmonization. Instead of cyclically changing a time delay, the audio signal is stored at one rate and read out at another which produces a fixed pitch-shift. Slight pitch-shifts produce subtle chorus effects, whilst larger pitch-shifts produce transpositions (often at the loss of audio quality). Feedback from the output to the input can produce sounds that transpose up or down repeatedly from the initial input.

Distortion

Guitar-type 'fuzz' ranging from subtle harmonic distortion through to gross distortion can improve electric guitar and solo synthesizer sounds. Small amounts of distortion can also help to enhance any subsequent effects processing of sounds with limited harmonic content: filters, phasers and flangers can sound very poor if they are processing sounds which are little more than simple sine waveforms.

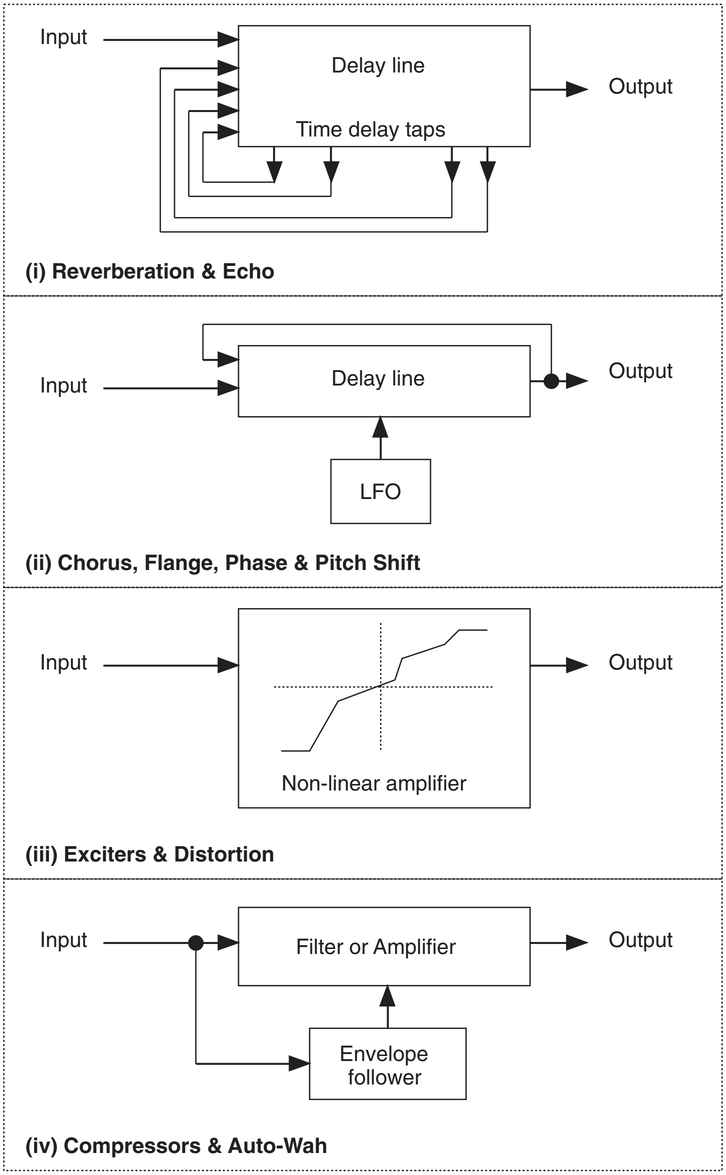
Distortion effects have very different results when used with monophonic and polyphonic sounds. Monophonic sounds are affected timbrally, with additional harmonics being added to produce a brighter sound. Polyphonic sounds are affected melodically, since the distortion produces a large number of new frequencies which are often not harmonically related to the input frequencies. These 'intermodulation' products are similar to the FM and ring modulation effects described in Section 2.4.3. In some circumstances, the results are useful: guitar power-chords, for example. But the result is often discordant: sounding rather too much like noise (Figure 7.7.1).

7.7.2 Effects synthesis

Until the early 1990s, the majority of effects processors that were built into synthesizers were still designed as an additional feature rather than as an integral part of the synthesizer – and they were usually little more than simple reverb or chorus units. This is most apparent when the topology of the effects processors

FIGURE 7.7.1

Effects summary. (i) Reverberation and echo can be produced using a tapped delay line. (ii) Chorus, flanging, phasing and pitch-shifting can be produced using a delay line whose delay time is modulated using an LFO. (iii) Exciters and distortion use non-linear amplifiers to produce harmonic distortion. (iv) Compressors and autowahs use an envelope follower to filter or attenuate the signal.



is examined: before the 1990s, most had a single monophonic input but produced stereo outputs. The output of the synthesizer section thus needed to be combined into a single channel before being processed by the effects, despite most synthesizers producing stereo outputs directly. True stereo effects, where the input and output are stereo, and the effects are part of the synthesis, only began to appear in the 1990s (Figure 7.7.2).

Using effects only for post-processing of sounds produced by a synthesizer is probably a consequence of the use of simple reverb or chorus units in early polyphonic hybrid synthesizers. As the effects have increased in complexity

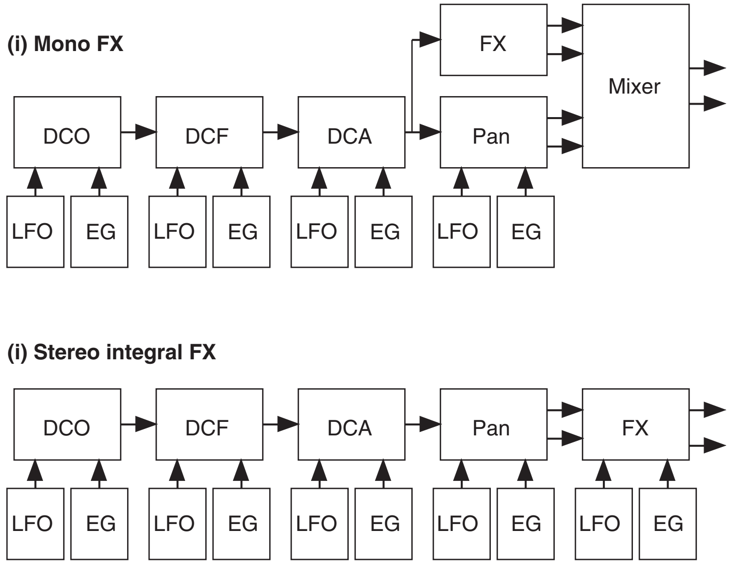


FIGURE 7.7.2 (i) Mono input effects processors are added into the output of some synthesizers, but they cannot use the stereo output, and so are mixed in with the output of the Pan processor. (ii) Stereo input effects processors can be integrated into the synthesizer, including using LFO and envelope generators (EGs).

and become an essential part of the timbre-forming modifier section of the synthesizer, the need for linking the effects to the synthesizer controllers has correspondingly increased. By having some of the important effects parameters controllable in real time from the synthesizer, the effects processing can be made an integral part of the synthesizer. Chorusing or flanging whose depth changes with filter cut-off or panning which also changes the reverb time are just two of the many possibilities which are often not possible with stand-alone 'out-board' effects processors which are not part of a synthesizer. This idea is not new: many modular analogue synthesizers have dynamically controlled effects, but it has only recently become available on commercial digital synthesizers.

When the effects become part of the 'sound', the 'traditional' isolation between the effects and the rest of the synthesizer does not work. Instead of choosing a sound, and then choosing an appropriate effects setting, the sound and the effect need to be linked together. This is feasible whilst the synthesizer is only producing one sound, but poses a problem when it is being used multitimbrally. Whereas most synthesizers can be split into 'voices' which each produce one note of a specific timbre by multiplexing the same circuitry, an effects processor requires a separate complete dedicated audio processing section for each timbre that it is processing. Most stand-alone effects processors are designed to process just one stereo signal, rather than anything up to 16 stereo signals simultaneously.

This means that an effects processor intended for use as part of a multitimbral synthesizer is not just a minor revision of an existing stand-alone – it requires additional real-time control inputs and the duplication of a large proportion of the audio processing. A partial solution developed during the late 1980s and early 1990s was to provide several simple effects processors and to provide independent inputs and outputs via effects send and return controls in much the same way as on an audio mixing console. Since many effects processors were already being designed to produce several simultaneous effects, this avoided any need for a major reworking of existing stand-alone designs. For multi-timbral synthesizers, this meant that two, three or perhaps four separate effects would be available but that these were shared amongst all the parts. The topology of the effects processors was often predetermined by their original conception as a series of effects, with the result that a chorus or echo effect would always be connected to a reverb, and only limited combinations of chorus and reverb would be available, and then only globally (Figure 7.7.3).

Korg's 1995 Trinity workstation featured multi-timbral effects, and so a sound could have the same effects applied when it was played on its own, or as part of a complex multi-part stack of sounds.

This limitation was first addressed in the mid-1990s by providing global reverb and chorus effects processors, and additional separate processors which could be applied to individual parts. This 'global and individual' approach changed the nature of the effects processing inside a synthesizer away from the stand-alone 'out-board' studio use and started the development of a new type of effects processor. This process was completed when the first 'multi-timbral' effects processors were introduced into high-performance workstations in the second half of the 1990s. Although still incorporating some global reverb and chorus effects, separate effects processing for each part meant that the effect could be treated as part of the modifier section of the synthesizer (Figures 7.7.4 and 7.7.5).

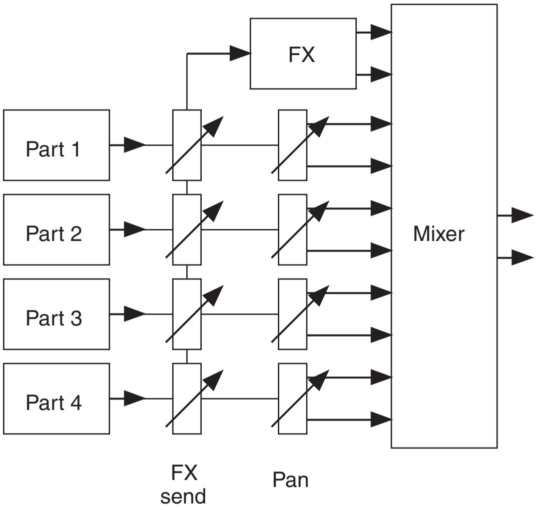


FIGURE 7.7.3 Some multi-timbral synthesizers use global FX busses to enable any of the parts to have an effect applied to them. This means that only one effect is available globally. More sophisticated synthesizers may provide two or more separate effects processors, but it is rare for there to be the same number of effects as parts.

7.7.3 Software effects

In the 1970s and early 1980s, the majority of effects processing used analogue circuitry. The late 1980s and mid-1990s saw this being replaced by DSP as part of the general trend to replace analogue circuits with digital processing. By the late 1990s, the processing power of ordinary microprocessors was such that they could carry out the same audio processing as the specialist DSP chips of 5 or more years before. This meant that a PC could be used as an audio processor, although a number of key elements needed to be in place for this to change from a specialist application to the norm. These triggers included:

- Audio input/output capability.
- MIDI and audio sequencers, rather than just MIDI.
- Sufficient processing power in the PC.
- Fast hard drives and cheap random-access memory (RAM).

It was not until PCs acquired these capabilities that the scene was really set for audio effects to move from hardware to software. From the vantage point of the twenty-first century, it is hard to appreciate just how limited the basic audio capabilities of many PCs were in the early 1990s, or how slow the hard drives were, or how expensive RAM was … but by the middle of the 1990s, the available processing power was sufficient to allow audio processing, and one final piece of technology changed things forever: the 'plug-in'.

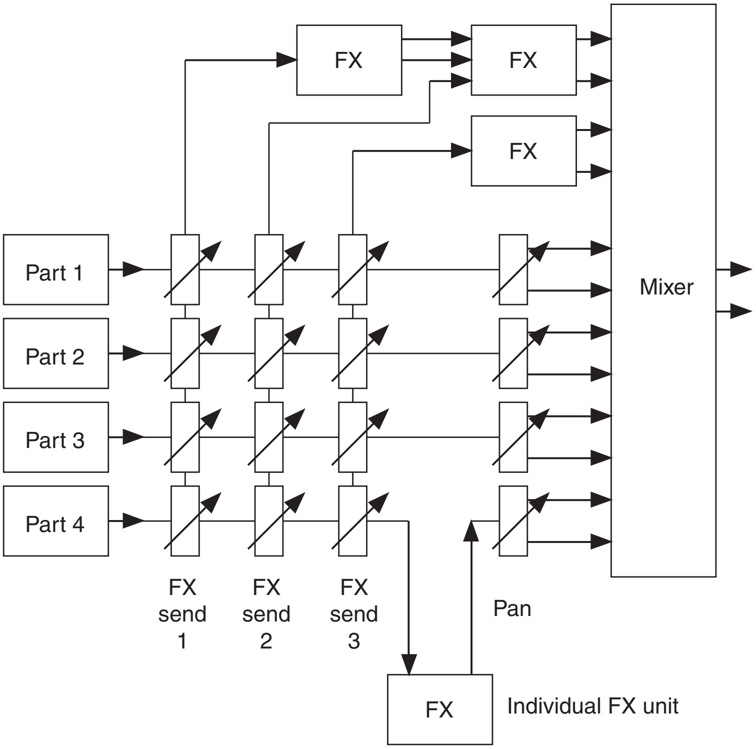


FIGURE 7.7.4 Some synthesizers allow access to the individual sections of multi-effects, so that a chorus followed by a reverb can then be used as a chorus and reverb as well as just a reverb. Individual effects processors that can be inserted into just one part are also found in some synthesizers.

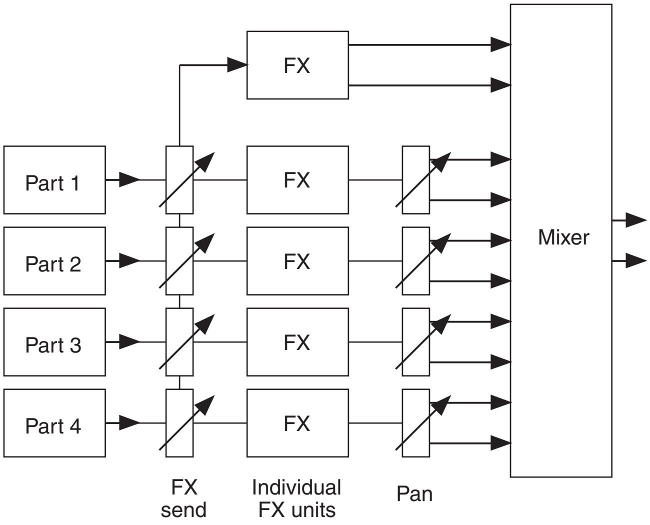


FIGURE 7.7.5 Fully integrated multi-timbral effects allow each part of a synthesizer arrangement to have the same effects when played individually or as part of the whole.

Taking a cue from the 'plug-in' method of allowing extra functionality to be added to software that had proved so popular in graphics applications suh as Adobe Photoshop, the audio effects 'plug-in' was created to allow third-party developers to create and market their own audio processing software for use inside audio sequencers. A 'plug-in' is just software, but instead of being independent and stand-alone, it requires a larger piece of 'host' software that provides it with the inputs, outputs and other essential services so that it can work effectively. It is really just a software equivalent of a piece of out-board studio effects equipment.

In the case of a twenty-first century audio sequencer, the generic audio mixing and routing support is part of the audio sequencer, but the effects are typically provided via a 'plug-in' architecture. The creators and publishers of the audio sequencers include support for 'plug-ins' because it means that they can allow users to customize and enhance the software themselves. It also allows a wider range of functionality to be available, and is therefore especially suited to specialist or unusual requirements. In about 10 years, the third-party marketplace for audio 'plug-ins' has grown from reverb effects to complete 'soft' recording systems.

The 'plug-in'

The software environment that is required in a 'host' to provide the support for an audio 'plug-in' is in six main parts:

1. initialization;
2. audio input;
3. audio output;
4. control signals;
5. user interface;
6. termination.

The initialization tells the 'plug-in' software to start up: this involves operations such as setting up its initial state, requesting RAM, registering with the host software and starting operation. The audio input and output can be streams of audio data or blocks of memory containing audio data. The control signals can be a wide variety of types: MIDI commands, general-purpose digital signals, trigger signals or digitized voltages. The user interface allows the 'plug-in' writer to provide a graphical interface for the end user to control the settings of the 'plug-in'. The termination causes the 'plug-in' software to unregister from the host software, free up memory, and finally, cease operation.

Simple digital effects such as reverb are programmed from a chain of time delays. Audio signals are passed through this chain and the outputs from several parts of the chain are mixed back into the chain, normally earlier in the chain. These feedback loops have gains of less than one, and so the audio signals fade away. Other effects are merely conversions of analogue effects into their digital equivalents and expressed in software rather than dedicated signal processing hardware. Synthesizer and sample-replay 'plug-ins' produce just an output audio signal.

Most users will only audition 'plug-ins' in order to choose their favorites, but user-programmed 'plug-ins' can be produced from various resources ranging from high-level graphical tools to low-level code compilation.

'Plug-in' history

One early ancestor of software synthesizer 'plug-ins' was Digidesign's Turbosynth, a software synthesizer released for the Apple Macintosh computer in 1988 which used a graphical user interface (GUI) to assemble audio oscillator and modifier modules together to produce sounds. Turbosynth produced audio sample files as its main output.

Digidesign's Pro Tools audio editing and sequencing software, released in 1992, gradually acquired a number of hardware add-ons to provide additional audio processing and sample-replay capabilities, and these were consolidated in 1995 with the launch of TDM, a digital bus designed to allow more flexible use of hardware expansions. In 1996, Steinberg released the first version of VST, their audio 'plug-in' standard, and other sequencer manufacturers followed with their own different and incompatible 'plug-in' formats. Many 'plug-ins' are platform-specific, meaning that they will only run on a PC or a Macintosh, and some formats are also platform specific. The early twenty-first century has seen some attempts to provide single 'universal' audio 'plug-in' formats that will work across platforms.

7.7.4 Using effects

The user interface metaphor which is almost always used for the output of multi-timbral synthesizers, especially in the context of effects processing, is the mixing console or mixer. This is an example of the way that synthesizers, sequencers and computers have gradually incorporated external equipment. First, the effects were built in, and then the mixer that would have mixed together several synthesizers becomes part of a multi-timbral synthesizer, and finally, the mixer becomes the output mixer for the sequencer and synthesizers.

The mixer takes the outputs of the voices and mixes these parts down into a stereo signal. Individual pan and level controls, and sometimes equalization, are included, but except on synthesizers with 'multi-timbral' effects processors, the effects provision is often implemented as several effects send and effects return controls. These are summed across the parts and sent to the specific effect, and then returned through the return level control; in some cases only the send control is implemented since the stereo outputs of the effects are mixed into the main stereo output and the effects output level control can be used to replace the effects return control. Individual send and return controls are used when an effects processor has one or more independent effects processors rather than a full 'multi-timbral' capability.

Making effective use of effects involves careful choice and frugality: using a deep flange on everything may sound impressive at first, but this initial impression soon fades:

- Reverberation is very good for providing a sense of space and size, as well as placing instruments closer to (less reverb, with short pre-delay and short reverb time) or further away (more reverb, with a longer pre-delay and longer reverb time) from the listener.
- Chorus can add extra movement to a static pad sound.
- Echo is useful for enhancing rhythmic elements and can be used to provide syncopation effects by arranging the echo time so that it either falls on or in between beats.
- Flanging is a very distinctive effect and needs to be used sparingly, although it can also be used as a chorus-type effect if the depth and feedback are kept deliberately low.
- Pitch-shifting can be useful as a subtle chorus replacement, but with feedback it can be used as a means of producing glissando or portamento effects.
- Distortion can improve the realism of electric guitar-type sounds, although this has become a cliché.

7.8 Editing

In this section, the word 'sound' will be used to indicate a broad concept so that the text does not become difficult to read. A 'sound' here refers to the waveform, or the parameters that create a sound in a synthesizer, or the samples that are used in an S&S synthesizer or sample-replay device to create a sound. Editing 'sounds' is thus a broad topic, and specific details depend on the synthesizer, sampler or computer being used, but there are some fundamental principles that can be considered.

There are two contexts in which editing of a sound can take place: live changes to parameters whilst the instrument is being played in a performance, and programming sounds in preparation for a performance. Although both can use the same controls, it is more usual for the live changes to be made using the performance controls provided by the synthesizer's wheels, pedals, etc. Programming sounds uses more detailed controls which can often be accessed from the front panel or via MIDI messages from a computer.

There are two principles of synthesizer editing which the author uses:

1. Never use a preset sound.
2. Never use the same note twice.

The avoidance of factory preset synthesizer sounds is the result of striving to sound different and to use new sounds. Some synthesists go to the extreme of clearing all the user memories in newly acquired instruments to force them to create new sounds. Of course, some factory sounds in sample-based instruments are intended as detailed reproductions of real instruments, and so grand piano samples can be used unedited, although the second principle would still apply.

Avoiding playing the same note with exactly the same timbre or intonation is an attempt to try and emulate the unpredictability and imprecise control of the real world. Professional orchestral musicians spend years learning to repeatedly produce performances with very tight control over the tone and expression, and this requires considerable skill and practice. The results are impressive, but still have minor imperfections that are a key part of the 'realism' of the sound. By emulating these slight variations in a synthesizer using editing, a mathematically precise performance can be turned into something much more human and hopefully appealing.

One example of this variation technique is sometimes called 'micro-tempo changes'. Instead of using a drum machine to provide a precise metronomic tempo, a sequencer control or conductor track is used to make small changes to the tempo, often less than one beat per minute of variation within a bar or phrase. In longer song structures, the chorus might be played several beats per minute faster or slower, or there might be a gradual acceleration or deceleration of tempo throughout a song.

7.8.1 Live parameter control

Making changes to a sound during live performance requires that the edits be easy to make and constrained to a specific parameter. This is usually set up by assigning a performance controller like a modulation wheel or foot pedal to a parameter which can be controlled using a MIDI controller message. The performance controller can then be used to change that single parameter with no concerns about changing any other parameters inadvertently. This is frequently used to provide live control over filter cut-off or a global envelope release time control when this is not available as a front panel control. Some synthesizers allow front panel controls to be set up in a similar way, although this is often as part of the sound editing facilities and is more prone to accidental editing of other parameters.

There are many different types of performance controller, and many ways of using them. It is perfectly acceptable to mix controllers that are intended for various uses, particularly when performing live, or when making live alterations to performances that have already been recorded into a sequencer. Various controllers are covered in Chapter 8.

7.8.2 Programming sounds

Programming sounds is normally achieved either from the front panel or via MIDI messages from a computer running specialized editor software. The front panel method has the advantage of being rapid and requires no additional equipment, but the display is normally small and not well suited to the display of large numbers of interrelated parameters or detailed graphical information. Computer-based editing has the advantages of a larger screen and a more sophisticated user interface, but requires a computer and extra cabling, and the speed is limited by the screen update speed and the transfer of editing commands to the synthesizer. Both methods have their strengths and weaknesses; in general, the smaller the screen the greater the advantage of the computer-based method for the beginner, whilst for the advanced user who requires quick edits, the front panel may well be preferred (Figure 7.8.1).

7.8.3 Editing techniques

There are a number of techniques that are used by sound programmers to produce useful sounds. Most of these involve an iterative process where a parameter is changed, and the result noted, and then another parameter is altered, and its effect on the sound is listened to. One essential requirement is that the programmer has a mental model of how the method of synthesis works, so that appropriate parameters can be changed. Without any clear understanding of how the synthesis technique produces sounds, then the programmer would effectively be changing parameters at random. A critical audition of the results of any of the computer-based editing programs which can produce sounds merely by randomizing the parameter values will quickly reveal that the majority of the sounds produced by this method are not usable.

The other mental model that is often required is the mapping of how the synthesis technique works to the way that the user interface is organized. This is closely related to the mapping difficulties of user controls to extracted parameters in resynthesizers. Many mental models of synthesis techniques are based on the flow of the sound from the source through the modifiers and to the audio output, whereas the user interface of many synthesizers is based on a hierarchical set of pages which are not always related to the sound flow and are sometimes very unrelated. Trying to maintain two sets of different mental models at once can be tiring and stressful, and this can be a strong contributing factor to avoid carrying out any editing.

Because S&S synthesizers normally contain a large number of preset sounds, both as programmed sounds and as the underlying samples, it is very easy to

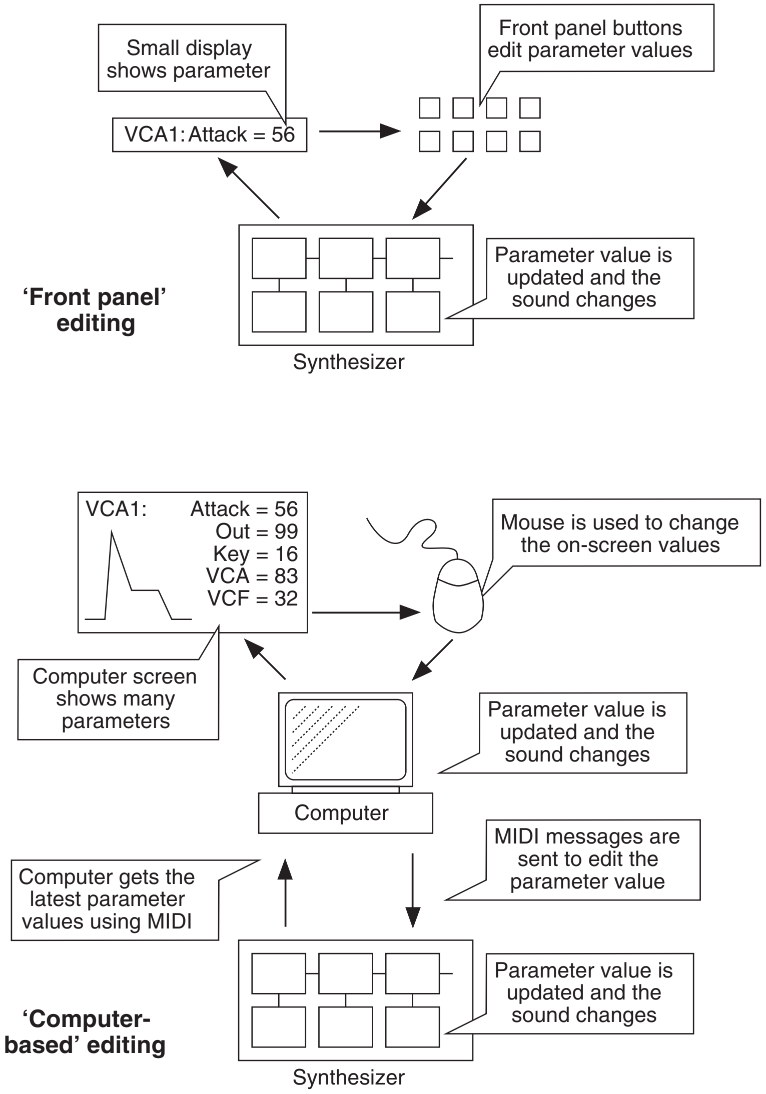

Small display shows parameter
VCA1: Attack = 56
Front panel buttons edit parameter values
Parameter value is updated and the sound changes
'Front panel' editing
Synthesizer
VCA1:
Attack = 56
Out = 99
Key = 16
VCA = 83
VCF = 32
Mouse is used to change the on-screen values
Computer screen shows many parameters
Parameter value is updated and the sound changes
Computer
MIDI messages are sent to edit the parameter value
Computer gets the latest parameter values using MIDI
Parameter value is updated and the sound changes
'Computer-based' editing
Synthesizer

FIGURE 7.8.1 'Front panel' editing involves a very direct interface between the user and the synthesizer – the front panel controls alter the parameter value which changes the sound. Computer-based editing uses the computer to display the parameter values that has been obtained from the synthesizer by using MIDI messages. Any changes to the parameters are made on the computer screen using a mouse or keyboard commands, and the changed parameters are transmitted to the synthesizer using MIDI messages.

use them as replay-only machines, rather than synthesizers. For this reason, they are often regarded as being harder to program, since avoiding the clichés which are provided by the samples and modifiers is often not very easy. Most users of S&S instruments will be more concerned with making minor changes to the sounds, so that they can be used in a specific idiom. The techniques for achieving this are somewhat different to other types of synthesizer, since they involve just changing the coarse modifier. Using an S&S instrument as a synthesizer rather than a sample-replay device involves much more detailed use of the facilities which are available. The examples which follow are thus designed to illustrate how even an instrument that is designed to replay samples can be used as a true synthesizer, with specific emphasis on making quick edits to a sound.

Sample changes

Choosing a different sample is probably the simplest technique for making changes to an S&S sound. The 'sample selection' parameter is located and then changed. This changes the raw sound source of the S&S sound, and so should radically change the overall sound. The source samples are normally arranged in groups with similar timbres, so that several piano samples will be followed by a range of string samples, which will, in turn, be followed by another group of sounds. Choosing a different sample from the same grouping will make only subtle changes to the final timbre, whereas a sample from a different group may have a very big effect on the final sound.

Most S&S synthesizers have two sample sources that are processed separately and then combined either at the filter or at the output. This means that it is simple to leave one of the two samples alone, and just change the other. For example, many piano sounds are chorused by detuning two piano samples: by replacing one of the piano samples with another sample, the result is two different instrument sounds with the same envelopes, but detuned away from each other. The user is not restricted to choose another piano sound, and in fact, there is no need to use a percussive sound at all; often string or brass samples, or even single-cycle synthesizer waveforms can make a very effective contrast to a piano sample. Just leaving the detuned piano preset and changing both piano samples for other samples can produce some unusual timbres.

Often, the mixing of two contrasting timbres can give a new 'composite' instrument which has some elements of the two component timbres, but has a new character all of its own. The relative levels of the two components may need some adjusting to maximize this effect – the volume of each is changed until the sound 'gells' into one new timbre. It is a very striking effect: two separate sounds suddenly become one. This technique is widely used in orchestral music.

If a 'familiar' timbre is mixed with an unusual or 'alien' timbre, then the extra information in the 'alien' sample can give a sort of 'halo' around the existing sound: it expands and enhances, often in an unexpected way. Adding bell-like timbres to electric piano sounds can emphasize the metallic nature of the sound and give it a new 'edge'. Often, a slight change to the envelope of the 'alien' sample section can enhance the result; reducing the sustain to zero and shortening the decay time will give a short 'blip' of the 'alien' sample at the start of the sound. Increasing the velocity sensitivity of the 'alien' part, so that it only sounds when a key is played hard, can produce a more realistic feel to the sound, since it mimics that way that harmonic structures tend to become more complex as the velocity of playing increases. Preset sounds that use velocity fading or switching to make changes to a sound depending on how hard they are played are very good for exploring a combination of 'familiar' and 'alien' samples; 'slap bass' sounds are a good place to start an exploration of this type of editing.

A common technique is to make up a sound from the attack part of one sample, and the sustain part of another. This first found commercial success in instruments such as Roland's D50 synthesizer, where the technique was somewhat confusingly called linear arithmetic (LA) synthesis. Changing the samples for the attack or sustain can produce large changes to the sound. Even swapping the two around, so that the attack comes from the sustain sample, and the sustain from the attack sample, can give some unexpected timbres.

It should be apparent from the aforementioned examples that editing S&S sounds is not the same process as with most other methods of sound synthesis. The limited set of available samples means that more sophisticated editing techniques are required if the resulting sound is to avoid being merely a replay of one of the existing samples. The use of envelopes to control separate parts of the sounds is a particularly useful method.

For hardware samplers and computer-based sample replay, the same principles of making the most of contrasting or complementary timbres also apply, but the samples which can be used are not restricted in the same way as in an S&S synthesizer. The limitations are often more to do with the storage and retrieval of samples, and the following sections in this chapter deal with naming, organization and management.

Pitch changes

Pitch can easily be overlooked when editing a timbre. Shifting the pitch of a sound up or down an octave is merely a transposition, but some very useful and interesting timbres can be found if you explore the boundaries, which means going as low in pitch or as high in pitch as possible. Shifting the pitch of samples down usually produces dark and mysterious timbres, whilst shifting the pitch upwards can produce aliasing effects and noise-like sounds.

How the samples are constructed can also be important. Short loops can give a pitched sound, and drum sounds can provide huge resources of suitably short sounds, provided that they can be looped. Longer looped sounds produce pitches when they are pitch-shifted upwards, but the complex rhythmic patterns that they produce when shifted down can form a useful background to a pad sound. If possible, fixing the pitch of this type of sample when it is transposed by a large amount, so that it does not track the keyboard, is the best idea, since then the rhythmic part stays at a constant tempo, and does not speed up as notes are played towards the right-hand end of the keyboard.

Pitch envelopes are often overlooked by inexperienced sound programmers. Some brass sounds will have a quick rise to the note, but the only other commonly encountered use is a much more exaggerated pitch chirp on a lead sound, when it is often labeled as a 'funny' sound. But there are quite a few more creative opportunities with pitch envelopes. Subtle slow envelopes on vocal sounds can give a very realistic sound, especially if this is mixed with a more conventional 'choir' sample. Putting a pitch envelope onto only one of the two parts of a sound can give a very useful 'detune'-type chorus effect at the start of a note, and which is then followed by the two parts of the note coming into tune for the sustain portion of the envelope. By tuning the envelope range, so that it covers one or more octaves, it is possible to create sounds which have octave doubling for only part of the envelope.

LFO modulation of pitch is often restricted to provide vibrato. But a slow LFO applied to one part of a sound will then produce a cyclic 'detuned in-tune detuned in-tune' effect, which can be very effective on string sounds, particularly if this is layered with a more 'realistic' string timbre. Using a sawtooth waveshape on the LFO is one of the elements of many string-like synthetic timbres.

Envelopes

Envelopes tend to have the reputation of being suitable only for detailed editing. Quick edits are restricted to merely changing the attack or release times. But many S&S instruments provide more than one envelope, and it is often possible to change the envelope which is being used by a part very easily. The quick edit advice is thus to use the wrong envelopes. Shortening or lengthening attacks, decays and releases can turn sustained sounds into percussive sounds, and vice versa. String timbres sound very different if they have piano-type envelopes.

For an S&S instrument with two separate parts to the sound, contrasting envelopes can be used for the two parts. This opens up possibilities like using the sustain part of a string sound to provide the attack, and the attack sample (looped) to provide the sustain sound. Since these sounds can be layered with presets, long and slow envelopes can be used to provide evolving sounds which are much more interesting to listen than static pads. Using envelopes to cross-fade from one sound to another is usually used just to paste together an attack and sustain samples, but this can also be used with contrasting sounds, and altering the cross-fading can provide more dynamic changes. Sounds which vanish in the middle and then reappear in a different form may appear quirky, but they are very useful for adding a bit of interest into washes and introductions. They can also provide some interesting song or melody ideas when they are used with a sequencer, since the timing will probably not coincide with the sequencer playback, and this can create complex poly-rhythms.

Filters

The filter controls probably take the second place after the sample choice, as the main focus of quick edits. The filter frequency is probably the most powerful of all the modifier control for determining the timbre of the sound. Filtering should be used to try and remove or enhance the sample 'fingerprint'. Removing parts of the spectrum of the sample may well disguise it, whilst emphasizing other parts of the spectrum may also help to lose some of the distinctiveness. The more complex the filter, the more control that it will give over the harmonic content of the sound, so simple low-pass filters are not ideal. Low-pass, high-pass and band-pass filters are a good starting point, whilst notch and comb filters can be very useful, and any other shapes are a bonus.

As a general rule, the resonance control acts as a selectivity control – the higher the resonance, the more changes the filter will make to the sound. This is especially noticeable with a low-pass filter which has a resonance control which gradually makes it more and more like a band-pass filter: the sound becomes more interesting and synthetic sounding as the resonance approaches the point where the filter is about to self-oscillate.

The overuse of low-pass filters can tend to give a very thick and mushy sounding bottom end to sounds, whilst high-pass filters tend to 'thin out' the sound. Filters should always be used in combination with the other elements of the sound, rather than hoping that they can provide a stand-alone sound. One exception to this is the 'synth brass' filter sweep sound, where an attack decay sustain release (ADSR) envelope is used to sweep the filter cut-off frequency and process a sawtooth cycle or brass sample. The resulting sound is one of the strongest synthesizer clichés of the mid-to-late 1970s!

Summary

Although these modifications to sounds have been described in the context of S&S synthesizers, almost all of the techniques also apply to samplers. Many of the techniques also apply to other methods of synthesis, particularly subtractive, hybrid and FM synthesis (Figure 7.8.2).

7.8.4 Using factory sounds

Commercial synthesizers are typically supplied with a set of 'factory presets'. These are sounds whose purpose is to demonstrate the capabilities of the instrument to a potential purchaser. They are often well suited to this purpose, but are rarely suited to subsequent use when combined with other instrument sounds in a mix.

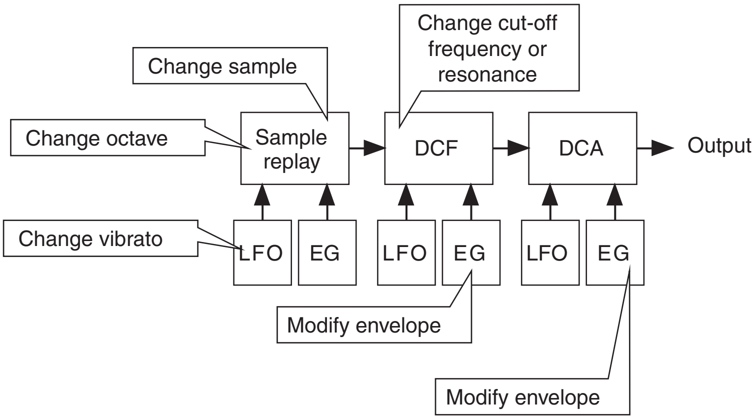


FIGURE 7.8.2
Some quick edits on an S&S synthesizer.

Editing factory sounds is a good way of learning how to edit a synthesizer, and can also be used to produce more widely applicable sounds.

The basic techniques for making factory sounds useful are relatively straightforward: often a combination of reducing the amount of effects which have been applied and changing the envelope and modulation. Here are a few typical symptoms and the required correction:

- *Buried in reverb*: Reduce the 'wet'ness of the reverb mix or the depth of reverb effect.
- *Drowned in chorus and detune*: Reduce the detuning of the oscillators, and reduce the depth of LFO modulation in the effects.
- *Excessively wide stereo image*: Adjust the pan of the components of the sound so that they are not panned hard left or right.
- *Auto-pan*: Sounds which move around in the stereo sound image excessively are interesting once. Reduce the depth of LFO or envelope modulation of the panning or stereo position control.
- *Slow rise and fall*: Very languorous strings and lots of chorus can sound very impressive on pad sounds, but the attack and release times are often too long. An expression or volume pedal is a better and more controllable way of getting the same sort of crescendo effect.
- *Polyphony stealing*: Long release times can make block chords sound impressive on their own, but can swamp other instruments in a mix. Reducing the release time and increasing the velocity sensitivity of some of the components will give a much more expressive sound which does not use up excessive amounts of polyphony, and which will work better when part of an accompaniment.
- *Filter resonance*: Filter oscillation can be overused, and the very distinctive accentuation of harmonics at high-resonance settings can also become a cliché. Lower settings of resonance, particularly when the resonance is controlled by keyboard note position, can be a much more powerful, controllable (and imitative) effect when used in performance.
- *Filter modulation*: Cyclic sample and hold modulation of filter cut-off frequency at about 4 Hz has become an overused cliché. Much more useful and interesting is a much slower modulation, or one which is related to keyboard velocity.
- *Echo and timed samples*: Repeated sample loops with widely panned stereo echoes can produce some very impressive demonstration sounds. But every user of the instrument will also be thinking exactly the same thing, and using such an immediately recognizable sound is not very imaginative (at best). Any use of these highly distinctive sample loops is probably best

avoided entirely, or else they need to be altered so drastically that they cannot be easily recognized: large pitch changes are one technique.

- *Hyper-reality*: String buzzes, harpsichord jack-clicks and other ephemera of real-world instruments tend to be over-emphasized in factory sounds. But they quickly pale when used with other instruments or for solo work of more than a few bars. Increasing the velocity sensitivity or using keyboard note position to reduce the ubiquity can make these sounds much more acceptable.
- *Stacks and 1-note chords*: Fifths, octaves and rich, thick sounds with lots of harmonics can be impressive in a demo. But parallel chords can quickly lose their appeal with repetition, and they restrict the possibilities for harmonization, cadences, suspensions, etc. Simpler, purer sounds, which can be used to make up chords when required, are much more useful in wider contexts.

It should be noted that the results of turning factory preset sounds into sounds that are more useful in an ensemble can be compared with samples of real-world instruments, where the use of the sounds together may be more familiar. Listening to arrangements of music performed using real instruments can provide a good starting point, but listening to music performed using samples can be even more interesting, particularly if the intention is to emulate reality.

7.8.5 Managing edited sounds

Although producing, controlling, manipulating and arranging sounds are the 'foreground' activities of the synthesist, a number of associated background management functions need to take place for these performance-related activities to happen. Since sounds and music are the major resources for a synthesist or a sample-user, they need to be cataloged, stored and made available for subsequent retrieval.

Although a large number of synthesizers and samplers now offer digital storage of their operating parameters, this is not always possible for analogue instruments or for some effects processing equipment, nor for custom-built apparatus. Notes on these items may well be stored on paper in notebooks or binders, but the principles of resource management remain the same.

Computers may seem to be an easy way of managing the large numbers of settings, samples, sounds and other parameters that are used to specify and perform a piece of music, but it is very easy to forget to 'look after' computers – the software they contain, and the data that they use. Because it is increasingly possible to have all of the sound-creating capability on a computer, that computer becomes a single point of failure, rather than being spread across a number of synthesizers, samplers, drum machines, effects units, sequencers, hard drives and manuscript paper notes.

Sorting

Part of the process of learning how to use a synthesizer is the production of sounds. Some synthesists prefer to start their investigations from the preset sounds which are provided, whilst others ignore or remove any presets and work from the basic 'initialized' sound, which is normally a simple sine wave.

The programming of sounds can be approached in many ways. A methodical exploration of the available parameters may suit one synthesist, whilst an iterative process based on making changes to the parameters more or less at random and observing their effects may work for another person. In this learning phase, a large number of sounds are typically produced, and these can either be used as finished sounds or provide the raw material for further programming. After some programming, the synthesist begins to develop a mental 'model' of the operation of the synthesizer, and this will be confirmed by further programming: followed by checking of the expected results against the reality. Once this stage is complete, the synthesist should feel confident of their ability to produce almost any required sound.

Regardless of their origin, the sounds which are produced by the programmer need to be filed in a way that allows them to be reused at a later date. Few synthesists have the ability to rapidly produce a specific sound when required: instead, they use these pre-prepared sounds or edit them in context. In order to file the sounds, they need to be sorted and categorized, and a number of systems can be used to provide this function. Perhaps the most useful technique in cataloging the sounds and music is to split the pool into categories. There are many different methods of naming sounds: ranging from ones which describe them in generalized terms such as slow, smooth, bright or dark, whilst others use references to conventional instruments: brassy, piano-like, flute-like or bell-like. Some composers have used references to the moods or emotions which the timbres suggest. Most of the commercial software packages use a mixture of generalized terms and references to specific instruments.

Whilst categories are useful for grouping sounds and retrieving them at a later date, they can be unwieldy for everyday use. Simple names for sounds are a much better way of ensuring that they will be remembered easily in day-to-day use. Most synthesizers use some sort of 'bank and number' system to numbering their sounds: often based on banks containing 8- or 16-numbered sounds. The banks can be either numbered or assigned letters; 4–12 and B5 are both typical examples. Numbers of these forms are not readily associated with sounds, and so the naming of sounds is very important. There are a number of methods used for naming sounds. The simplest is to attempt to describe each sound as accurately as possible given the limitations of the number of character which can be used. Since this is normally between 8 and 16 characters, names 'chorused electric piano with bell-like attack sound' are need to be abbreviated. One useful technique is to divide the names into two parts. The first two or three characters are used to indicate the rough grouping of sound: piano, string, brass, etc. The remaining characters can then be used to provide more detailed description, albeit in an abbreviated form. This method produces names of the form 'EP:ChrBelAtt' where the descriptive part is split into meaningful parts by using capital letters instead of spaces.

More egotistical methods of naming sounds exist. These sometimes use any additional available characters to provide a graphical name: '-[V]-' being a typical example. Other programmers use in-jokes or clever plays on words: one rather obvious example being 'sdrawkcab' for a sound which gives the impression of being played backwards. Foreign language words are often popular because they have a mystique due to their unfamiliarity, or they may further plays on words: 'bête noire' (literally French for 'black beast', but English for a weakness) and 'koko wa?' (Japanese for 'What is this?') are two examples. For this type of name, the more memorable a name is the more effective it is.

Finding

Once filed away with a suitable set of category words, name and perhaps grouping, the sound is ready to be used. Finding the right sound involves using the reverse process of naming: deciding on the type of sound to be used, and then choosing category words and perhaps groupings so that a 'short-list' of potential candidate sounds can be produced. Auditioning of these sounds can then begin, with the user testing each sound in context with other sounds which will be used. Once the sound set has been chosen, then the sounds will probably be stored as a single entity temporarily whilst the project is in progress.

Librarians

Software that carries out the naming, categorization, grouping, sorting and finding functions is called librarian software. A number of programs exist for this purpose, and they are often associated with programs which can be used to edit sounds, called editors. Two types of these programs exist. Generic or 'universal' versions can be used with almost any synthesizer or related equipment since their behavior can be programmed and new equipment can be added merely by loading in new programs. Specific versions are designed to be used with only one synthesizer, although more limited they are often lower in cost than the generic versions.

Storage

Storing sounds can be achieved in several ways. Most modern hardware synthesizers have internal flash memory storage, often with additional memory card-based or, in older examples, floppy disk-based storage. Older synthesizers and samplers use volatile RAM storage which has to be saved to a more permanent form of storage because the information disappears when the equipment is powered down. Many computers use the same volatile memory, but have standby modes and can copy the RAM to hard disk, thus appearing to have reliable storage. System exclusive MIDI messages can be used to send and receive sound data, and this method is used by librarian software. Librarians can then store the sound data on floppy, hard or optical disk storage.

The way that information is organized when being stored can either aid or hinder subsequent retrieval. Logical organization of projects and their constituent files, particularly taking care to use folders or directories to keep the number of files small, can be a powerful way to ensure that you can find things when you need to. Most storage systems provide date information, although many people prefer to create folders or directories based on years or months, and to store projects in further folders inside them. An alternative for projects that may be long-lived is to use project folders or directories as the highest folder or directory structure, and to put date-based folders or directories inside them. The best system is the one that you understand and will use repeatably and consistently.

When the storage is a mixture of hardware and software, as with a hardware dongle, CD or key for a piece of software, then the hardware needs to be stored somewhere safe, and purchase details should be stored somewhere else so that the hardware can be replaced should it be lost. CDs or DVDs containing samples which are then transferred to hard drives for everyday use should be stored carefully, since the working copies may become corrupted or lost if a hard drive fails. It may be worth deliberately storing hardware such as dongles or sample CDs in various locations to avoid any problem with a single storage location being compromised.

In general, the storage of important sound data should be in several different locations and on different media, so that the chances for losing sounds are minimized whilst the possibilities for recovering sounds are maximized. This applies particularly to computers, where regular backups are important to protect against the many ways that computers can fail. The author has experienced several hard disk failures, and in each case, the failure was sudden, with no warning slowdown or strange noises, and the data loss was immediate and total.

7.9 Sequencing

Using the techniques described in this chapter may require some translation in order to be used with specific sequencers. Sequencers vary quite a lot in their detail, particularly: their depth of support for editing and processing MIDI messages; their selection of notes according to functional criteria like 'highest note played at this time'; and the amount of automation information that they can store (like setting presets on external effects and mixers, sending MIDI sysex, etc.). Some solutions include the following.

Stacking notes can be achieved by copying a track and changing the sound being played on the copy track, but it may also be possible to double the MIDI note messages and assign them to two different channels. Do not forget to try putting the sounds at different pan positions, or even adjusting the timing of the notes slightly, which can be very good for enhancing the impression of ensemble sounds.

Selecting notes for hocketing may require careful manual selection if the sequencer only provides simple selection tools. It may be worth lobbying the manufacturer to encourage more sophisticated note and event selection if you intend to do lots of hocketing and it is not supported in your sequencer of choice. An alternative solution is to purchase a sequencer that does allow advanced selections and to use this to separate the track into parts and then to use MIDI file export and import to move them to your preferred sequencer.

Record the MIDI messages to set up the effects and other outboard devices and save as a MIDI File, and play this first to set up the outboard equipment. You may also be able to create the required MIDI messages as events in a spare track. Unfortunately, some sequencers do not allow editing of MIDI messages, in which case a second sequencer that does allow MIDI message editing may be required.

Sequencers can be used to produce delayed envelopes by moving note events back in time so that the undelayed notes produce the attack sound, and the delayed notes produce the decay and sustain sound. For stacks, then moving notes with slow attacks ahead in time can make stacks sound more coherent.

7.10 Recording

Workstations can be used as synthesizers or samplers: just because there is a sequencer inside it does not mean that it has to be used. A computer sequencer can drive several devices without worrying about synchronization.

Recording takes time, and preparation can save lots of time. Being organized can also prevent mistakes, and prevent forgetting that flash memory stick with the vital files on it.

7.11 Performing

Groove boxes, drum machines and workstations synced together can be very good for experimentation, but when playing live, less is more. Simple hardware or software setups will be more reliable, and easier to get going again if things go wrong.

Fine details are often not needed live. Factory presets on one convenient instrument may be able to replace several other rare or fragile instruments, particularly if you add some effects.

Know what the names of the sounds you use are.

Know how to get back up and running if there is a power failure.

Know the running order of the set.

7.12 Questions

1. What types of control over a performance does synthesis provide?
2. Outline three approaches to stacking.
3. Describe three methods of layering sounds.
4. What is hocketing?
5. What is the difference between multi-timbrality and polyphony?

6. Outline a simple note assignment strategy. How could this be improved?
7. Describe the major types of audio effects and their uses.
8. Describe the differences between an effects unit intended for use in a monophonic synthesizer and one designed for use in a multi-timbral polyphonic synthesizer.
9. Outline some of the ways of controlling a synthesizer during performance.
10. Which editing techniques would be common to an S&S synthesizer and a subtractive analogue synthesizer?

7.13 Timeline

1500s	Barrel Organ	The barrel organ. Pipe organ driven by barrel covered with metal spikes.	The fore-runner of the synthesizer, sequencer, and expander module!
1700s	Orchestrion	Orchestrions made in Germany. Complex combinations of barrel organs, reeds and percussion devices. Used for imitating orchestras.	
1804	Leonard Maelzel	Leonard Maelzel invented the Panharmonicon, another mechanical orchestral imitator.	
1840s	Jacquard	Jacquard card driven organs replaced barrel organs.	
1870s	Gavioli	Fairground organs from Gavioli began to use real instruments to provide percussion sounds.	
1877	Thomas Edison	Thomas Alva Edison invented the cylinder audio recorder – the ‘Phonograph’. Playing time was a couple of minutes!	Cylinder was brass with a tin foil surface, replaced with metal cylinder coated with wax for commercial release.
1887	Torakusu Yamaha	Torakusu Yamaha built his first organ.	
1896–1906	Thaddeus Cahill	Invented the Telharmonium, which used electromagnetic principles to create tones.	Telephony.
1898	Valdemar Poulsen	Invented the Telegraphone, which recorded telephone audio onto iron piano wire (also known as the Dynamophone).	30 seconds recording time, and poor audio quality.
1915	E.C.Wente	Produced the first ‘Condenser’ microphone using a metal-plated insulating diaphragm over a metal plate.	Now known as a ‘Capacitor’ microphone.

(*Continued*)

Timeline (*Continued*)

1915	Lee de Forest	The first Valve-based oscillator.	
1920	Lev Theremin	The Theremin – patented in 1928 in the US. Originally called the ‘Aetherophone’.	Based on interfering radio waves.
1929	Couplet & Givelet	Four voices, paper tape driven ‘Automatically Operating Oscillation Type.	Control was provided for pitch, amplitude, modulation, articulation and timbre.
1930s	Baldwin, Welte, Kimball & others	Opto-electric organ tone generators.	
1930	Friedrich Trautwein	Invented the Trautonium – an early electronic instrument.	Wire pressed onto metal rail. Original was monophonic. Later duophonic.
1930s	Ondes	Ondioline – an early synthesizer.	Used a relaxation oscillator as a sound source.
1933	Stelzhammer	Electrical instrument using electro-magnets to produce a various timbres.	
1934	John Compton	UK patent for rotating loudspeaker.	
1934	Laurens Hammond	Hammond ‘Tone Wheel’ Organ used rotating iron gears and electro-magnetic pickups.	Additive sine waves.
1937	Tape recorder	Magnetophon magnetic tape recorder developed in Germany.	The first true tape recorder.
1939	Hammond	Hammond Novachord – first fully electronic organ.	Used ‘master oscillator plus divider’ technology to produce notes.
1939	Hammond	Hammond Solovox – monophonic ‘synthesizer’.	British Patent 541911, US Patent 209920.
1951	Hammond	Melochord.	
1954	Milton Babbitt, H.F.Olsen & H.Belar	RCA Music Synthesizer mark I.	Only monophonic.
1955	E.L.Kent	Kent Music Box in Chicago. Inspired RCA mark II synthesizer.	
1957	RCA	RCA Music Synthesizer mark II.	Used punched paper tape to provide automation.
1958	Charlie Watkins	Charlie Watkins produced the Copicat tape echo device.	
1959	Yamaha	First ‘Electone’ organ.	

(*Continued*)

Timeline (*Continued*)

196?	Clavioline	Clavioline.	British Patent 653340 & 643846.
1960s	Clavioline	Clavioline.	British Patent 653340 & 643846.
1963	Don Buchla	Simple VCO-, VCF- and VCA-based modular synthesizer: 'The Black Box'.	Not well publicized.
1965	Robert Moog	First Moog Synthesizer was hand-built.	Only limited interest at first.
1968	Ikutaro Kakehashi	First stand-alone drum machine, the 'Rhythm Ace FR-1'.	Designed by the future boss of Roland.
1969	Peter Zinovief?	EMS (Electronic Music Studios) produced the VCS3, the UK's first affordable synthesizer.	The unmodified VCS3 is notable for its tuning instability.
1969	Robert Moog	Minimoog is launched. Simple, compact monophonic synthesizer intended for live performance use.	Hugely successful, although the learning curve was very steep for many musicians.
1970	ARP Instruments	ARP 2600 'Blue Meanie' modular-in-a-box released.	
1970	ARP Instruments, Alan Richard Pearlman	ARP 2500. Very large modular studio synthesizer.	Used slider switches – a good idea, but suffered from crosstalk problems.
1971	ARP Instruments	The 2600, a performance-oriented modular monosynth in a distinctive wedge-shaped box.	The 2600 got modulars out of the studio and was hugely influential.
1972	Roland	TR33, 55 and 77 preset drum machines launched.	
1973	Oberheim	First digital sequencer.	The first of many.
1975	Moog	Polymoog is released.	More like a 'master oscillator and divider' organ with added monophonic synthesizer.
1975	New England Digital	Synclavier is launched. First 'portable' all-digital synthesizer.	Expensive and bulky.
1976	Lol Creme & Kevin Godley	The Gizmotron – a mechanical 'infinite sustain' device for guitars.	A variation on the 'bowing with steel rods' technique.
1977	ARP	ARP Avatar Guitar Synthesizer.	Monophonic synthesizer with Hex pickup (for 'Hex Fuzz').
1977	Roland	GR-500 Guitar Synthesizer.	Hex pickup and string drivers gave 'infinite' sustain.
1977	Roland	MC8 Microcomposer launched: the first 'computer music composer' – essentially a sophisticated digital sequencer.	Cassette storage – this was 1977!

(*Continued*)

Timeline (*Continued*)

1978	Electronic Dream Plant	Wasp Synthesizer launched. Monophonic, all-plastic casing, very low-cost, touch keyboard – but it sounded much more expensive.	Designed by Chris Huggett & Adrian Wagner.
1978	Roland	Roland launch the CR-78, the world's first programmable rhythm machine.	
1978	Sequential Circuits	Sequential Circuits Prophet 5 synthesizer – essentially five Minimoog-type synthesizers in a box.	A runaway best seller.
1978	Yamaha	Yamaha CS series of synthesizers (50, 60 and 80), the first mass-produced successful polyphonic synthesizers.	Korg, Oberheim and others also produced polyphonic synthesizers at about the same time.
1979	Fairlight	Fairlight CMI announced. Sophisticated sampler and synthesizer.	The start of the dominance of computers in popular music.
1979	Roland	Boss 'Dr. Rhythm' programmable drum machine.	
1979	Roland	Roland Space-Echo launched. Used long tape loop and has built in spring line reverb and chorus.	A classic device, used as the basis of several specialist guitar performance techniques (e.g. Robert Fripp).
1980	Electronic Dream Plant	Spider Sequencer for Wasp Synthesizer. One of the first low-cost digital sequencers.	252-note memory, and used the Wasp DIN plug interface.
1980	E-mu	Emulator, first dedicated sampler.	
1980	Roland	GR-300 Guitar Synthesizer.	
1980	Roland	Jupiter-8 polyphonic synthesizer.	8-note polyphonic, programmable poly-synth.
1980	Roland	Roland TR-808 launched. Classic analogue drum machine.	
1980	Roland	Jupiter-8 polyphonic synthesizer.	8-note polyphonic, programmable poly-synth.
1981	Casio	VL-Tone. Rhythm, drums, chords and monophonic synthesizer in a low-cost 'overgrown calculator'.	Electronic music for the masses!
1981	Moog	Robert Moog is presented with the last Minimoog at NAMM in Chicago.	The end of an era.
1981	Roger Linn	The Linn LM-1: world's first programmable digital drum machine.	Replayed samples held in EPROMs.
1981	Roland	Roland Jupiter-8. Analogue 8-note polyphonic synthesizer.	

(*Continued*)

Timeline (*Continued*)

1982	Moog	Memorymoog – 6-note polyphonic synthesizer with 100 user memories.	Cassette storage! Six Minimoogs in a box!
1982	PPG	Wave 2.2, polyphonic hybrid synthesizer, was launched.	German hybrid of digital wavetables with analogue filtering .
1982	Roland	Roland SH-101, a monophonic synthesizer with an add-on hand-grip performance-oriented pitch bend and modulation controller.	Notable for its range of color finishes: red, blue and gray.
1983	Oxford Synthesiser Company	Chris Huggett launched the Oscar, a sophisticated programmable monophonic synthesizer.	One of the few monosynths to have MIDI as standard.
1983	Roland	Roland launched the TR-909, the first MIDI-equipped drum machine.	
1983	Sequential Circuit	Sequential Circuit's Prophet 600 was the first synthesizer to implement MIDI.	Prophet 600 is marred by awful membrane switch keypad.
1983	Yamaha	'Clavinova' electronic piano launched.	
1983	Yamaha	MSX Music Computer: CX-5 launched.	The MSX standard failed to make any real impression in a market already full of 8-bit microprocessors.
1983	Yamaha	Yamaha DX7 is released. First all-digital synthesizer to enjoy huge commercial success. Based on FM synthesis work of John Chowning.	First public test of MIDI was Prophet 600 connected to DX7 at the NAMM show – and it worked (partially!).
1984	Kurzweil	Kurzweil 250 provided 2 Megabytes of ROM sample playback.	
1984	Sequential	Sequential launched the Max, an early attempt at mixing home computers and synthesizers.	A complete failure – too early for the market.
1984	Sequential Circuits	SixTrak, a multi-timbral synthesizer with a simple sequencer.	The first 'workstation'?
1984	SynthAxe	MIDI-based guitar controller with separate strings, trigger switches and fretboard switches.	Nice design, but very expensive.
1985	Akai	The S612 was the first affordable rack-mount sampler, and the first in Akai's range.	12-bit, Quick-Disk storage and only 6-note polyphonic.
1985	Ensoniq	Introduced the 'Mirage', an affordable 8-bit sample recording and replay instrument.	

(*Continued*)

Timeline (*Continued*)

1985	Korg	Korg announced the DDM-110, the first low-cost digital drum machine.	The beginning of a large number of digital drum machines…
1985	Yamaha	DX100 (4 operator mini-key) FM synthesizer was launched.	
1985	Yamaha	DX21 (4 operator full size keyboard) FM synthesizer was launched.	
1986	Sequential	Sequential launched the Prophet VS, a 'Vector' synth which used a joystick to mix sounds in real-time.	One of the last Sequential products before the demise of the company.
1986	Steinberg	Steinberg's Pro 16 software for the Commodore C64.	The start of the explosion of MIDI-based music software.
1986	Yamaha	Clavinova CLP series electronic pianos was launched.	CLP pianos were pianos – the CVP series add on auto-accompaniment features.
1986	Yamaha	DX7II was revised DX7 (a mark II).	Optional floppy disk drive.
1986	Yamaha	Electone HX series organ was launched.	Mixture of FM and AWM (sampling).
1987	Casio	Introduced the Casio CZ-101, probably the first low-cost multi-timbral digital synthesizer.	Used Phase Distortion, a variant of waveshaping.
1987	DAT	DAT (Digital Audio Tape) was launched. The first digital audio recording system intended for domestic use.	Worries over piracy severely prevented its mass marketing.
1987	Kawai	K5 digital additive synthesizer launched.	Powerful and not overly complex.
1987	Roland	MT-32 brought multi-timbral S&S synthesis in a module.	The start of the 'keyboard' and 'module' duality.
1987	Roland	Roland D-50 combined sample technology with synthesis in a low-cost mass-produced instrument.	S&S synthesis (Sample & Synthesis).
1987	Yamaha	Yamaha DX7II centennial model – second generation DX7, but with extended keyboard (88 notes) and gold plating everywhere.	Limited edition.
1988	Korg	Korg M1 workstation was launched. Used digital S&S techniques with a excellent set of ROM sounds.	A runaway best seller. Because it put synthesis, sequencing and mixing/effects into one device. Notably, the filter has no resonance.

(*Continued*)

Timeline (*Continued*)

1989	Akai	XR10 Drum Machine was launched.	A digital drum machine using sampled drum sounds.
1989	Breakaway	The Breakaway Vocalizer 1000 was a pitch-to-MIDI device that translated singing into MIDI messages and sounds via its on-board sampled sounds.	Somewhat marred by a disastrous live demonstration on the BBC's 'Tomorrow's World' **program**.
1989	Waldorf	MicroWave, a digital/analogue hybrid based on wavetable synthesis.	Effectively a PPG Wave 2.3 brought up to date.
1990	Korg	Wavestation was launched. An updated 'Vector' synth, using S&S, wavecycle and wavetable techniques.	Powerful and under-rated.
1990	Technos	French-Canadian company Technos announced the Axcel – first resynthesizer.	There was no follow-up to the announcement.
1990	Yamaha	SY77, a digital FM/AWM hybrid synthesizer, mixed FM and sampling technology.	Followed in 1991 by the larger and more powerful SY99.
1991	Roland	JD-800, a polyphonic digital S&S synthesizer.	Notable for its front panel – controls for everything!
1992	Kurzweil	The K2000 was launched. A complex S&S instrument, which mixed sampling technology with powerful synthesis capability.	
1993	E-mu	Morpheus synthesizer module was launched. Used real-time interpolating filter morphs to change sounds.	Sophisticated DSP.
1995	Yamaha	VL1, world's first Physical Modeling instrument was launched.	Duophonic, and very expensive.
1996	SMF	Standard MIDI File specification announced by MIDI Manufacturer's Association.	
1997	DVD	First DVD video players were released. DVD Audio standard did not appear until 1999.	
1997	Korg	Z1 polyphonic physical modeling synthesizer	
1998	Ensoniq	Fizmo.	Advanced wavetable synth.
1998	Waldorf	MicroWave XT.	The MicroWave XT was an analogue wavetable and FM polysynth with arpeggiator and FX.

(*Continued*)

Timeline (*Continued*)

1998	Yamaha	DJ-X, a dance performance keyboard disguised as a ‘fun’ keyboard.	Followed by a keyboardless DJ version, the DJXIIB.
1999	General MIDI 2 (GM2)	Enhanced superset of GM, now called GM1.	Backwards compatible. Increases the tightness of the specification.
1999	MP3	First MP3 audio players for computers appeared.	Internet music downloading begins…
2000	Yamaha	mLAN, a FireWire-based, single cable for digital audio and MIDI.	Slow acceptance for a brilliant concept.
2001	General MIDI Lite	Reduced GM1 specification designed for use in devices such as mobile phones.	
2001	Korg	Karma, a combination of a synthesizer with a powerful set of algorithmic time and timbre processing.	
2002	Hartmann Music	Neuron Resynthesizer.	Arguably the first commercially produced resynthesizer.
2002	SP-MIDI	Scalable Polyphony was layered note priority approach to making GM files to drive polyphonic ring tones in mobile phones.	Some mobile phones were GM1 compatible.
2003	Yamaha	Vocaloid, mass-market singing synthesis software.	Backing vocals would never be the same again!

This page intentionally left blank

CHAPTER 8

Controllers

CONTENTS

Controllers

Environment

Musical performance is a combination of the instrument and a player. The interaction between the two produces the music, and the interfacing between the player and the instrument involves the control of the instrument. There are many ways by which this control can be achieved. In a conventional instrument, the control interface is often predetermined by the instrument itself. For example, a guitar has six strings, a fretboard, bridge and so on and the player can pluck, strum or tap the strings, which may be open or fretted. But for a synthesizer, there is no fixed set of controllers because a synthesizer is not constrained by any physical limitations. The control is thus determined by the flexibility of the synthesis technique and its physical implementation.

The organ-type keyboard that is used as the major controller on many synthesizers seems to have been chosen for all the wrong reasons. Whilst early synthesizers were monophonic, the keyboard is naturally polyphonic since it is all too easy to play more than one key at the same time. The only opportunities for expressive control using the keyboard are attack velocity when the note is initially pressed and after-touch or key pressure once it has been pressed, which means it does not match the continuous and the diverse expression capabilities of a synthesizer. But the keyboard was easy to wire up to produce simple control voltages and triggers for testing the first synthesizers, and this is probably why it was adopted. Since then, the keyboard has been continued to be used as the major control interface to the synthesizer.

A synthesizer with just a keyboard is very limited in control terms. Most synthesizers add a number of additional controllers to augment the keyboard's limited capability. The pitch-bend wheel is used to provide changes in pitch that is similar to those produced by pulling the strings on a guitar; although in the case of a keyboard, the use of the pitch-bend wheel requires one hand to be removed from the keyboard, usually the left hand, since the convention is that the performance controls on a synthesizer are always on

To simplify the text, the term 'synthesizer' and 'synthesist' have been used, but these should be taken as referring to both synthesizer and sampler, and the users of them.

The Theremin is an example of an early synthesizer that is naturally monophonic, with just pitch and volume control in the performer's hands.

the left-hand side. The modulation wheel is used to provide control over additional performance parameters such as vibrato, tremolo or filter cut-off and again requires the left hand for operation. On a violin modulation, effects like vibrato are produced by moving the fretting position of the strings by the fingers. Modulation effects can be controlled by using the after-touch or key pressure on a keyboard, but this limits their use to when a key can be pressed down and sufficient pressure applied to activate the pressure sensing circuitry; vibrato can thus be applied only after the note has started, and cannot continue uninterrupted when the note changes, nor can it be used when a rapid run of notes is required. Foot pedals and breath controllers are just some of the additional controllers that are employed to try and provide more flexible and expressive control over sounds that are merely pitched and triggered by the keyboard.

Many of these control problems also apply to the use of piano-style keyboards to synthesizers, although this does improve the interaction between the player and the synthesizer if a piano type of sound is being controlled. For almost all other types of sound, the piano keyboard is less suitable.

In contrast, a synthesizer controlled by a stringed instrument controller requires a much more sophisticated interface to be able to extract the pitch, trigger and performance information, but the resulting control is much neater and more intuitive. Pitch bend can be applied using the same hand that is fretting the strings by pulling the strings away from their rest position, and vibrato can be combined with pitch bend using the same fingers; in fact, each finger can be used to apply different pitch bend or vibrato, if necessary, which is difficult for most keyboard-controlled synthesizers where the modulation is global to all notes being held down.

The complexity of the user interface for a wind instrument as the controller for a synthesizer is simpler because only a single monophonic pitch needs to be determined, but this ideally suits a monophonic synthesizer. The control over pitch is now spread between the hands and the mouth, with modulation control coming from the mouth, tongue, teeth and lips. The volume of the sound produced by the synthesizer can be controlled by breath pressure and can be continuously changed during the course of a note: something that is very difficult using a keyboard-controlled synthesizer without using a foot pedal or wheel controller (or a breath controller!).

8.1 Controller and expander

Separating the controller from the sound generating parts of a synthesizer produces two separate devices. The sound producing part of the synthesizer is the expander module: a 'keyboard-less' version of the synthesizer, where the term 'synthesizer' almost always implies the inclusion of a keyboard. Because of this ubiquity of the organ-type keyboard as the 'master' keyboard controller for several expander modules, any other form of controller is normally referred to as an 'alternative controller': wind instruments, guitars and drums are three of the commoner forms.

Regardless of the type of controller, there are a number of elements that are used in all of them. Each needs to provide control signals or voltages that produce some or all of the following information:

- pitch of a note event,
- start and end of a note event,
- dynamics of the note (volume),
- changes in pitch,
- modulation changes,
- sustain,
- additional expression controls.

Some controllers combine several of these into one composite controller, whilst others provide just one. A master keyboard may provide nothing other than the basic pitch and dynamics information, whilst a more sophisticated version may provide pitch bend, modulation and sustain information as well. A guitar controller will provide up to six separate sets of information (one set per string) whilst a keyboard will normally provide either monophonic or polyphonic information. Wind controllers attempt to interface blown instruments to synthesizers and are usually monophonic. The form of a controller may change depending on the application. The pitch-bend wheel on a keyboard controller may function identically to the 'tremolo arm' on a guitar controller, although the physical realization is very different.

No one controller can provide complete control over all the available performance parameters – each has specific advantages and disadvantages. A guitar controller gives detailed control over the pitch, level and modulation of up to six separate 'strings', but is limited in its expression capabilities for sustained sounds, and control over the release segment of envelopes. In contrast, a keyboard provides pitch and dynamics information, and by using after-touch or key pressure, it can control modulation and expression whilst the notes are sustained, and the sustain can be modified by using a sustain pedal. But keyboards do not provide control over pitch bend without using an additional controller (or using after-touch, in which case the modulation control ability is lost) and the control over the level of the sound is limited to the attack dynamics of the key velocity when it is pressed, or the release dynamics when it is released.

It is possible to mix the individual controllers to form larger controllers. Guitar synthesizers have been produced which use strings to determine pitch, but use separate keys to trigger the sounds (The SynthAxe is one example). Conversely, 'string' controllers consisting of six short strings mounted on top of a small box have been produced; they can be used to produce strumming and plucking effects in conjunction with a keyboard (or alternative pitch controller) that provides the pitch information.

Controllers have been developed to utilize most of the available resources of the performer. The pitch-bend wheel and the modulation wheel occupy the left hand (there is an underlying assumption that keyboard players are all right handed), whilst the right hand is presumably playing the keyboard, plus controlling after-touch. One foot is used for the sustain foot-switch, whilst the other foot is usually employed controlling volume or 'expression'. This leaves the mouth to blow into a breath controller with elbows and knees still available for future development (some organs apparently use knee controllers). The left hand may not be continuously occupied with pitch bend or modulation control, and therefore a few extra controllers can be provided for the left hand – some knobs or sliders that can be set to control parameters such as brightness or release time, which may well need adjusting in the course of a performance. With all these possible controls, a synthesizer player can begin to look rather reminiscent of a one-man band.

8.1.1 Performance

Regardless of how the control is achieved, the important consideration is that the player of a synthesizer should be able to interact with the instrument in a way that allows expression to be conveyed to the listener. The more natural and intuitive the controls are, the more they need to avoid imposing limitations on the player. Although the keyboard imposes many limitations on the player in performance, it is still the most common interface, and careful use of the additional performance controllers can minimize the problems.

Familiarity with the instrument's controllers can help considerably. The use of a single 'master' keyboard is useful in this context because the player will be comfortable with the keyboard and its controls and will not need to try and locate wheels or pedals in poor lighting conditions on stage. Careful preparation of the music can help to identify points where additional performance controllers such as pedals or even alternative controllers such as a string or a wind controller, may be required. For a player of an orchestral instrument, the instrument and its controllers are one and the same, and therefore no consideration is required for how to accomplish a specific musical result, whereas a synthesizer and its controls can be separated and can be changed to suit the player.

Some of the possible controllers are described in the remainder of this chapter.

8.2 MIDI control

8.2.1 MIDI

The wide adoption of the musical instrument digital interface (MIDI) has meant that it has become the dominant method of digital control for hardware. Even in software standards such as VST for audio 'plug-ins', MIDI-style controllers are still assumed.

Although MIDI (Rumsey, 1994) provides a large number of controllers, the basic underlying assumption is that a keyboard will be used to provide the main source of pitch and performance information; since the pitch and the velocity parameters are tied together in the note on and off messages while volume information, pitch bend, modulation and other parameters all require separate continuous controllers.

MIDI is capable of a detailed and precise control in some circumstances, but it has some limitations that reflect its keyboard-based origins. MIDI is poor at specifying the transitions between notes, which makes controlling guitar sounds difficult because the reuse of a string cannot be controlled properly, and attempting to control polyphonic vibrato requires the use of several channels of MIDI's mono mode, which is not widely supported by commercial synthesizers. Glissando and portamento effects are controllable only for parameters such as time through MIDI, and MIDI expects that each change in pitch associated with a note will initiate a new envelope.

MIDI provides two ways to control synthesizer (or sampler) parameters: controller messages and system exclusive messages.

8.2.2 MIDI controllers

MIDI controllers are a sort of general case of the pitch-bend or pressure MIDI messages. They are performance controls: modulation, expression, vibrato and so on. There is provision for a large number of possible MIDI controllers: more than 64,000, but only a few are typically used. The most common is the modulation wheel (controller number 1), often found next to the pitch-bend wheel on the left side of a synthesizer keyboard.

Controller messages have 3 bytes. The first byte identifies the message as a controller message and also indicates which of the 16 MIDI channels the controller is on. The controller number is the second byte. The 128 possible controller numbers are organized into three basic types:

1. Continuous controllers, for example modulation wheels and pedals.
2. Switches (theoretically just two values: on and off).
3. Mode messages to control note reception conditions.

Continuous controllers are divided into 14 bit (from 0 to 63) and 7 bit (from 64 to 95). Switches are a special case of a 7-bit continuous controller, and true on/off switches occupy a few numbers (from 64 to 69). Controllers 96–119 are split into two sections: the first covering 96–101 is the registered and non-registered parameters section, where huge numbers of parameters can be assigned to MIDI controller messages without using up valuable 14- or 7-bit controller numbers. In fact, it is possible to use just controllers 6 and 38 to control the most and the least significant bytes (MSByte and LSByte, respectively) of all these extra parameters. Controllers 102–119 were 'undefined', which means that they can be used, or mis-used for any purpose, by anyone.

In summary, the controller numbering looks like the following:

- 0–31, 14-bit controllers (MSByte);
- 32–63, 14-bit controllers (LSByte);
- 64–69, Switches (pedals and foot switches);

- 70–95, 7-bit controllers;
- 96–101, Registered and non-registered parameters;
- 102–119, Undefined;
- 120–127, Mode messages.

Not all controller numbers are assigned to specific devices such as pedals and wheels; in fact, most are not. Those that are not assigned seem to have been driven by historical precedent.

The 7- and 14-bit controllers

MIDI controllers come in several varieties. Some act as switches for just on and off controls: sustain pedals, for example. Some act as 7-bit controllers with 128 different values: volume pedals and even some slightly esoteric sustain pedals. But there are also 14-bit controllers where detailed control is needed. Using 14 bits allows up to 16,384 different values, which is enough for most purposes, and in fact, very few manufacturers take advantage of 14-bit controllers. For volume pedals, 128 values are often too much precision, and only 8 or 16 different volume settings are actually used in some cases. Modulation wheels often use all 128 possible values to give smooth transitions for vibrato amount, filter cut-off, and other parameters. The most common use of 14-bit precision is the pitch-bend message, which is technically not a controller at all! The ultimate limit on the number of values is the switches, where there are effectively only two values that are used.

The 7- and 14-bit controllers can work together. The range (maximum to minimum values) of a MIDI controller is normally thought of as being 0–27, which makes sense for a 7-bit controller, but what about 14-bit controllers? Where are those 16,000 values? In fact, the 14-bit controllers 'fill in' the gaps between the 7-bit values, which provide much finer resolution. Therefore the 7-bit controllers give coarse control, whilst the 14-bit controllers allow much more precise adjustment.

The 7- and 14-bit controllers share messages. The first 32 MIDI controllers appear to be 7-bit controllers, but they are actually just the most significant part of a 14-bit controller, which uses another controller number as the fine resolution part. Using computer-speak, the 'most significant byte' and 'least significant byte' are referred to by acronyms – MSByte and LSByte. The MSByte controllers can be used on their own as 7-bit controllers, whilst the LSByte controllers just add 32 to the MSByte controller number, thus volume (MSByte controller number 7) can also be finely tweaked with LSByte controller number 39, if your equipment supports this feature.

Note that an MSByte controller message on its own resets the LSByte value to zero.

Controller numbers

- Controller 1: Modulation wheel, MSByte
  Wheels are not the only source of control for this message. Levers, joysticks, sliders and pressure-sensitive plates can all provide the same physical control.

- Controller 2: Breath controller, MSByte
  Breath controllers convert breath pressure into a control signal. They are especially useful for controlling monophonic synthesizers that are used to play melodies.

- Controller 4: Foot controller, MSByte
  Usually a pedal, although some foot controller inputs will accept a control voltage instead.

- Controller 5: Portamento time, MSByte
  This control can be used as a switch or as a continuous control over portamento time.

- Controller 6: Data entry, MSByte
  The data entry front panel control may be a slider, a rotary dial or a keypad.

- Controller 7: Main volume, MSByte
  Often only the MSByte is used to control volume, which can cause 'zipper' noise because of the large jumps in volume caused by just 127 steps in volume.

- Controller 8: Balance, MSByte
  This controls the volume balance between two sounds, and is taken from 'organ' terminology.

- Controller 10: Pan, MSByte
  This controls the position of the audio in the stereo sound field.

- Controller 11: Expression controller, MSByte
  Typically a pedal, the expression controller provides a volume boost in addition to the main volume, and is intended to be used for accenting.

- Controller 12: Effect control 1, MSByte
- Controller 13: Effect control 2, MSByte
- Controllers 16–19: General-purpose controllers 1–4, MSBytes
  These are intended to be general-purpose controllers, which means that they will be defined by individual manufacturers to suit specific parameters in their instruments.

- Controllers 32–63: LSBytes
  Controllers 32–63 are the LSByte equivalents to the above-mentioned controllers.

- Controllers 64–69: Switches (7-bit controllers)
  64 Damper pedal (sustain)
  65 Portamento on/off
  66 Sostenuto
  67 Soft pedal

68 Legato
69 Hold 2

These are actually 7-bit controllers, although the most MIDI equipment will expect them to be of only two values: on and off.

- Controllers 70–74: Sound controllers 1–5
  Controllers 70–74 are the defined sound controllers. These have a default descriptive name, which is shown later, and are designed to provide simple 'global' quick editing facilities. They are intended to allow slight changes to a sound, probably during a performance, rather than editing actual parameter values permanently.

  The names are guides to suggested usage, but they can be redefined to suit a particular application. They are in two groups:

  1. Timbre controls
     70 Sound variation
     71 Timbre/harmonic content
     74 Brightness
  2. Envelope controls
     72 Release time
     73 Attack time

- Controllers 75–79: Sound controllers 6–10
  Controllers 75–79 are the undefined sound controllers. These do not have any predefined name, and can be assigned freely by a manufacturer.
  75 Sound controller 6
  76 Sound controller 7
  77 Sound controller 8
  78 Sound controller 9
  79 Sound controller 10
- Controllers 80–83: General-purpose controllers 5–8
- Controller 84: Portamento control
  This is an unusual controller where it indicates the note number from which the currently 'portamento'ed note started.
  Controller numbers from 102 to 119 are undefined.
- Controller 120: All sounds off
  Controller 120 is the all sounds off message. This is an improved version of the all notes off message, since it forces any sounds that are currently playing to stop as quickly as possible.
- Controller 121: Reset all controllers
  Controller 121 is used for the reset all controllers message, a sort of all notes off but for controllers instead of notes. Although this message can be interpreted as meaning just the MIDI controllers, it actually indicates

that the current state of all controllers, pitch bend, modulation wheel and pressure should be returned to the reset state (typically zero for most controllers, with the exception of the pitch-bend wheel, which resets to its center position (no pitch bend), and the volume control (controller number 7), which returns to full volume (i.e., 127)).

- Controller 123: All notes off
  The all notes off message is designed for use by sequencers when a sequence playback is stopped whilst some notes are still playing, although it often seems to be used to indicate when all the keys on a keyboard have been released.

  Finally, remember that the MIDI controllers are channelized, that is, the data is sent using a specific MIDI channel (Table 8.2.1).

8.2.3 System exclusive

The system exclusive message is unusual because it is the only MIDI message that has special bytes that indicate the beginning and the end. All other MIDI messages start with a special byte that indicates the type of message that follows, and also includes the MIDI channel number. The length of a message can normally be determined by knowing what type of message it is, but system exclusive (sysex) messages can be of any length, and therefore the start and the stop bytes are used to indicate the limits of the message.

The sysex provides a carefully designed 'loophole' to the MIDI specification that allows manufacturers to have extra information in their own specific formats embedded in MIDI messages whilst still retaining full MIDI compatibility.

The byte that immediately follows the sysex start byte is used for the identification of the owner of the message, and it is commonly called the manufacturer's ID number. This byte is rather like the address on an envelope: the envelope is ignored by everything except the correct addressee who can open it and see what is inside.

The actual contents of the sysex message are entirely up to the individual manufacturer, but typically it consists of sound data, editing or control functions. The format of these messages is left entirely to the manufacturer and although most manufacturers maintain a reasonably consistent standard within their own products, there is almost no commonality between manufacturers. Often, the format includes a few bytes that indicate the intended destination device; this ensures that only editing or control messages that are applicable to a specific device are acted upon. Most sysex formats have 1 or 2 bytes that indicate which parameter value is being changed, and then 1 or 2 bytes for the value of the parameter itself. To ensure that the message has been correctly received, some formats also include a checksum byte that can be used by the receiving instrument to determine if an error has occurred during transmission of the MIDI data.

Table 8.2.1 Some MIDI Controller Numbers

14-bit MSBytes	
0	Bank select
1	Modulation wheel
2	Breath control
4	Foot controller
5	Portamento time
7	Volume
8	Balance
10	Pan
11	Expression
12	Effect control 1
13	Effect control 2
7-bit controllers	
64	Hold, damper, sustain
65	Portamento
66	Sostenuto
67	Soft pedal
91	Effects 1 (ex external effects depth)
92	Effects 2 (ex tremolo depth)
93	Effects 3 (ex chorus depth)
94	Effects 4 (ex celeste depth)
95	Effects 5 (ex phaser depth)
Non-registered and registered	
6	Data entry MSBytes
38	Data entry LSBytes
96	Increment data
97	Decrement data
98	Non-registered parameter number LSBytes
99	Non-registered parameter number MSBytes
100	Registered parameter number LSBytes
101	Registered parameter number MSBytes
Mode	
121	Reset all controllers
122	Local control on/off
123	All notes off
124	Omni off
125	Omni on
126	Mono on
127	Poly on

Sysex editing and control messages are normally short: less than 10 bytes in length. Longer messages are normally complete sets of data bytes for a sound or a set of sounds. Information on the contents of the sysex messages for a specific synthesizer or a sampler is often included in the owner's manual.

8.2.4 After MIDI

MIDI has proved to be a major unifying force within sound synthesis and sampling for more than 20 years. But the leading edge of technology in the early 1980s was the transmission of 31,250 sets of 7 bits per second using affordable mass-market chips. A designer of a replacement for MIDI in the 20th anniversary year, 2003, would probably have looked at general-purpose rapid data transfer technologies like IEEE-1394, known as FireWire, with a rate of 50 million sets of 7 bits per second. Although FireWire saw wide acceptance as a method of connecting digital video camcorders to computers, it was less successful for general-purpose computer peripheral connections. USB 2.0, with similar data transfer speeds, but a hub-based approach, has seen very broad acceptance. MIDI signals conveyed along USB cables have become the 'invisible' face of MIDI as it passes into its third decade.

One protocol based on IEEE-1394 (FireWire) rather than USB 2.0, called mLAN, and launched by Yamaha in the early twenty-first century, is intended to allow for the moving of multiple channels of digital audio and other music-related data such as MIDI and synchronization signals around recording studios, although the number of pieces of equipment that supported mLAN is still small in 2008. The integration of several separate pieces of conventional cabling into one mLAN cable carrying them all within a single digital link has the advantage of neatness, but has less of the immediacy of MIDI or audio cabling, where connecting a keyboard to a sound module, or that module to a mixer, has a simple directness.

In one way, the success of MIDI has led to it becoming a ubiquitous lower limit of connectivity, but this wide acceptance has meant that the conventions of MIDI (and its limitations) have become part of the technology of music-making and are unchallenged and unquestioned. This may have restricted the development of extensions for specific purposes like ZIPI and may also restrict the future development.

ZIPI

ZIPI (*The Computer Music Journal*, Vol. 18, No. 4) was a proposal made in 1996 for an alternative interface for connecting a performance controller to a synthesizer. It provided much more detail and flexibility in the way that pitch and performance information are conveyed than what the MIDI offers. In particular, it avoided making any assumptions about the type of controller that would be used, and therefore there were no keyboard-specific limitations, unlike some of the assumptions built into MIDI. But ZIPI was not intended as a replacement for MIDI, instead it was more a means of transporting performance control from a player's instrumental controller to a remote synthesizer. The exact format of the player's controller was not defined: it could be a keyboard, a stringed instrument or a wind instrument. ZIPI did not see wide commercial adoption, but many of its aims of increased performance controllability could be implemented using the Open Sound Control (OSC) protocol (a second visit at the aims of ZIPI, from the same team) or HD-MIDI (from the MMA) with future mLAN-based or USB 2.0 connected devices, or over IP connections.

Gibson's MaGIC

Gibson's Media-accelerated Global Information Carrier (MaGIC) was a patented multi-channel audio-over-IP proposal that Gibson worked on from an announcement to the Audio Engineering Society in 1999 through to 2003 and beyond. A guitar was announced, the Gibson HD.6X-Pro (also known as the Digital Les Paul or DLP), and was demonstrated in 2007, using the protocol. MaGIC seems to have been part of the Residential Ethernet study that eventually became part of the IEEE A/V Bridging study for providing audio and video over Ethernet, which is still going through the standardization process.

The website for MaGIC does not seem to have been updated since 2003, and an inquiry email bounced. But reviews of the HD.6X-Pro guitar using MaGIC appeared in the early 2008, and therefore MaGIC would seem to be still active.

RTP-MIDI

RTP-MIDI was introduced by the same team at Berkeley that produced ZIPI and the same team that produced OSC, RTP-MIDI is a payload format, that is, a way of specifying how MIDI information can be carried over IP 'Ethernet' LAN cabling and wireless LANs. RTP-MIDI is the Internet Engineering Task Force (IETF) standard RFC 4695 (and RFC 4696 for the implementation guide). It has been produced in collaboration with the MMA and the MPEG.

8.3 Keyboards

The music keyboard has a distinctive appearance that is widely used as a visual metaphor for music or pianos. The interlaced black-and-white keys are widely used for note selection, as well as for starting and ending envelopes.

Keyboards provide two major outputs: discrete pitch information about the notes that are being played and event information about the start and the end of note events. Both of these are produced by pressing down on the keys and releasing them. The convention is that pressing down on the keys produces both a note pitch indication and a note start event. Releasing the key allows it to return to its rest position and produces a note end event.

Some keyboards also provide additional information on the velocity with which the key is pressed or released; this is known as either 'attack velocity' or 'release velocity', although sometimes the term dynamics is used as a synonym for velocity. When a key is pressed down and extra pressure is applied, some keyboards will produce information about after-touch or key pressure. These are both covered in Section 8.4.1.

'Velocity' measures the speed of the key movement from the 'off' to the 'on' state, or from the 'on' to the 'off' state.

8.3.1 Polyphony

The simplest keyboards are monophonic. A chain of resistors connected in series with a voltage applied across them can be used to output a voltage whose value is proportional to the position on the chain. By using the keys of a keyboard to switch a contact wire onto the chain of resistors, a keyboard can be used to provide a monophonic note output. By using a second switch contact to produce a note status indication, note event information can also be produced. This is the type of keyboard that was used in early analogue synthesizers (Figure 8.3.1).

By extending the basic design of the monophonic 'analogue' keyboard, it is possible to produce a keyboard that will produce two voltages: one for the highest note that is being pressed down and the other representing the next lowest note that is held down. Such 'duophonic' keyboards are often found on analogue performance synthesizers.

Keyboards based on chains of resistors are normally designed so that they have the so-called top-note priority. This means that if several keys are pressed down, then only the highest voltage will be produced. This allows a two-handed performance technique to be used where notes are held with the left hand whilst additional notes are played staccato with the right hand. Each time the right hand is released, the pitch jumps down to the note being played by the left hand. Duophonic keyboards provide separate storage for these 2 voltages and allow two-note chords to be played instead.

Polyphonic keyboards can be produced by extending the 'resistor' chain method, but in practice, digital scanning techniques are used. A counter is used

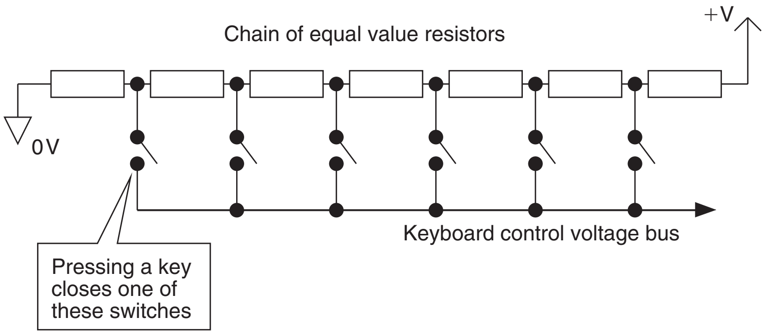

+V
Chain of equal value resistors
0V
Keyboard control voltage bus
Pressing a key closes one of these switches

FIGURE 8.3.1 An analogue synthesizer keyboard uses a chain of resistors, and pressing a key closes a switch that connects a bus to the chain of resistors. Only six switches are shown, but a real keyboard would have 30 or more keys. The position in the chain determines the output voltage. A separate switch (not shown here) produces the gate signal and a trigger pulse.

to send a pulse to each of the key contacts in turn, and any keys that are pressed down will allow the pulse to be switched onto a common bus-bar contact, and the keys that are pressed down can be determined by examining their relationship to the counter. The keyboard scanning is thus based on time-division, and the rate at which the keyboard is scanned determines the minimum timing resolution with which a key-press can be measured. To minimize the required wiring, and to align with byte-wide scanning chips, most scanned keyboards arrange the keys in the form of an 'N × 8' matrix rather than a linear form (Figure 8.3.2).

Scanned keyboards produce note pitch and event information directly from a single contact switch. The polyphony and note priority (if a monophonic keyboard is being emulated) are controlled by software. For velocity information to be produced, a second switch or a changeover switch is required so that the time between the start and the end of the key movement can be determined.

8.3.2 Types of keyboard

Keyboards come in two major forms: organ type and piano type. Although both are fitted to synthesizers, the organ type tends to be used in low-cost and mid-range products, whilst the piano type is more often found in 'master' controller keyboards and 'top-of-the-range' or 'flagship' products.

Organ-type keyboards have light, hollow, plastic keys, and the black keys normally slope downwards away from the front of the keyboard. Organ keys tend to be about 140 mm from the front of the keyboard to the back of the playable key surface. The key action is fast and very light, with the key being returned to the rest position by a spring. Organ-type keyboards sometimes have small weights attached to the underside of the front of each key to provide additional weight when they are used as master keyboards; this is intended to emulate some of the feel of a piano-type keyboard. Organ-type keyboards as used on synthesizers are normally either five octave (61 keys) or just over six octaves (76 keys) – frequently referred to as an 'extended' keyboard.

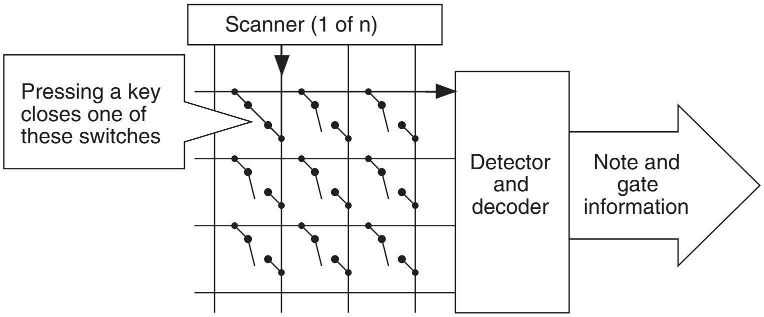

Scanner (1 of n)
Pressing a key closes one of these switches
Detector and decoder
Note and gate information

FIGURE 8.3.2 A scanned keyboard uses a matrix of switches that connect one of the outputs of a scanner (it cycles round each of its outputs, putting a logic one signal on each in turn) to a detector and a decoder circuit. One switch can be used to produce note and gate information. Velocity information would require an additional switch or a changeover switch.

For use in small portable 'fun' keyboards and synthesizers, a variant of the organ-type keyboard is often found. This has narrower keys (an octave fits into the space of an 11th on a normal keyboard) and the playable surface of the keys is only about 80 mm from front to back.

Piano-type keyboards have heavier keys that are normally wood covered by a plastic molding. The black keys normally have flat tops. Piano keys are about 150 mm from the front of the keyboard to the back of the playable key surface. The piano 'action', a complex mechanical arrangement of levers and weights, was originally designed to transfer the downwards movement of a key into the hammer striking the string, but in this application, only the mechanical 'feel' is required since there are no strings. In a piano action, the return of the key to the rest position is not a simple spring; it is related to the bounce of the hammer on the string. Piano-type keyboards are often referred to as 'weighted' because of the heavy keys and the feel of the 'action' relative to an organ-type keyboard. Piano-type keyboards normally have at least 76 keys, and frequently more.

8.4 Keyboard control

8.4.1 Physical keyboards

Although the basic music keyboard produces only pitch and event information, the action of pressing the key and then applying pressure to the key can be used to produce additional information.

'Velocity' is the name given to the output from the keyboard that represents the rate at which the key is pressed or released. A pair of contacts is used to measure the time that the key takes to travel from just after when it is first pressed to when it is almost fully down. Velocity is thus polyphonic, since it is measured individually for each key-press. Changes in velocity whilst the key is being pressed do not normally affect the velocity, since the measurement is based on just the time difference between the switching of the two contacts. Attack velocity is measured when the key is first pressed and is used to control the dynamics of the sound, often the level and/or the timbre, whilst release velocity is normally used to control the length of the release segment of the envelope. Release velocity is only rarely implemented in keyboards, although many synthesizer sound generators will respond to release velocity.

'After-touch' or 'key pressure' is the name given to the additional pressure that is applied to a key when it is being pressed down. The key requires a certain amount of pressure to overcome the spring that pulls it back to the rest position when the key is released, and the after-touch measures any additional pressure that is applied in excess of this. After-touch can be a global parameter where the highest pressure applied over the entire keyboard is used as the output value, or it can be measured individually for each key. These are called 'monophonic' and 'polyphonic', respectively.

Some keyboards have tried to provide additional control by using side-to-side movement of the keys, whilst others have tried to use the position of the finger on the key's upper surface as a controller. None of these extensions to the keyboard as a controller has been commercially successful.

The keyboard is thus a limited controller that enjoys wide commercial success. The rise of the computer as a means of producing music has gone some way to reducing the importance of the keyboard.

8.4.2 'Qwerty' and controller keyboards

When using sequencer software on a computer, a commonly used alternative to entering notes using a conventional music keyboard (usually connected through MIDI) is to use the qwerty keyboard of the computer itself. Because the 'qwerty' mapping is not universal, the mapping is easiest to explain by relative positioning. For example, in French-speaking countries, computers have 'aserty' keyboards, for example. In general, for 'Computer MIDI keyboard', the middle row of keys is used for the white notes, and corresponding keys on the row above are used for the black notes. Some of the remaining keys can be used for octave transposition and velocity control. A modifier key plus key combination is used to switch between the music keyboard mode and the qwerty mode (e.g., Control + Shift K).

Computer MIDI keyboards are convenient for playing notes that are restricted to the limited range of pitches that are available. Outside that range, octave switching is required. But the convenience of having a keyboard immediately and directly available at the computer is very great, and computer MIDI keyboards are used a lot. Similarly restricted pitch ranges, plus the same octave switching, and with additional controllers, can be found in the 'controller keyboards' that are often used as an alternative to computer MIDI keyboards. These normally only have a couple of octaves of keys, but this keeps the width narrow and therefore does not require the space of a five octave (or wider) conventional keyboard. Keyboard-less versions of these are often used as DJ controllers, mixer controllers or as performance controllers for sequencer software.

8.4.3 Virtual keyboards and piano rolls

Sequencer software often uses a 'piano roll' metaphor for creating and editing sequences, particularly step sequences. Pitch is normally mapped vertically, whilst time moves horizontally across the screen, although these conventions are not always true. The mouse is used to enter notes by clicking at points on the 'pitch/time' grid and then dragging horizontally to set the length of notes. Some piano roll displays allow notes to be freely 'painted' in using the mouse. Velocity is typically adjusted by changing the height of vertical bars in a window below the pitch/time grid. Piano roll displays provide a very clear and unambiguous display of notes and pitches, although for multiple simultaneous notes, the velocity display can be difficult to interpret and most software provides different colors or a means to cycle through the notes to allow individual control.

A variation of the piano roll display is the notation display, where notes can be entered using music notation. These normally require the use of a palette of symbols in much the same way as bit-mapped paint software.

Controllers related to pitch input inside computer sequencers include transposers and arpeggiators.

8.4.4 Drum pad controllers

Drum machines typically provide a number of rubber velocity-sensitive pads that are allocated to drum sounds. These are often also capable of producing MIDI note messages with specific pitches, and therefore can be used to generate pitched note events – a 'struck' keyboard. There are also stand-alone trigger pads that are generic pads that can be used as controllers for drum machines or as pitched note controllers for sequencers. All of these can be used as alternative to the 'key-press' or computer MIDI keyboards as inputs of pitch and velocity information to sequencers.

Drum pad controllers are also useful for MIDI-controlled drums that are produced either by drum machines, by samplers, or by sequencers or host software running on a computer. The tactile physical interface provided by these pads enables drummers and percussionists to convert their physical playing into digital form and so to put their performance into a sequencer. Drum sounds are also well suited to hocketing, where key switching or legato mode controller switching can be used to dynamically and interactively change the sounds produced by drum pads as the drummer is performing.

Drummers are often used to provide the timing for live performance with a tempo clock derived from their drumming used to synchronize sequencers and other time-related stage effects like slide projectors or DMX-controlled lighting sequences.

The idea of using a non-keyboard set of pads or other controllers to trigger musical events is also used in stage and theater performance. MIDI start and stop messages can be generated by pad or switch controllers, and sampled music, sequenced phrases and even complete compositions can be triggered using this type of controller.

8.5 Wheels and other hand-operated controls

The early analogue modular synthesizers used rotary control knobs to control the majority of the non-switched functions. Knobs were thus used for the first controls for pitch bend, and modulation controls were merely the output level controls for low-frequency oscillators (LFOs). But knobs were not very satisfactory for live performance use, and a number of alternative approaches were tried for providing the same sort of pitch-bend and modulation control, but in a more intuitive way. The four major contenders were the following:

1. wheels,
2. levers,
3. joysticks,
4. pressure pads.

8.5.1 Wheels

Wheels are a development of the rotary edge controls that turn a rotary control onto its side and use a disk instead of the knob. Pushing the disk forward or backward turns the rotary control. The disks used range 40–80 mm in diameter, and only about a third or a half of the disk protrudes above the panel surface. Some disks have a textured or grooved edge, whilst others are smooth; some manufacturers make them out of clear plastic with polished edges. A small semicircular cut-out or detent provides a reference point for the position of the disk. The disk movement normally uses about 90° of rotation, which is a quarter of a complete circle.

For pitch-bend purposes, the detent is in the center of travel of the rotary control, and therefore the pitch can be changed up or down, the rotation is about 60° away from the center point of the pitch bend. A spring arrangement is normally used to return the pitch-bend wheel back to the center position, and this is reinforced by the use of a second detent and a sprung cam follower that slots into the detent to provide a tactile center position into that the wheel clicks. Pressure on the wheel is required to overcome the detent and the springs before it will move. For modulation use, the detent is with the disk full towards the player, and modulation is added as the disk is pushed forward. No spring return is used with most modulation wheels.

Although most wheels are located so that they move away from or towards the player, some wheels have been designed that move from side to side, so that to bend the pitch upwards the wheel is moved to the right, whilst for pitch bend downwards the wheel is moved to the left. Modulation is added by pulling the wheel towards the player or pushing the wheel away from the player.

8.5.2 Levers

The lever replaces the disk with a short lever with a length of about 50 or 60 mm. The rotational movement is normally less than a wheel: a maximum of 90° and a minimum of about 45°. As with wheels, the pitch-bend lever is sprung so that it returns to the central detent position.

8.5.3 Joysticks

Joysticks combine the pitch-bend and modulation functions into a single control. Both side-to-side pitch and forwards/backwards modes have been used, with modulation using the opposite axis. Additional control has been provided on some joysticks by allowing the joystick to be rotated or moved up and down.

8.5.4 Pressure pads

Soft conductive rubber pressure pads and metal plates fitted with strain gauges have been used to provide pitch-bend and modulation controls on some synthesizers. Separate pads are required to bend the pitch up and down, whilst the modulation pad can only be used to add modulation interactively – it cannot be set to a value and left, unlike the modulation controls provided by wheels and levers (Figure 8.5.1). The two main controllers that wheels and similar devices are used to control are pitch bend and modulation.

8.5.5 Pitch bend

Continuous control over the pitch is achieved by using a 'pitch-bend' controller. These are normally rotating wheels or levers and usually change the pitch of the entire instrument over a specified range (often a semitone or a fifth). They produce a control voltage whose value is proportional to the angle of the control. Pitch-bend controls normally have a spring arrangement that always returns the control to the center 'zero' position (no pitch change) when it is released. This central position is often also mechanically detented, so that it can be felt by the operator, since it will require force to move it away from the center position.

8.5.6 Modulation

Modulation is controlled using rotary wheels or levers, where the control voltage is proportional to the angle of the control. Modulation controllers are not normally sprung so that they return to the center position. Some instruments allow pressure on the keyboard to be used as a modulation controller.

There have been several attempts to combine the functions of pitch bend and modulation into a single 'joystick' controller, but the most popular arrangement remains the two wheels: pitch bend and modulation.

8.5.7 2D controllers

Two-dimensional (2D) controllers, in the form of pads rather than joysticks, have become popular in the early twenty-first century and are built into some keyboards and performance effects units. The Korg Kaoss pad is one example that started out as an audio effects unit, but has been extended for use with twin record decks by DJs, and even into the video effects or video synthesizer area too.

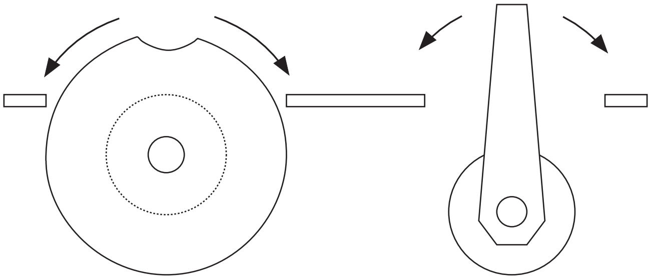


FIGURE 8.5.1 Wheels, levers and joysticks are all alternative ways of presenting a user interface to a rotary control. This diagram shows a cross-section through a wheel and a lever.

8.6 Foot controls

Foot controllers or foot pedals are rotary controls that are operated by the player's feet. A flat rectangular plate is covered with a ribbed or a textured rubber surface and allowed to rotate and thus controls a rotary control. Pedals provide a control voltage that is proportional to the angle of the pedal: the pedal is hinged at about one-third of the way along the upper plate. Although usually associated with volume control, they can also be used as modulation controls. When used as a volume control, foot pedals normally have a smooth travel over about two-thirds of the rotary range (about 30° total travel) and then a spring is used to provide additional resistance to further movement. This allows extra volume to be added for specific expression purposes (Figure 8.6.1).

Not all foot pedals use rotary potentiometers to sense the angle of rotation. Opto-electronic and magnetic rotation sensors can also be used. Some pedals can also be used as pitch-bend controls, with springing to return the pedal to a central default position where no pitch bend is produced.

8.6.1 Foot switches

Foot switches are foot-operated switches. They can be produced in several forms. Some operate like the sustain pedals on a piano, where a short metal lever is pressed downwards to activate the pedal. Others are like small variants on foot pedals, whilst a third type is merely push button switches. Whilst foot pedals are continuous controllers, foot switches normally have two values only, although there are some rare multi-valued variants used to control sustain on

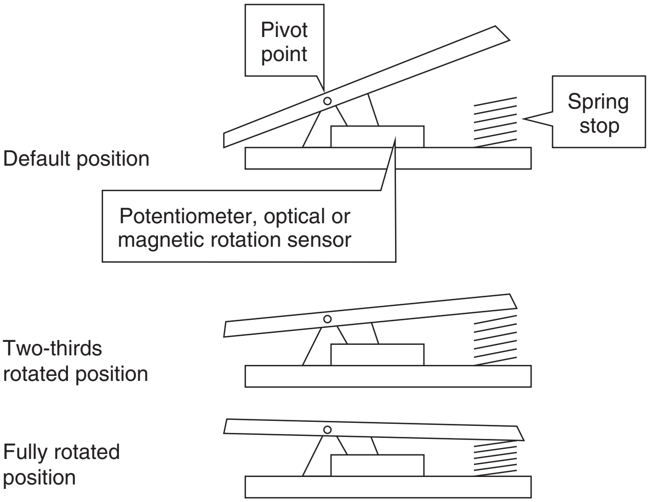

Pivot point
Spring stop
Default position
Potentiometer, optical or magnetic rotation sensor
Two-thirds rotated position
Fully rotated position

FIGURE 8.6.1 Foot controllers have a restricted angle of rotation, that is, about 30°. The rotation is sensed by any methods: mechanical linkage to a potentiometer, optical sensors or magnetic rotation sensors. The spring provides a 'soft' end point that can be used to provide additional volume for expression purposes by using additional pressure on the foot control.

pianos. Foot switches are used to control parameters such as sustain and portamento but they can also be used to select sounds on synthesizers and to start or stop drum machines.

8.6.2 Foot-operated keyboards

Organs sometimes feature keyboards that can be operated by the feet of the performer. The use of foot-operated keyboards for synthesizers has been less successful. One notable example was the Taurus bass pedal, a single octave keyboard produced by Moog in the 1970s, and designed to be played by the feet.

8.7 Ribbon controllers

The ribbon controller is a variation on the pitch-bend wheel. Instead of a rotating wheel, a flexible conductive material is held above a metal plate, and when pressed with a finger, the material touches the plate and a voltage is produced. (Although the ribbon hides what is actually happening from sight, it is rather like holding a guitar string down with a finger.) Moving the finger on the material changes the voltage, and therefore the controller can be used as a pitch-bend control. Unlike a pitch-bend wheel, it is possible to jump in pitch by pressing a finger onto the ribbon, and then removing the finger causes a jump back to the default pitch. Ribbon controllers have the advantage of mechanical simplicity and small size, but the material does tend to wear out.

There are two main types of ribbon controllers:

1. Short ribbon controllers, which have a silver-colored cloth-like appearance, are about 100-mm long and 15-mm wide and have a central raised indication. These have been used in Moog, Yamaha and Korg synthesizers.
2. Long ribbon controllers, which have a black flocked plastic appearance, are about 300-mm long and 10-mm wide. The pitch changes relative to the starting point and the amount of movement along the ribbon by the finger. They are thus more suited to producing portamento and glide effects rather than pure pitch bend. The Yamaha CS-50/60/80 series of polyphonic synthesizers used this type of ribbon controller (Figure 8.7.1).

Korg's Prophecy performance monosynth extended its use of a ribbon controller, it provided additional rotary movement control by mounting the ribbon on a 'log'-like wide modulation wheel, as with the joystick, this combined two-axis control remains the exception rather than the rule.

8.8 Wind controllers

Wind controllers come in two forms: breath controller and wind instrument controllers.

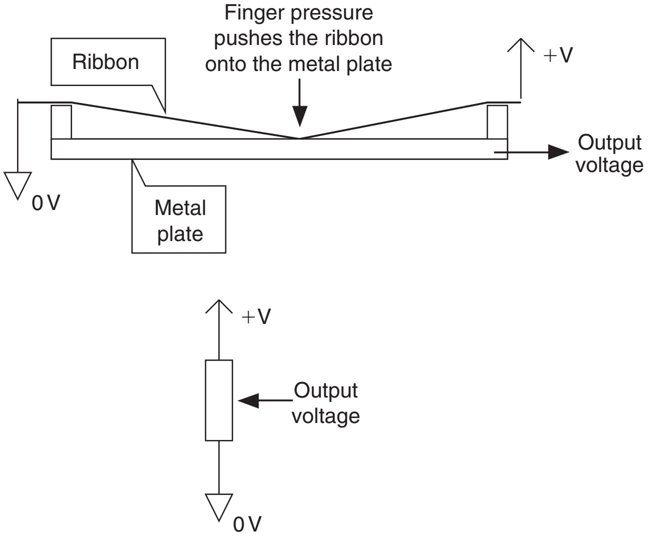


FIGURE 8.7.1 Ribbon controllers are actually a variation on a potentiometer or slider volume control on a mixing desk. Instead of having a slider that moves up and down a resistive track, the ribbon is moved onto a metal plate and therfore produces a voltage that is proportional to its position.

Breath controllers are simple pressure measuring devices that are blown into by the player. The output voltage is used to control either modulation or the envelope of a sound by replacing the volume control.

Wind instrument controllers use a controller that is based on a wind instrument such as a flute, clarinet or saxophone and convert the finger presses on the keys into pitch information and the blowing into additional modulation and volume information. One or more selectable modified forms of Boehm fingering are normally used, with extensions to cope with the larger pitch range provided by a synthesizer sound generator. Additional control keys allow octave switching, portamento control and sustain effects to be controlled from the wind controller.

Wind controllers are particularly effective when they are used to play lead line and solo melody sounds.

Although wind controllers are normally monophonic, some provide the ability to hold notes and then play additional notes on top, thus allowing the building up of sustained chords (Figure 8.8.1).

8.9 Guitar controllers

Given the enormous interest in using electronics to change the sound of the electric guitar, with a huge range of effects pedals being available, using a guitar as the controller for a synthesizer seems like an obvious development. Unfortunately, using a guitar for this purpose is not easy. There are a number of technical problems whilst using a guitar as the interface between a player and a synthesizer (Figure 8.9.1).

The sound that is heard from an electric guitar is produced by the steel strings vibrating over small magnetic coils in the pickup. This induces a current in the pickup coils and this is then amplified to produce the guitar sound.

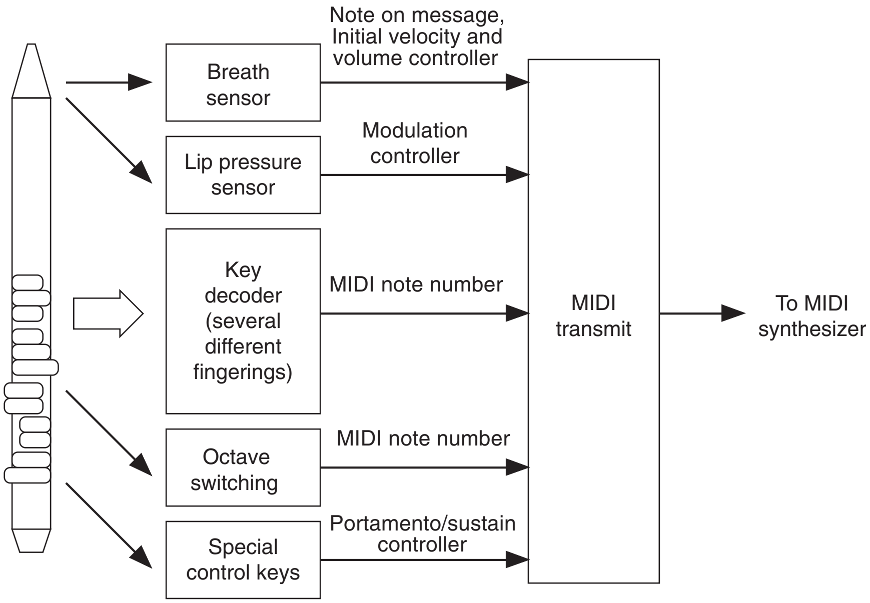


FIGURE 8.8.1
Wind controllers provide sensing for breath pressure and lip pressure (which are converted into MIDI controller messages for volume or modulation) and convert the keying into MIDI note numbers. Special octave switching keys and control keys (portamento/sustain) are also normally provided.

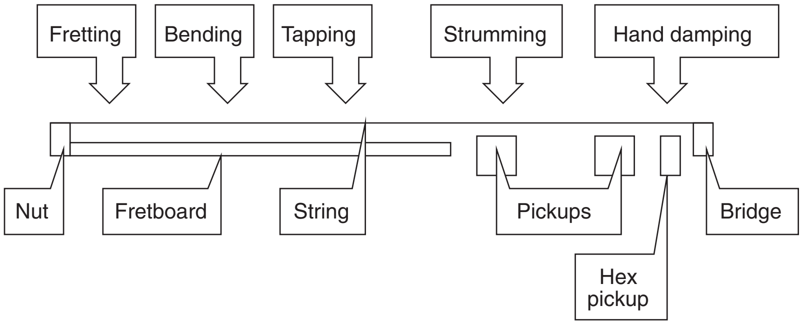


FIGURE 8.9.1
A guitar player can pluck, strum, tap or bend any of the six strings, which may be open or fretted. This cross-section through a guitar shows some of the major performance features and techniques.

Normal pickups are designed to ‘pickup’ the vibrations from all the strings, but in order to process each string’s pitch separately, the pickup needs to produce individual signals for each string. This is normally achieved by using a ‘hexaphonic’ or hex pickup, which has separate coils for each string.

(The Gibson HD.6X guitar also features a hex pickup)

Even with a separate signal for each string, extracting the pitch that a string is playing is far from straightforward, the pitch is dependent on the position at which the string was fretted, as well as any string bending or use of the tremolo arm, and it may also depend on how the string was plucked or strummed, for example, the player may have deliberately produced a harmonic rather than the pitch that might be expected.

In addition, the player can control the volume of each string by the amount of effort that is used in plucking or strumming the strings. This needs to be

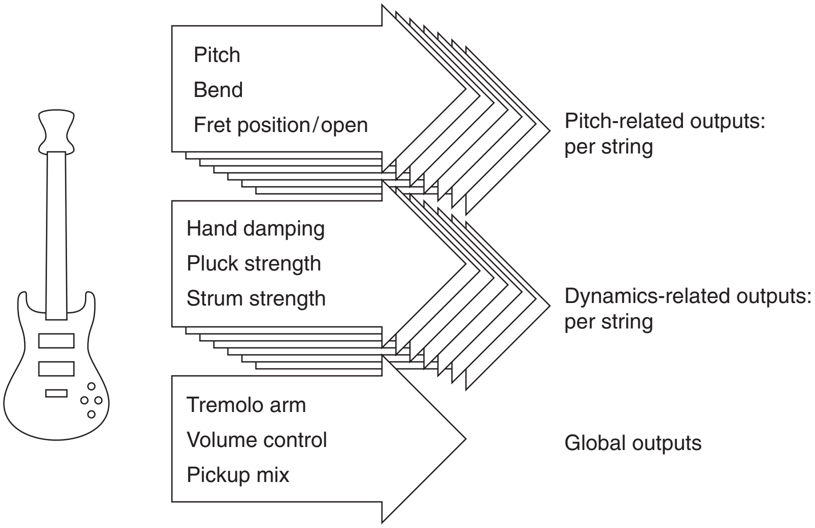


FIGURE 8.9.2
A guitar controller produces a large quantity of performance information.

determined by the guitar controller so that appropriate control signals can be transmitted to the synthesizer. But the player also has additional control like string damping, which is achieved by placing the hand lightly against the strings near the bridge. The timbre that the strings produce can be varied by altering the position at which the string is plucked or strummed, that is, the closer the position is to the bridge, the brighter the sound. Strings may also be tapped onto the fretboard to produce a sound, in which case the position is nowhere near the bridge at all. All of these performance techniques, which are available to the guitarist, make the task of converting a normal electric guitar into a suitable controller for a guitar synthesizer very difficult (Figure 8.9.2).

Approaches that have been tried include the hexaphonic pickup with sophisticated signal processing to try and extract the pitch; wiring up the frets so that they can be used as electronic switches so that the fretting can be monitored; and even using acoustic radar signals along the strings so that the position of the plucking and fretting can be determined. None of these methods offers a complete solution, but development work is ongoing, and the limitations of the guitar as a controller seem to be gradually diminishing as the complexity of the solutions rises.

Rather than to use a normal guitar as the controller, some manufacturers and experimenters have chosen to use a controller that has some elements of a guitar, but deliberately modified so that they suit the conversion process. In addition the wiring of the frets so that the fret position can be determined, some of these designs separate the strings that are fretted from those that are plucked or strummed, and others provide keys to trigger the string events instead. Most of these solutions are also technically complex, and none has been commercially successful.

One of the most useful results of guitar synthesizer research is arguably the hex pickup. By taking the six separate audio signals and applying distortion effects to them individually, a range of effects can be produced that do not have the large amounts of inter-modulation distortion present when a normal guitar pickup signal is distorted, can be produced instead the result is a bright, synthetic sound that has many of the expressive capabilities of the guitar, but without the problems of chords producing too much distortion. Some guitar synthesizers have used this hex signal to drive the strings through heads similar to the record heads on a tape recorder, which allows the vibration of the string to be captured and amplified, opening up possibilities for increased sustain.

Although not a guitar synthesizer, the review, by Craig Anderson in the January 2008 issue of *Guitar Player* magazine, of the Gibson HD.6X 'digital' guitar is still interesting in the context of hex pickups, since the review hardly mentions the digital transport of the audio using the MaGIC technology. Instead, the review concentrates on the possibilities opened up by having six independent audio outputs for the strings. The hex pickup does seem to be a very significant alternative to the conventional guitar pickup.

Using a guitar as a controller for synthesizers is only one aspect of using a guitar to create sound. Most electronic keyboards do not actually generate an acoustic sound, whereas a guitar produces an intrinsic sound regardless of any additional capability from a hex pickup or other sensing mechanism. The guitar is thus a composite instrument, mixing together an acoustic sound, a processed sound from the pickups, separate sounds from a hex pickup and synthesized sounds.

8.10 Mixer controllers

Mixer controllers control mixes. They can use either rotary or linear sliders, but are essentially a compact controller that can be used to control the volume or pan of a number of channels in one convenient device. Whilst they can be used for digital mixers, their primary use is with sophisticated digital audio workstation software, where many parameters can be controlled through MIDI controls. On-screen controls are fine for simple, edit-one-parameter-at-a-time operation, but for live use, it is often essential to be able to change two or more parameters more or less simultaneously. A controller with lots of sliders or rotary controls provides a solution and makes the computer much more immediately controllable.

DJ controllers are a specialized version of the more generic mixer controllers.

8.11 DJ controllers

DJ controllers are not a way of controlling synthesizers by moving DJs around! Instead, the term means controllers that DJs might use, or controllers derived from those that DJs use.

Twin decks and mixer with the all-important cross-fader have become synonymous with dance music and clubbing, but the user interface has applications beyond just the many software emulations and hardware controllers that have been produced. The metaphor of one deck playing whilst the other is prepared can be adapted to controlling two different tracks, perhaps with different sounds or different rhythmic variations. The cross-fader is a useful device for switching between two track variants, and the same techniques for moving back and forth in time to the beat between the two tracks can be used. Controllers intended for controlling DJ software are designed for live use, and therefore are rugged, particularly the cross-fader, which makes them very suitable for controlling any electronic music-making in a live environment.

DJs also use a number of other live performance controllers: 2D controllers such as the Korg Kaoss Pad and 3D controllers such as the Alesis airFX can be used to provide control over sounds in a visual way.

8.12 3D controllers

Three-dimensional (3D) controllers all provide control by moving the hands around in space. Whilst this sounds like it should offer unrestricted control over three parameters at once, in practice, holding your hands in a specific position and moving them around without anything to lean on or support them can be very tiring. The following are several ways to achieve 3D control:

- Capacitive, as used in the Theremin, where the hands affect the electric field around two aerials that are controlling the pitch and volume of a sine wave oscillator.
- Infrared (IR), as used in the Alesis airFX effects unit, where infrared light emiting diodes (LEDs) send infrared light upwards, where hand movements then reflect it back to sensors to detect 3D movements. Roland's D-Beam controller uses a simpler arrangement of LEDs to give 2D control.
- Radar or more precisely, ultrasonic Doppler-shift controllers are similar to some home burglar alarm sensors and depend on movement changing the pitch of very high-frequency sounds, which are then detected and decoded into controller signals.

8.13 Front panel controls

Controllers have had a marked influence on synthesizer design. In addition to the performance controllers described in the majority of this chapter, a synthesizer also has front panel controls. The knobs, switches and sliders that are used to control the synthesizer in non-realtime performance are just as important to the function of the instrument, and they have changed significantly in the lifetime of the synthesizer.

Early analogue modular synthesizers used knobs and switches to set the operating parameters of the modules. Interconnections were made using patch-cords that were plugged into front panel sockets. This imposes a number of constraints on the use of such a synthesizer: the patch-cords can obscure the front panel, making it difficult to see the settings of the knobs and the switches and it also makes it awkward to change the patching rapidly.

As synthesizers developed, both monophonic and polyphonic versions streamlined the patching, reducing it down to the voltage-controlled oscillator/filter/amplifier (VCO/VCF/VCA) format. Switches could now be used to control the routing of controllers like LFOs and envelopes, and the front panel could be designed so that it represented the signal flow through the synthesizer. The result was the front panels of the late 1970s and the early 1980s, where the front panel used a large number of knobs and switches, and occasionally some sliders. In most cases, each knob or switch controlled a single parameter, which meant that learning the front panel layout tells you how the synthesizer produced the sounds and what to adjust to change a sound.

A much more minimalistic arrangement was introduced by the first digital synthesizers. The Yamaha DX7 provided one knob and a lot of buttons. The buttons were assigned so that there was usually one button per parameter, although this was complicated slightly by having several different modes, where the buttons had different meanings. Even so, for most editing, you pressed a button to select the parameter with one hand (the right), and adjusted the value with the other hand (the left). This two-handed approach to editing can be very fast, but it requires the co-ordination of both hands and the eyes, and a considerable amount of concentration.

Computer-based editors can be used to replace this minimalistic interface, but screen redraws with large numbers of displayed parameters tend to be slow. The advantage of using this type of interface is that it is easy to scale to a rack-mounting expander module with a limited front panel area since few controls are required and the multiple selection buttons can be replaced by a single parameter selection knob or slider. As this two-handed editing interface has been developed, the slider has been augmented by additional controls such as a rotary dial and increment/decrement buttons.

When larger liquid crystal displays (LCDs) became available, the emphasis changed from the button selection of parameters to using the display itself. By arranging a row of assignable buttons or softkeys underneath the display, the selection of parameters could be achieved by reusing the softkeys. LCDs gradually developed from character-based to full graphics capability, which allowed the screens to become more and more the focus of the editing, with an increasing number of softkeys clustered around the increasingly large display.

With large displays, the front panel controls became less specific to the synthesizer, and sound selection keys, numeric keypads and play/edit mode buttons replaced the named parameter buttons. This placed an increasing load on the display, but allowed the front panel to stay static whilst the software could be changed and updated. As less and less controls were required when the display became dominant, front panel space was then available for additional performance controllers – track-balls and joysticks are two examples of methods used to permit rapid real-time control of parameters from the front panel of a synthesizer (Figure 8.13.1).

The mid-1990s saw the first introduction of touch screens with sophisticated graphical interfaces. Although novel, and stylistically impressive, the sparsely populated front panels tend to force the use of the touch screen with only limited alternative methods of controlling the synthesizer. Early touch

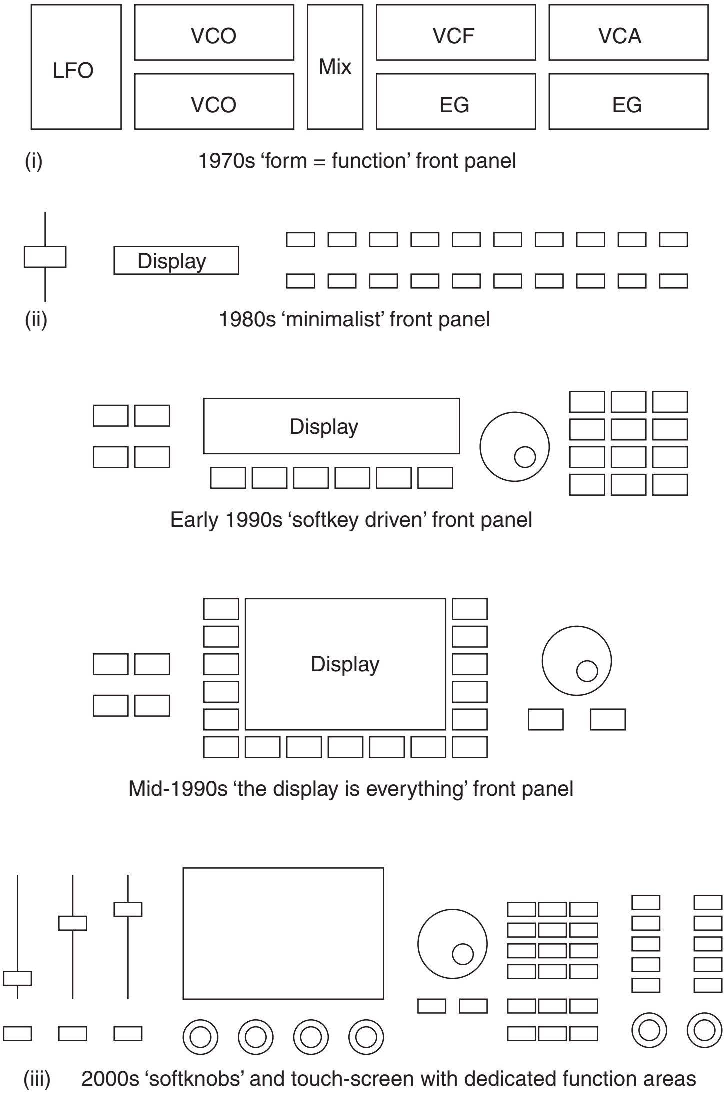

LFO
VCO
VCO
Mix
VCF
EG
VCA
EG
(i)
1970s 'form = function' front panel
Display
(ii)
1980s 'minimalist' front panel
Display
Early 1990s 'softkey driven' front panel
Display
Mid-1990s 'the display is everything' front panel
(iii)
2000s 'softknobs' and touch-screen with dedicated function areas

FIGURE 8.13.1 Front panels have developed with technology. (i) In the 1970s, the layout reflected the structure of the synthesis method. (ii) In the 1980s, the use of a single slider control with parameter selection buttons introduced a minimalistic interface. (iii) The 1990s saw a move towards softkeys and displays, with the displays increasing in size and more softkeys being added as the decade progressed.

screens were slow and awkward in their response to touch. The late 1990s and the early 2000s saw three trends:

1. A return to the minimalism of the 1980s for low-cost instruments.
2. A mixture of display-based softkeys and softknobs for mid-range instruments, often organized into functional groupings.
3. Touch screens for high-end instruments. Combined with a small number of dedicated controls for specific functions.

The first color touch screens appeared in the early 2000s, and one expectation is that these will gradually move down to mid-priced instruments.

Front panel design continues to evolve and has moved away from hardware to solutions that increasingly depend almost entirely on software: thus mirroring the developments in synthesis techniques. Software 'front panels' tend to have lots of controls and are almost always in color (although gray predominates) and of course, there is no screen! Despite several attempts to gain broader acceptance, touch screens still remain stubbornly unpopular on computers, even on laptops that could be used as tablet-style computers.

The year 2007 saw the first moves towards touch screen functionality that might change touch on computers, but it came from Apple's iPhone and iTouch MP3 player, as well as several academic research projects and the 'me too' Microsoft Surface tabletop. All of these devices use 'multi-touch' interfaces, where you can touch the controlling surface with more than one finger, and each finger is tracked separately. Trackpads on laptop computers can normally only track one finger at a time.

Multi-touch enables a number of additional ways of interacting with on-screen displays. Pictures can be scaled by touching diagonally opposite corners and moving the fingers away from each other. Rotating the two fingers causes the picture to rotate. Fingers can do different things in different parts of the screen at once, which stops the screen being a one-function display, and turns it into something more gestural and multi-channel.

8.14 MIDI control and MIDI 'Learn'

The mixer and DJ controllers described earlier are used to turn the on-screen controls in digital audio workstations, sequencers and plug-ins into physical controls that can be adjusted in real time. MIDI controllers used for live 'playing' of parameters such as filter cut-off frequency, resonance or oscillator sync have been used for some time, and the MIDI Learn feature is a software aid to make it easy to set up and use controllers by simplifying the mapping of physical controllers to on-screen parameters.

MIDI Learn allows the user to select an on-screen parameter and put it in 'Learn' mode, and then any MIDI controller information that is received by the software will cause that controller to be mapped to that parameter. Without MIDI Learn, then a user would have to know the MIDI controller number for each of the physical sliders or rotary controls and enter them into a mapping table for the parameters in the sequencer or plug-ins. This would require two sets of looking up numbers in device documentation and would be prone to errors as well as time consuming. In contrast, MIDI Learn makes re-mapping controllers to parameters intuitive and fast.

8.15 Advantages and disadvantages

Each type of controller has advantages and disadvantages. On a large scale, the three major controllers are the keyboard, the wind controller and the guitar controller. The keyboard is good for producing complex polyphonic performances based on notes and dynamics, but not so good when expression is required. The wind controller is excellent for detailed monophonic melodies, but not suitable for polyphonic use beyond simple sustained chords. The guitar controller is good for simple polyphonic performances using notes, dynamics and expression, but it has a polyphony limit of 6 notes and has limitations in the complexity of the notes used because of the span of the single hand that is used to fret the notes.

Drum controllers are normally used to provide information on dynamics, triggering and pitch. Each drum pad is normally assigned to a fixed pitch, although this can sometimes be altered by using the velocity to change the pitch. Drum controllers are good for controlling percussion sounds, but the limitations of two hands restrict the polyphony to 2, or 4 notes at once, and therefore the complexity of the note patterns is limited by the handling of the drumsticks.

On a small scale, the individual controllers are the wheels, levers, pressure pads and pedals. Wheels and levers are largely similar, although wheels are much more common than levers on commercial synthesizers. Joysticks seem to be less popular than wheels, at least, manufacturers seem to prefer fitting wheels, joysticks tend to be used for mixing between sounds using a vector mixing approach. Pressure pads are rare and require an artificial separation of the pitch bend into two pads. Pedals suffer from problems of response, the pedal is much heavier than the wheel or the lever, and therefore this tends to limit the speed at which the foot can move it. The control of a pedal with a foot does not seem to have the same degree of precise control as with a hand on a wheel or lever.

After-touch or key pressure has severe limitations as a pitch or modulation controller because it is limited to use only when the key is pressed down. This limits its use and the speed with which notes can be played whilst attempting to use after-touch. One technique, which can be applied when monophonic after-touch is used, is to use the right hand to play the melody notes, and use a finger on the left hand to play a related bass root note or a pedal bass note, whilst also using the same left-hand finger to control the after-touch. Since monophonic after-touch applies to the entire keyboard, this allows expression to be added to the right hand, even though the right hand is apparently moving too quickly to be able to apply after-touch. Breath controllers or foot pedals can be used instead of the left hand to produce similar effects.

8.16 Sequencing

Some software sequencers have provided additional controllers so that the mouse is not the only way of editing parameters. Opcode's Studio Vision allowed qwerty keyboard inputs to control the order of sequences in the playback queue, so that typing 'abacab' would result in sequence 'a' starting, then sequence 'b' starting when sequence 'a' finished and so on. Other control options exploited the keys at the extreme high end of 88-note wide keyboards to start, stop or pause playback of sequences.

Groove boxes or phrase remixers are essentially hardware sequencers with additional controls intended to make live performance of phrase-based music easier. They typically provide control over muting or volume of tracks or parts, tempo and repeats of phrases. This makes the performer very much a conductor of music. But in order to provide this type of control over music, a great deal of preparation needs to be done to produce the loopable phrases split into musically useful tracks or parts. MIDI files are often designed to be played in a conventional sequencer and to sound good with GM instrumentation, and therefore are not always well suited to phrases being looped or even to having parts turned on and off. DJs are good at the opposite end: taking already finished recordings and reusing them in new ways, even though they cannot work with individual parts or adjust the instrumentation. Perhaps what is needed is something nearer to half-way between a sequencer and twin decks rather than the simple emulations of the end points.

8.17 Recording

Controllers can be used to over-dub additional performance information onto an existing track. This allows detailed control over parameters that could be difficult to achieve if attempted at the same time as playing the notes. But it also changes the perception of the recording process.

In a tape recorder, recording happens live, and if something goes wrong, like an unwanted sound or an incorrect note, then you can either record it again, or you might be able to edit it out. For a multi-track tape recorder, the editing option is not practical, and therefore you re-record until it is right or until you are so used to the mistake that it sounds okay.

By allowing recording to happen in more than one take, so that controllers can be added after the notes, the recording process in sequencers has lost some of its liveness and has become a filter that can be used to remove imperfect performance and replace it with perfect performance. The ability to record at one tempo and replay at another allows superhuman precision or speed of playing to be achieved and gives enormous control of the final sound.

8.18 Performing

Using DJ controllers to control two channels of audio, two different software synthesizers or perhaps even two sequencers can provide live performances with a great deal of interaction between the performer/conductor and the sound that is produced. Similarly, mixer controllers can be used to provide immediate control of a large number of parameters throughout a software sequencer, and this can provide detailed control of the final output.

Despite the wide range of possible controllers that are available, it is interesting to note that few of them are parts of musical instruments. It is more usual to have them as performance controllers for pre-recorded music, as with a DJ or a musician controlling a groove box. The history of electronic music is full of experimental alternative controllers, and yet very few of them are in regular use currently, they have not resulted in new musical instruments – the Theremin is one notable exception. The author eagerly awaits a synthesizer that does not have a keyboard, Wind-instrument-style fingering, loop antennas, or a hex pickup, but that becomes popular as an instrument because of the sound it makes and the control that it provides over the sound.

Stephen Kay's KARMA (Kay Algorithmic Real-time Music Architecture) approach to providing comprehensive control over all real-time playing controls arpeggiation, stackings, chordings, hocketing, controller mappings and more, continues to be developed and is now in its second generation (which added features like wave sequencing control). Originally launched in a Korg keyboard called the Karma, it has been built into subsequent Korg instruments such as the Korg OASYS, the Triton and the M3, and it is due to be available as standalone software in 2008. KARMA is the performance equivalent of Reaktor: powerful, deep and well worth the time invested in exploration.

It is still possible for some controllers are so far outside of the expected that they are novel and unusual, and yet useful in performance. For example, many years ago, the author designed and built a foot pedal that outputs MIDI Clock messages, and so is a 'Tempo' pedal. It is very useful in live performance when tempo changes or accompaniment of another performer are required, but it truly excels at making dancing almost impossible!

8.19 Questions

1. Describe some alternative synthesizer controllers to the music keyboard.
2. Give examples of performance parameters being produced by controllers?
3. Compare and contrast a guitar controller, a wind controller and a keyboard.
4. What are the two main types of keyboard controllers?
5. What is the difference between attack velocity and release velocity?
6. What limits the use of after-touch by a performer?
7. What are the differences between a wheel designed for pitch bend and one designed for modulation?

8. Why is it difficult to extract parameters from a guitar to control a synthesizer? Why is it easy to extract parameters from a keyboard to control a synthesizer?
9. What are the limitations of drum controllers when they are used to control a synthesizer?
10. Outline the history of synthesizer front panel controls.

8.20 Timeline

Date	Name	Event	Notes
1969	Robert Moog	Minimoog is launched. Simple, compact monophonic synthesizer intended for live performance use.	Hugely successful, although the learning curve is very steep for many musicians.
1974	Lyricon	The Lyricon was the first commercially available wind-controlled electronic instrument.	
1976	Lol Creme & Kevin Godley	The Gizmotron – a mechanical 'infinite sustain' device for guitars.	A variation on the 'bowing with steel rods' technique.
1977	ARP	ARP Avatar Guitar Synthesizer.	Monophonic synthesizer with Hex pickup (for 'Hex Fuzz').
1977	Roland	GR-500 Guitar Synthesizer.	Hex pickup and string drivers gave 'infinite' sustain.
1979	Realton	Variophon launched – simple electronic wind instrument.	
1980	Roland	GR-300 Guitar Synthesizer.	
1982	Roland	Roland SH-101, a monophonic synthesizer with an add-on handgrip performance-oriented pitch bend and modulation controller.	Notable for its range of color finishes: red, blue and gray
1983	Yamaha	BC1 breath controller launched as add-on for CS01 monosynth.	A small silver box that you grasped with your teeth!
1984	SynthAxe	MIDI-based guitar controller with separate strings, trigger switches and fretboard switches.	Nice design, but very expensive.
1985	Yamaha	FC7 Foot Pedal launched. Uses optical rotation sensor.	
1987	Stepp	DG1 digital guitar – ambitious guitar synthesizer/controller.	
1988	Casio	DH100 Digital Horn launched. Very low-cost wind controller.	
1988	Yamaha	WX11 Wind controller launched. Designed to be used with WT11 synthesizer expander.	Low-cost replacement for the WX7 predecessor.
1991	Yamaha	BC2 Breath Controller launched. Improved version of BC1 built into a headset.	Significantly improved appearance over the BC1.

(*Continued*)

Timeline (*Continued*)

Date	Name	Event	Notes
1995	Roland	GI-10 Guitar-oriented hex pickup pitch-to-MIDI converter.	
2001	Alesis	airFX, 3D-controlled effects unit.	Uses infrared sensors to detect hand position and movement.
2001	Korg	Karma, a combination of a synthesizer with a powerful set of algorithmic time and timbre processing.	Karma 2 added extra facilities and appeared in the OASYS, Triton and M3 instruments, with a stand-alone software version planned for 2008.
2002	Korg	Kaoss Pad KP2, 2D controller and effects unit.	Real-time control over effects.
2007	Yamaha	Tenori-on, a tactile graphical electronic musical instrument.	16 × 16 matrix of lightable switches that acts as a controller and a step sequencer display.

PART 4

Analysis

CHAPTER 9

The Future of Sound-Making

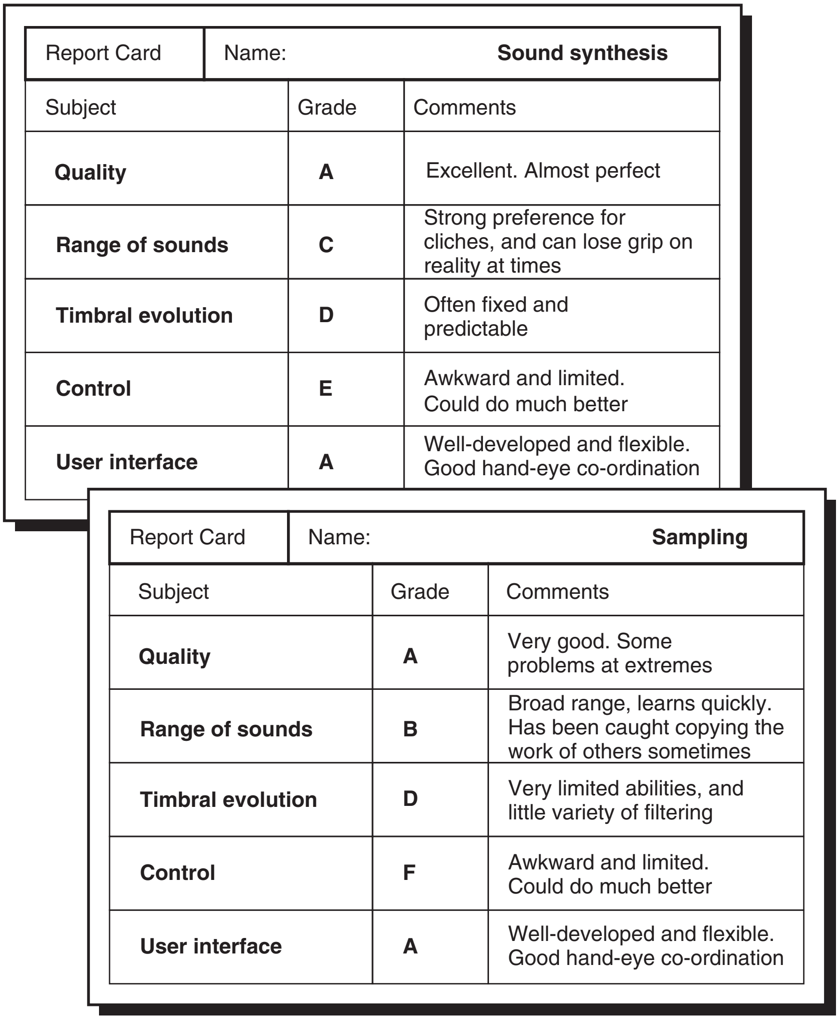
CONTENTS

In some ways, the development of sound synthesis can be viewed as being nearly complete. The technology has now reached the point where the quality of the sounds that can be produced is close to the physical limits of the human ear. Improvements still need to be made to the details of the synthesis techniques, particularly to the parameters that are used to control synthesis: the same problem mentioned in the context of resynthesis. In terms of timbral evolution, there is still scope for new ways of changing the timbre of a sound dynamically to avoid the predictability of low-pass filtering, the metallic textures of frequency modulation (FM), the purity of additive synthesis… Finding viable alternatives to the widely accepted source-modifier model could be one area where there is scope for innovation, but producing intuitive and appropriate control parameters for synthesis techniques such as FM or granular synthesis is not a simple challenge (Figure 9.1).

But the major area that still requires large amount of work is the controls: the interfacing between the performer and the instrument. When this is refined sufficiently, then electronic instruments will truly be able to join their well-developed conventional predecessors.

To appreciate just how complex this problem is, consider the way that sounds are made. The underlying sound-producing mechanism may be straightforward: something vibrating in response to energy input. But the vibrating part may be constrained by barriers or supports that affect the vibration, and those constraints may have resonances that will also affect the vibration. There may be a number of playing techniques that may affect the vibration too, perhaps requiring changes to the input energy to maintain the vibration in some circumstances or various methods for reducing or muting the vibration as a performance effect. There can be a large number of ways of a performer interacting with a musical instrument, and so the vibration might also be affected by the presence of the performer's interaction with the constraints, supports and the vibrating part. And there is always the possibility that someone will invent a new way of interacting.

This decomposition of the sound-making process shows that whilst there may be a well-understood mathematical model for the basic vibration being



Report Card	Name:	**Sound synthesis**
Subject	Grade	Comments
Quality	**A**	Excellent. Almost perfect
Range of sounds	**C**	Strong preference for cliches, and can lose grip on reality at times
Timbral evolution	**D**	Often fixed and predictable
Control	**E**	Awkward and limited. Could do much better
User interface	**A**	Well-developed and flexible. Good hand-eye co-ordination

Report Card	Name:	**Sampling**
Subject	Grade	Comments
Quality	**A**	Very good. Some problems at extremes
Range of sounds	**B**	Broad range, learns quickly. Has been caught copying the work of others sometimes
Timbral evolution	**D**	Very limited abilities, and little variety of filtering
Control	**F**	Awkward and limited. Could do much better
User interface	**A**	Well-developed and flexible. Good hand-eye co-ordination

FIGURE 9.1

Chapter 9 in summary form.

used, each of the additional layers of detail adds to the complexity of the model and to the difficulty of providing an appropriate controller for use in performance. Adding refinements to the mathematical model to deal with additional performance techniques also requires detailed knowledge of the instrument and how it is played and an understanding of how and why the technique works.

Perhaps the theremin provides a glimpse of what is possible, as well as a warning of just how complex the problem is. As a simple monophonic instrument with one simple control for pitch and one simple control for volume, the Theremin might appear to be easy to play, and yet the exact opposite is true.

Digital technology enables software models of real and imaginary musical instruments to be combined dynamically with purely synthetic, mathematically derived instruments and with high-quality samples of real and unreal instruments, with almost no foreseeable limit. The future lies in combining the control, sequencing, recording, composition, performance and arrangement of these sound-making tools into an integrated whole where the composer or the performer is an active and controlling part of the sound-making system.

9.1 Closing the circle

It is interesting to return to the opening chapter of this book and to consider some of the other types of synthesis and their 'completeness':

- Three-dimensional scene synthesis, or rendering, has now reached a level of sophistication where true photorealistic scenes are almost possible and where convincing human beings are tantalizingly close. Advances in this area are likely to reach the limits of human perception in the next 10 or 20 years, if it follows the same timeline as sound synthesis. But the area that requires work is how to control rendering and animation. In particular, animated and rendered human beings are still some way from being mistaken for real actors. There is even evidence that as the synthetic human beings become very realistic in appearance, the perception of real human viewers becomes increasingly critical. This is very different to the way that the human ear is quite willing to accept sampled instruments as real, or even poorly synthesized sounds as acceptable in a musical context.
- Speech synthesis continues to make incremental advances, but as with sound synthesis, the main limiting factor now is the control mechanisms, not the synthesis technology. This can readily be shown by using a technique very similar to resynthesis. If a spoken sentence is analyzed to extract suitable control values, then these values can be used to reproduce the spoken sentence with a high degree of perceived quality. But try to take the same sentence in plain ASCII text form and produce the same effect by creating the control values from the text and the result tends to sound synthetic.
- Word processors and authors are still some way from completeness, and this author expects large amount of interfacing work to be required for the foreseeable future. The control interface is not perfect, but it suffices and is in wide use.

The long-term problem of synthesis thus seems to be one of control and interfacing, rather than one of the generation technologies. It can be well illustrated with a simple analogy: this author is perfectly capable of renting a high-performance sports car, but is totally incapable of driving it with sufficient skill to impress anyone.

9.2 Control

Despite the best efforts of some advertising hyperbole, the future of synthesis does not depend on acronyms, larger read-only memory (ROM) samples containing even more sample sets trying not to feature cliched sounds, or even networked compact disk (CD)-ROMs, DVD-ROMs or Blu-Ray disks (BD) of pre-packaged sounds. At the end of the twentieth century, it seemed to lie with controllable flexible synthesis, and mathematically based modeling techniques seemed to be in pole position, although the resulting synthesizer seemed likely to incorporate multiple methods of control and synthesis combined using software rather than the instrument simulation, which the early implementations concentrated on. Just as all-digital synthesis has been used to emulate even analogue synthesis, advanced modeling techniques will be able to produce sounds that combine the best of real, analogue and digital instruments. As the twenty-first century has moved to the end of the first decade, the future seems to lie with more than just synthesis on its own. In fact, the synthesizer has increasingly become a part of a larger music-making system, and this may be the true future of synthesis: as a component part of a highly capable and flexible sound-making technology.

The current split into solo and accompaniment instruments actually makes a great deal of sense when viewed in the context of an overall role of making musical sounds, since many real-world instruments naturally divide into monophonic solo instruments and polyphonic accompaniment ones. This is especially true of the control interfaces that are used by the performer: accompaniment instruments seem to be naturally limited in their expressive capabilities in comparison with solo instruments. But as synthesis becomes incorporated into equipment that is concerned with producing more than just audio, these divisions will become different parts within a much larger framework, and the individual differences will become less relevant to the end user.

Computer processor development has now produced powerful and widely available general-purpose chips as used in personal computers (PCs), as well as more dedicated digital signal processor (DSP) chips. Recent general-purpose microprocessors have also started to add performance enhancements (Intel's MMX instruction set is one example) that are similar to those used in DSPs. The start of the twenty-first century has seen the rise of processing chips with multiple 'cores', which are effectively several separate microprocessors put onto one chip, and this is increasing the available processing power without requiring the race for ever-increasing clock speed that happened in the twentieth century.

This increasing use of cores could be very significant to future computer design and the way that those computers are used. It may be that the processor of the future is not a composite machine which does some processing on-chip and the numerically intensive calculations on-DSP. Instead it could be one where cores are used to suit different purposes, perhaps with the ultimate being a reconfigurable system of cores. In the future, processor capability may be limited only by the number of cores and associated memory, and not by arbitrary physical limitations like processing speed or specialized command sets. The ideal end result will be a 'multimedia' computer that has the processing power and the flexibility to suit various tasks with diverse needs, and that matches the requirements of sound-making more by convergence than by design.

Processors that are used for producing sounds can also be used to process sounds, and so sound synthesis and digital recording will continue to converge. Many PC-based music creating systems now provide a completely digital path from the creation of the audio through a software synthesis, modeling or sample replay, through sequencing, arranging, mixing and effects and final output as multi-channel digital audio. As storage prices continue to fall, these 'hard disk' music recording systems or Digital Audio Workstations (DAW) will become more and more attractive and will continue the move from the top end of the market to the lower end. The use of the term 'hard disk' recording may also disappear as flash memory continues to displace mechanical hard disk storage. Stand-alone music recording systems will provide increasingly powerful tools for those who do not wish to build a PC-based system, but these may not have the same rapid development cycles.

MIDI seemed to reach a stable point where its capabilities were adequate for the requirements of most of its users in the last decade of the past century, whilst more recent digital audio networking technologies such as mLAN and MADI have only had a limited effect on enabling a similar integration for digital audio. MIDI was powerful and ubiquitous enabling standard whilst the abstraction between notes and sounds was forced by low-bandwidth serial digital connections, but is now less important in a world where the separation of synthesizers, mixers, effects and sequencing of these elements is no longer necessary. In many current computer-based systems, MIDI is almost entirely hidden within a USB connection and is used primarily as a way of connecting performance controllers, or is used internally as part of the MIDI sequencing and virtual plug-in instrument messaging.

The bringing together of all the software from synthesizers, samplers, mixers, effects and sequencing is increasingly happening at the basic 'entry' level, with the result that sophisticated music creation capability is becoming a mass-market commodity, rather than something which requires complex customization. This should allow live performance instruments to become more responsive to the playing skill of the user.

The performance-oriented keyboards of the end of the twentieth century incorporated features such as split keyboards, phrase recorders and arpeggiators, with the Korg Karma adding a great depth of sophistication to the way that these could be controlled and interacted with by a performer, and this functionality has become part of the palette of features that are expected in a high end performance instrument. A wider set of performance controllers that do not depend on the limited control capabilities of the music keyboard are also beginning to appear, driven by the increasingly flexible facilities offered by sequencers and host software. There have been some indications of price drops in this area of software, and over time, this type of functionality is unlikely to remain at a price where only professional musicians can utilize it.

At some time before the middle of the twenty-first century, the synthesizer will have changed into something much more like the 'orchestra in a box' of public perception. In fact, it is likely to be much more than that, since the audio will be part of a larger 'multimedia' facility which can work with pictures and moving images with the same ease as audio. This brings together all of the types of synthesis that can be imagined, and some that may not be expected. Software 'expert' systems are likely to be able to assist with much of the work of composing and arranging in the widest sense, which means that the role of the end user may well become more akin to that of a combination of performer and director. The creative input from a human being at the beginning and the end of making music (and associated moving pictures) will still be essential – but the processes in between may well be largely devolved to computers.

The 'synthesizer' will thus no longer be a simple musical instrument that requires large amount of additional equipment to produce music but will be a smaller and much more capable creative assistant. It may even be so different in appearance that many of the musicians of the twentieth century would not recognize it – or they might mistake it for a laptop computer.

9.3 Commercial imperatives

In order to understand why a future synthesizer could be so very different, the role of the synthesizer in terms of both making music and making money needs to be considered.

Fundamentally, musicians want to make sounds. Making sounds can be accomplished using a number of technologies, and sound synthesis has typically used electronic means to produce sounds. As science and technology have developed, the methods that are available to designers have changed, and this has, in turn, influenced the way that synthesizers have evolved.

Synthesizers in the 1950s used analogue circuitry because it was cheap, widely available and well understood, whereas digital technology was new and expensive. Subtractive synthesis is a cost-effective solution to the problem of producing lots of sounds with a minimum of complexity, at least for monophonic sounds, and so it became widely used. The music keyboard was a cheap, widely available and well-understood piece of organ technology, and thus was used for the playing interface of the first synthesizers. To keep the cost down, monophonic instruments were designed, and by the1970s they were well suited to produce melody lines that suited the progressive rock music of the time, as well as having timbres that contrasted well with the electric guitars, organs, string machines and electric pianos of the time, whilst polyphony or modular instruments were rare and expensive. As digital technology began to become suitable for tasks such as scanning keyboards and providing sets of control voltages, microprocessors were used to add functionality to analogue subtractive synthesis, but the sound generation was still analogue, and polyphony was produced by using multiple monophonic synthesizers.

In the 1980s, FM provided the first cost-effective way to use digital circuitry to produce sounds, since it requires a small ROM containing a numerical representation of a sine wave, and some numerical processing. By using that same circuitry repeatedly for each part of the FM generation process, and for several notes, generating FM sounds in real time became possible with low-cost digital chips. FM also provided larger polyphony than analogue synthesizers, and the polyphonic analogue synthesizer faded away into an item for collectors only. Samples and synthesis (S&S) continued this theme of making the most of available resources by maximizing the use of ROM or random access memory (RAM) with short-looped samples, and then processing those samples initially through analogue filters and enveloping, but later through digital filters, enveloping and effects. As the price of memory fell, it became less and less of an expense, and so the number, sophistication and quality of the on-board sound samples increased over time.

With very low-cost storage, sample replay provided large ranges of sound without great expense, and in such a device, the savings produced by synthesis were not really required any longer. Since the majority of users required just the sounds, boxes with keyboards and sounds from memory chips are what economic pressure will produce. Once you have the basic processing power in place, then it is easy to add in additional related functionality, hence the addition of sequencing, mixing and effects to a synthesizer to produce a workstation. The end result would seem to be workstations with ever-increasing numbers of sounds, but this provides little scope for differentiation of products, since every workstation can produce the same sounds. Software synthesis seems to have solved this problem, driven partly by fashion, and partly by the ease of making changes in real time. Opening up synthesis to ordinary computer owners (rather than academics) seems to have produced a lot of people who are now producing sounds (and samples!).

There are many anecdotes about synthesizers arriving in service centers with the factory presets still intact. This is sometimes used as an indication that 'very few people actually programmed synthesizers'. An alternative interpretation is that people who spent hours creating sounds would not leave the results of their hard work in the synthesizer when sending it away for servicing.

The start of the twenty-first century brought with it an increasing emphasis on software synthesis rather than sample replay, particularly for software emulation of older 'retro' techniques such as analogue synthesis. So the predominant technology used to produce sounds from keyboard instruments has become an increasingly flexible synthesis architecture that can produce most of the methods of synthesis in software as well as sample replay-oriented S&S technology, but with an increasing emphasis on the user-controllable 'synthesis' processing part. It could be argued that this type of instrument is almost not a synthesizer at all, but merely a sophisticated sample replay device. Perhaps it is because it has simplified and abstracted editing controls to ease the task of making changes to the sounds. Or it might be because the designer's expectation is that the majority of the users will never carry out anything more than minor edits to the sounds, often by adjusting parameters such as filter cutoff in real time. The commercial drive to produce instruments that meet the needs of the playing musician is unlikely to do anything other than continue along this path of increasingly sophisticated synthesis and replay capability for live performance usage.

However, whilst the instrument used on stage to produce sounds might not be a true synthesizer any longer, this is not the case for software synthesis. The software sequencer has expanded to encompass both the sound generation and the sound mixing extremes of its operating environment. Generating sounds in software has limitations set by the budget of the end user, not the synthesizer designer, and so the resulting synthesizer 'plug-ins' can use esoteric and unusual synthesis techniques whose complexity and processing power requirements would prohibit their use in a mass-market keyboard instrument. The future of synthesis is thus not in the keyboard instruments that are actually workstations with sequencers and effects, but instead, it lies with software synthesis engines that are themselves components within software-based 'studios'.

It is very tempting to think that it is hard to see where the limits to this software synthesis approach lie, but this is always the case with the future. Apart from the very near-term, the future is rarely a simple extrapolation of now. It is often something unexpected and so ignored, unusual and so derided, or overlooked and dismissed, and is often something discarded or abandoned by the mainstream. As a first example, consider the pinnacle of analogue synthesis techniques. The ability to draw waveforms on acetate sheet might have seemed to offer complete freedom of sound generation, but it was not until the Fairlight CMI gave them the ability to do this, digitally, that the limitations of this method of controlling sound became apparent, and by then, FM and sample replay were in the ascendant. For a second example, look at how physical modeling seemed to be poised to replace sample replay, but analogue modeling and a retro fashion were the short-term winners when the complexities of the user programming of physical modeling became apparent.

The laptop computer controlled by a non-keyboard controller could well become the early twenty-first century's preferred device for synthesizing sound. Whilst this may already be true for many musicians, the general public still think first of music as keyboards or the electric guitar, or for the younger generation, that relative newcomer, the DJ with twin turntables. This synonymity of the DJ with music creation rather than music replay is new, and reflects the rise in importance of performance ability. All that is now needed is for a user interface to make the laptop the next synonym for music-making, and another step in the creation of the future has been taken. What might that interface be? One possibility is the multi-touch interface that has started to appear on some MP3 players, mobile phones/cellphones, track pads and projection tables. Another possibility is some sort of wind/string hybrid that uses the powerful performance controls that are possible with a guitar combined with the extra control from blowing. With hindsight, future readers might well be reading this chapter and placing the Yamaha/Toshio Iwai Tenori-On as the predecessor of multi-touch–controlled laptop computer music creators. Of course, with alternative hindsight, readers in a different future are now laughing at just how wrong this prediction is and turning back to making music with their…

Each edition of this book has required extensive rewriting of this section, and this is no exception. Future editions of this book will almost certainly be very different in their content, as well!

9.4 Questions

1. Why is the subtractive analogue synthesis 'source-modifier' model so successful, and so pervasive?
2. Take a musical sound (guitar, piano, flute, etc.) and break it down into successively deeper levels of the sound creation. Take the primary sound generation method first, and look at the ways that the sound might be varied by a performer. Then look at the physical environment of the instrument itself and look at how it shapes the sound. Then look at different performance techniques and how these can affect the sound. Continue as deep as you can with the analysis of what gives the instrument its characteristic sound, and then estimate how much of the true mechanism of that instrument's sound generation you have covered.
3. Can you think of any alternative electronic musical controllers to the conventional keys, ribbons, wheels, velocity and pressure?
4. At what point do auto-assist features such as arpeggiation, auto-chords, auto-bassline, rhythm patterns, patch randomization, stacks, effect chains and loops turn a musical instrument into something that is no longer fully under the creative control of the performer?
5. Is physical control technology or software synthesis technology going to be the main feature of the next generation of electronic musical instruments?
6. When each method of sound synthesis reached its peak, was there any indication that this peak had been reached? What indications are there for any technology that it is about to become replaced with an alternative? If you go back in time to a world where analogue subtractive synthesis is at its peak, would you know that it was about to be sidelined for a time by hybrid and digital approaches?
7. Consider Figure 9.4.1, where on the 'S-curve' is current 'state-of-the-art' software synthesis? Is software synthesis the final, ultimate method of sound synthesis? What might replace it?
8. If a synthesizer allows a performer to make large changes to the timbre of a sound during performance, then how can this be captured in notation, and is it a performer, conductor or arranger level of control?
9. Can new technology allow old techniques to be revisited? Suggest one possibility.
10. It has long been asserted that there are many more people who would like to be able to play a musical instrument than who actually do. Is technology a way to make sound-making more accessible? In what ways would you try to make sound-making easier? Would this enhance or devalue the art and craft of people who make music without such assistance?

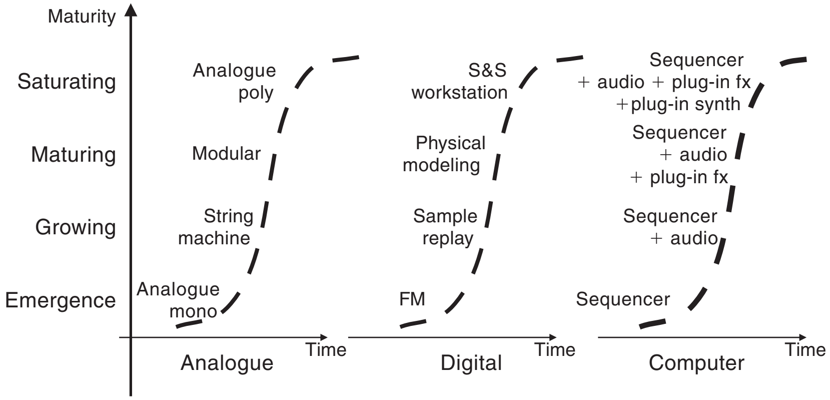


FIGURE 9.4.1 Technologies often see development progress to a point where a new approach can quickly replace them. This 'S-curve' behavior is seen in many fields, and is here tentatively applied to three synthesis-relevant developments. Note that this is heavily idealized: the techniques actually overlap in time.

9.5 Timeline

Date	Name	Event	Notes
2010	Vocalit 4 released	The industry-standard synthetic vocal performance software gets a major rework.	A Vocalit performance was in the top 10 of the charts at the time of the release.
2012	German Synthetic Guitar Systems	The GSGS BTAG synthetic guitar is launched to critical acclaim.	The BTAG stands for 'Better than a guitar', and it gets rave reviews.
2014	Advanced Playing Technology	Band4U 37 is released.	B4U has already had 4 number 1 hits in the last year.
2015	Pippin Music Systems	Mars Cube released.	15 cube sound generation workstation with virtual keyboard, modeled orchestra library and IET-772 storage interface.
2017	The Supergroup Group	Recreate 42 modeled performers from the 1960s, 1970s and 1980s.	Triggers a 50 years ago retro boom.
2020	Synxola	Radical new instrument is released, with a sophisticated user interface not unlike a saxophone.	The Synxola is a synthesizer disguised as a physical musical instrument.

Bibliography

The first edition of this book contained references to books in this section, and these are repeated and updated here. But the internet has changed the way that many people look for reference material, and my advice now would be that readers of this book also consider using the many World-Wide Web (WWW) resources available on the Internet as an alternative to printed books. Manufacturer information is easily available on the Internet, and has formed the major reference source for the second edition of this book.

Books

Bogdanov Vladimir, Chris Woodstra, Stephen Thomas Erlewine, John Bush (eds). All Music Guide to Electronica: The Definitive Guide to Electronic Music (AMG All Music Guide Series). Backbeat Books, 2001.ISBN 0879306289

Buick Peter, Lennard Vic. Music Technology Reference Book. PC Publishing (technology), 1995.

Clayton George B. Experiments with Operational Amplifiers. MacMillan (hybrid), 1975.

Capel Vivian. Audio and Hi-Fi Engineers Pocket Book. Butterworth-Heinemann (Newnes) (technology), 1988.

Cary Tristram. Illustrated Compendium of Musical Technology. Faber and Faber (technology), 1992.

Chamberlin Hal. Musical Applications of Microprocessors. Hayden Books (digital), 1987.

Chowning John, Bristow David. FM Theory and Applications or Musicians. Yamaha Music Foundation (digital), 1986.

Colbeck, Julian (1985) Keyfax 1, 2, 3, 4, 5, ... (ongoing) Virgin/Music Maker/ Making Music (general).

Cook Perry R. Real Sound Synthesis for Interactive Applications, ISBN: 1568811683 (physical modeling). A K Peters Ltd, 2002.

De Furia Steve. The Secrets of Analog and Digital Synthesis. Ferro Productions/ Hal Leonard Books (analogue, digital), 1986.

Forrest Peter. The AZ of Analogue Synthesizers. Susurreal Publishing (analogue), 1994.

Kettlewell Ben, Electronic Music Pioneers, ArtistPro.com, 2001 ISBN 1931140170

Lee Lara, Reynolds Simon, Shapiro Peter (eds). Modulations:A History of Electronic Music: Throbbing Words on Sound. Distributed Art Publishers, 2000.ISBN 189102406X

Mellor David. How to set up a Home Recording Studio. PC Publishing (general), 1992.

Newcomb Martin. Guide to the Museum of Synthesizer Technology. The Museum of Synthesizer Technology (analogue), 1994.

Pellman Samuel. An Introduction to the Creation of Electro-acoustic Music. Wadsworth (general), 1994.

Pierce John R. The Science of Musical Sound. Scientific American/Freeman (general), 1992.

Prendergast, Mark, The Ambient Century: From Mahler to Trance: The Evolution of Sound in the Electronic Age, Bloomsbury, 2001 ISBN 0747542139, ISBN 1582341346 (hardcover eds.) ISBN 1582343233 (paper)

Reynolds, Simon, Energy Flash: a Journey Through Rave Music and Dance Culture (UK title, Pan Macmillan, 1998, ISBN 0330350560), also released in US as, Generation Ecstasy : Into the World of Techno and Rave Culture (US title, Routledge, 1999, ISBN 0415923735) (If only this book contained technical details of how the music described was produced!)

Roland Corporation. A Foundation for Electronic Music. Roland Corporation (analogue), 1978.

Roland Corporation. Practical Synthesis for Electronic Music. Roland Corporation (analogue), 1979.

Rumsey Francis. MIDI Systems and Control. Focal Press (digital), 1994.

Schaefer, John, New Sounds: A Listener's Guide to New Music HarperCollins, ISBN 0060970812, 1987.

Sicko Dan. Techno Rebels: The Renegades of Electronic Funk, ISBN 0823084280. Billboard Books, 1999.

Tooley Michael. Computer Engineers Pocket Book. Butterworth-Heinemann (Newnes) (digital), 1987.

Vail Mark. Vintage Synthesizers. GPI/Miller Freeman (general), 1993.

Watkinson John. The Art of Digital Audio. Focal Press (digital), 1988.

Some of the more obscure or 'out of print' books above are included because they may either be located in libraries, or may become available via a republishing facsimile. I also thoroughly recommend the Focal Press audio books, of which this book is just one example!

Magazines and Journals

Greenwald, Ted (magazine article) Samplers laid bare. Keyboard Magazine, March 1989 (digital).

The Computer Music Journal (quarterly journal) The MIT Press, Fitzroy House, 11 Chenies Street, London WC1E 7ET. Telephone 071 306 0603 Fax071 306 0604 (history, analogue, digital, sampling).

Sound On Sound (monthly magazine) SOS Publications Limited, Media House, Trafalgar Way, Bar Hill, Cambridge CB3 8SQ, UK. Email: sos@soundonsound.com Telephone: +44 (0)1954 789888 Fax: +44 (0)1954 789895, February 1996 (history, analogue, digital, sampling).

Keefe, Douglas H. (journal paper) Physical Modelling of Wind Instruments. Computer Music Journal, Vol 16, Number 4, Winter 1992 (digital).

Karplus, K. and A. Strong, (journal paper) Digital. Synthesis of Plucked String and Drum Timbres. Computer Music Journal, Vol. 7, No. 2, Summer 1983 (digital).

Cook, Perry R. (journal paper) SPASM, a Real-Time Vocal Tract Physical Model Controller. Computer Music Journal, Vol. 17, No. 1, Spring 1993 (digital).

Sullivan, Charles R. (journal paper) Extending the KarplusStrong Algorithm to Synthesise Electric Guitar Timbres with Distortion and Feedback. Computer Music Journal, Vol. 14, No. 3, Fall 1990 (digital).

Julius O. Smith III (online book) Physical Audio Signal Processing for Virtual Musical Instruments and Audio Effects, Center for Computer Research in Music and Acoustics (CCRMA) Department of Music, Stanford University, Stanford, California 94305 USA, May 2008.

Julius O. Smith III (journal paper) Physical Modelling using Digital Waveguides. Computer Music Journal, Vol. 16, No. 4, Winter 1992 (digital).

White, Paul (magazine article) General MIDI. Sound On Sound, August 1993 (using synthesis).

ZIPI (journal papers) Computer Music Journal, Vol. 18, No. 4 (Winter 1994).

Other sources

I have deliberately avoided using Internet Uniform Resource Locators (URLs) to indicate links to web-pages in this book. URLs are often long and complex, which results in typing errors, and the nature of the Internet itself means that links in books (or on web-pages) frequently point to web-pages which no longer exist.

Instead, I recommend that readers use a popular search engine and adopt a cautious analytical attitude when reading the search results. I do not recommend a search engine because their long-term longevity is uncertain. At the time of writing the first edition of this book, then Alta Vista was the clear leader. By the time the second edition began to be considered, Alta Vista had seen a change of corporate owner, and Google had become very widely used. Google's technology produces results which are typically of high quality and 'on topic', but it is just as prone to being replaced if and when a better solution is found.

(The reader might like to consider how this ongoing competitive evolution of web-browsers mirrors (or otherwise) the development of sound synthesis and sampling technology.)

The mental approach to dealing with web-page contents is almost more important than the source of the links. Whereas most books, magazines and journals have been through a formal publishing process, the informality and ease of creating a web-page means that the validity, correctness, lack of bias and truthfullness can all be called into question. Cross-referencing facts is a good tool for checking that web sources of information at least corroborate each other, although checking that one is not a slighlty edited copy of the information presented by the other is also a good check. In general, specialist web-sites which have been recently (and frequently) updated, and which can be reached via links from more general web-sites, and which have high standards of grammar and spelling, are often good candidates.

The best technique for finding information about book that you intend to purchase is to use an on-line selling site which has large volume, broad choice, and provides sophisticated methods for allowing purchasers to feed back their comments to other prospective purchasers. At the time of writing, Amazon was arguably the largest and best organized of the on-line retailers, although there were also a number of dedicated on-line book-only retailers with similar recommendation systems. Peer review of books, magazine articles and journals can provide a way of avoiding the book which appears to be perfect, right up until the moment you've paid for it and start reading it properly.

Jargon

Jargon

Specialist	Used only in a given subject
Particular	Words used in one context
Subject	Synthesizers is the current subject
Alternative	Jargon often replaces more common words
Acronym	Jargon frequently uses acronyms

Jargon is a descriptive term for those specialist words which are used only in a particular subject. This entry shows how they are dealt with here. Look up the word which is used throughout this book, and the alternatives will be shown. This is somewhat like a thesaurus, but for jargon. The entries include both synonyms and antonyms.

After-touch

Touch	Additional key pressure used as a controller
Pressure	Alternative name for touch
Sustain pressure	After-touch is often used in the sustain segment
Second touch	Alternative name for touch (initial touch = velocity)
Expression	Alternative name for touch
After-touch	Alternative name for touch

After-touch is the name for the additional pressure that some key-boards allow to be used as a controller after the key has travelled from the up to the down position. After-touch sensors vary from soft rubber, which provides several millimetres of movement, to ones which are hard rubber and hardly move at all. After-touch sensing can be global for the entire keyboard (the hardest pressure is normally the output) or polyphonic (much rarer).

Amplitude

Level	Alternative name. How big the signal is?
Volume	Alternative name. How big the signal is?
Size	Alternative name. How big the signal is?
Peak	The highest amplitude reached by the waveform
Normalised	Using all of the available volume range
Compressed	Increasing the volume of quiet audio, reducing loud
Dynamics	The ratio between the loudest and quietest sounds
Loudness	Alternative name. How big (loud) the sound is?
Loudness	Increasing the bass equalisation at low volumes

The amplitude of a sound is how big the waveform of the signal is. The volume of a sound is how loud it is. This is related to the amount of power which is required to move the air to produce that sound.

Detent

Dead band	The centre of the control has an area where nothing happens
Dead zone	Alternative name for dead band
Detent	A physical click at the centre
Notch	Alternative name for detent
Click	Alternative name for detent
Stop	Alternative name for detent
Spring	Springing to return to the detented position

Pitch wheels and levers need to have a way for the performer to know when the pitch bend is centred: no pitch bend. This can be achieved by using a dead band where moving the control has no effect, or a detent for the 'no pitch bend' position, or a combination of springs and a dead band or detent.

Envelope

Contour	Alternative name for an envelope
ADSR	Attack decay sustain release, the commonest form
ADR	Attack decay release, the old Moog-style of envelope
AD	Attack decay, the percussive envelopes only
EG	Abbreviation for envelope generator
Segment	One part of an envelope: attack is the first segment
Stage	Alternative name for segment
Slope	The rate of change of the envelope during a segment
Rate	The rate of change of the envelope during a segment
Time	The length of time of a segment
Level	The level at the start of a segment (and the sustain level)
Break-point	Indicates a change of slope at a specific time or level

Pivot	Alternative name for break-point
Trapezoid	Function generator sometimes used as an envelope
Follower	Extracts an envelope from an audio signal
Scaling	Changes in rates or levels

Envelopes are one of the major sources of control voltages and control signals in a synthesizer. Whereas most other control sources are cyclic (low frequency and voltage controlled oscillators (LFOs and VCOs, respectively)), the envelope can produce complex shapes which happen only once (although some envelopes do allow looping between segments). Envelope terminology can vary considerably between instruments. The method used to indicate the time of a segment may be a time, or a slope, and may use low numbers to indicate a short segment, or high numbers.

Harmonics

Partial	A single frequency component (could be an inharmonic)
Overtone	First overtone is second harmonic
Line	A single frequency component (could be an inharmonic)
Spectral line	One frequency in the spectrum (could be an inharmonic)
Component	One frequency in the spectrum (could be an inharmonic)
Frequency	A single spectral component (could be an inharmonic)
Harmonic	Related to the fundamental
Inharmonic	Not related to the fundamental

Harmonics are based around multiples of the fundamental frequency. The 2nd, 4th and 8th harmonics are octaves, whilst the 3rd harmonic is a perfect 5th. Inharmonics are frequencies which are not harmonically related to the fundamental. On a spectrum, individual frequencies appear as vertical lines or peaks.

Layer

Split	Layering of two sounds controlled from different notes
Double	Two sounds playing the same notes
Stack	Two sounds playing the same notes
Layer	Two sounds playing the same notes
Hyperpreset	The setup of two or more sounds in a layered form
Setup	Alternative name for hyperpreset
Performance	Alternative name for hyperpreset
Multi	Alternative name for hyperpreset
Combination	Alternative name for hyperpreset
Combi	Shortened form of combination
Function	Alternative name for hyperpreset

A layer is two or more sounds playing the same notes, but often arranged so that the two sounds are complementary rather than very similar (unless detuning is the purpose of the layering). A split allows two timbres to be layered and to be controlled independently by allocating them to different areas of the keyboard. Layering thus requires only one-handed playing or single notes, whilst splits require two hands or two notes. The setup of two or more voices has many names. Some synthesizers enable the complete setup of the synthesizer to be stored as a hyperpreset.

Memory

Program	Used by MIDI. Computer programming term
Patch	Analogue modular synthesizers
Voice	Yamaha
Preset	Usually permanently stored in ROM memory
Sound	The real world?
Tone	Electronic organs?
Waveform	Electronics term
Wave	Electronics slang
Combi	More than one sound layered or stacked
Performance	More than one sound layered or stacked
Stack	More than one sound layered or stacked
Split	More than one sound across the keyboard range
Timbre	Musical term for sound quality
Store	Computer term
Location	Computer programming term
Bank	Several sounds
Set	Alternative name for several sounds

Synthesizers store sounds in ‘memories’. There are many alternative names for the memories. The distinction between a sound and a combination of several sounds is becoming blurred. Many sounds are made up of simpler sub-sounds, parts or elements.

Messages

Data	The information contained in the message
Signals	Alternative name for messages
Packets	Alternative name for messages
Commands	Alternative name for messages
Streams	Used for several messages in sequence
Information	The data in the message

MIDI messages are sent in the form of one or more bytes in sequence.

MIDI

MIDI	Musical instrument digital interface
ZIPI	Superseded 1994 extension proposal from CNMAT, Berkeley
GM	General MIDI, formalisation of sound and drum mapping
GS-MIDI	Roland control extensions (similar to XG-MIDI)
XG-MIDI	Yamaha control extensions (similar to GS-MIDI)
Mapping	Allocation of sounds and drums to MIDI channels
mLAN	Yamaha's Firewire (IEE 1394) protocol for audio and MIDI
DMIDI	(IEE 1639) wrapper for carrying MIDI over Ethernet LANs
In	The input of a MIDI device (a sink)
Thru	A repeat of the input to a MIDI-in, presented as an output
Out	The output of a MIDI device (a source)
Channel	MIDI provides 16 separate channels for device information
Sysex	Provision within MIDI for specialized device information

MIDI has its roots in 8-bit computer hardware from the early 1980s. It has become almost ubiquitous for all but the purest analogue synthesizers. Modern revisions are extending the life of MIDI into the 21st century by adapting it for use on a variety of networks.

Pulse width

Width	Duration in time of mark portion of a pulse waveform
Duty cycle	US term for pulse width
Shape	Used for non-rectangular waveforms: sawtooth, sine, etc.
PW	Abbreviation for pulse width
PWM	Pulse width modulation: cyclic change of pulse width
Skew	Used for non-rectangular waveforms: sawtooth, sine, etc.
Per cent	Alternative measure of PW: 50% = square, 1% = narrow
Ratio	Alternative measure of PW: 1:1 = square, 99:1 = narrow
Rectangle	Alternative term for pulse width
Symmetry	Alternative term for pulse width

The pulse width of a waveform is a measure of the duration of a specific state. Usually it refers to a rectangular waveform, and is then a measure of the length of the mark portion of the waveform. Pulse waveforms are made up of two parts: the mark (usually the most positive portion) and the space (usually the most negative portion). When the mark and the space times are equal, then the waveform is said to be 'square' in shape.

Resonance

Q	The sharpness or selectivity of a filter
Feedback	The signal which is feedbacked from the filter output to the input

Emphasis	Alternative name for Q
Quality	The full name for Q
Regeneration	Alternative name for feedback
Sharpness	Alternative name for resonance

Low-pass filters are good for making tonal changes to the brightness of a sound, but for making more major changes a band-pass filter is required. By adding feedback to a low-pass filter, it is possible to turn it into a resonant band-pass filter which has a strong peak at the cut-off frequency.

Sink

Destination	The input to the circuit = destination of control signal/voltage
Input	The input of the circuit
In	Abbreviation for input
Sink	Alternative name for destination

Controls can be divided into sources and destinations. Destinations take control voltages and signals and use them to make changes to parameters. Matrix modulation schemes use the words source and destination for the outputs and inputs to a matrix of connection points.

Source

Output	The source of the control voltage = output of circuit
Out	Abbreviation for output
Source	The output of the circuit
Origin	Alternative name for source

Controls can be divided into sources and destinations. Sources provide control voltages and control signals. Matrix modulation schemes use the words source and destination for the outputs and inputs to a matrix of connection points.

Storage

Disk	Hard or floppy: rotating magnetic/optical data memory
Disc	Vinyl record or album: holds audio information
RAM	Random-access memory: R/W solid-state data chips
ROM	Read-only memory: permanent solid-state data chips
Media	The physical carrier/holder of the data: disk, tape, card, etc.
Floppy	Flexible circle of magnetic material in a rigid casing
DAT	Tape-based data media using rotating helical scan heads

Card	Generic term for memory card (using ROM or RAM)
PCMCIA	Specific type of standardized card and interface
Network	A means of connecting computers together: non-local
Server	A remote computer used to hold data, accessed via a network
Optical	Disk using optical rather than magnetic storage for data
Virtual	RAM used as a hard disk replacement (faster)
Tape	Flexible plastic strip with magnetic coating for data
Off-line	Offline storage needs to be physically loaded by hand
Flash	Rewriteable memory technology used in memory cards
Removable	Opposite to built-in (memory cards are removable)
Built-in	Flash, ROM or RAM memory inside the device.
CD/DVD	Compact disk/digital versatile disk, optical storage

Computers use many forms of storage for holding data (information). Storage is really a shortened form of the compound word 'mass storage'.

Vibrato

FM	Frequency modulation: cyclic pitch changes
AM	Amplitude modulation: cyclic volume changes (not vibrato)
Tremolo	Cyclic changes in volume (not vibrato)
Flutter	Rapid random changes in pitch (usually on a tape recorder)
Wow	Slow cyclic pitch changes (usually on a tape recorder)
Wobble	Cyclic pitch changes
Modulation	Can be any pitch, volume or timbre change, often vibrato
Leslie effect	A complex combination of vibrato and tremolo effects

Vibrato is the name applied to cyclic changes in frequency. Tremolo is the term for cyclic changes in volume. The terms vibrato and tremolo are often mistakenly used interchangeably. For example, the 'tremolo' arm on a guitar produces vibrato and pitch bending, whilst a violinist's 'vibrato' is a mixture of vibrato and tremolo.

Volatile

RAM	Random-access memory = read/write memory
Battery-backed	The battery maintains the data when the power is down
Eraseable	The memory contents can be changed
Non-volatile	The device holds its data without any power present
Permanent	The device holds its data without any power present
EPROM	Erasable programmable read-only memory
EEPROM	Electrically erasable ROM
EAROM	Electrically alterable ROM

Flash EPROM	EPROM that behaves like non-volatile RAM
Flash chips	Alternative name for flash EPROM
Flash memory	Alternative name for flash EPROM
Flash	Alternative name for flash EPROM

'Volatile' means that the storage medium depends on a supply of electrical power to hold data. If the power fails then the data is lost. RAM is usually volatile. 'Non-volatile' means that the storage does not depend on an external power supply. It may have its own local power supply, or use a technology which requires no power to maintain the data.

Index