COMPUTER CONTROLLED
MUSIC SYNTHESIS

A thesis
submitted in partial fulfilment
of the requirements for the Degree
of
Master of Engineering (Electrical)
in the
University of Canterbury

by

P.J. Cheyne B.E.(Hons)

University of Canterbury
1978

ABSTRACT

Improvements made to a computer controlled digital music synthesizer are described. The basic elements of the synthesizer are discussed to give the background for these developments. Hardware additions have been made to extend the technical capabilities of the system and provide the user with computer control over various aspects of a musical sound including tremolo (frequency, amplitude and waveshape), the attack and decay rates of the notes and the volumes of the notes. Emphasis, however, is placed on describing the software developments, which are designed to provide the user with powerful facilities which are easy to use even if the operator has no computer training (as is likely with musicians). The software allows the user to interact with the timbres being produced by the synthesizer and to develop complete performances which can then be played by a computer. All aspects of the synthesizer are controlled by the computer.

ACKNOWLEDGEMENTS

Very sincere thanks to to my supervisor, Mr. W.K. Kennedy, who worked hard to help me and who showed tremendous patience. Without him I would have got nowhere.

Thanks also go to Professor R.H.T. Bates for his enthusiasm and helpful suggestions and to Warren Tucker, Susan Frykberg and David Fitzgerald whose advice and help were appreciated.

I am indebted to Mrs. Percival for doing the typing and to the Christchurch City Council Electricity Department for financial assistance during this course.

I am especially grateful for the sacrifices made by my fiancée, Christine, and for the hours she has spent encouraging and helping me.

"I thank you, Lord, with all my heart ......
You answered me when I called to you;
with Your strength You strengthened me."

Psalm 138:1,3

PREFACE

Originally only a scientific tool, the computer is now used in a wide variety of disciplines, including many that have long been considered the exclusive domain of humans. This is not to say that computers can do all that man can, but it is becoming clear that they can assist in much of the work done in most disciplines. This is as true in the arts and humanities as in any other field of study and it is with this in mind that the computer has been used as a tool for musicians. With varying success computers have been used for musical instruction, composition, production and analysis.

Since 1973 a system of computerized aids for musicians has been developed, as a series of research projects, in the Electrical Engineering Department of the University of Canterbury. Under the supervision of Professor R.H.T. Bates and Mr. W.K. Kennedy this system has grown to cover a wide variety of applications. A musician can record (on an organ keyboard) a piece of music and then edit this to remove errors or introduce particular effects and replay it as many times as he likes. The music can be replayed at any speed without affecting the pitch or can be transposed any whole number of semi-tones. The music can be displayed on a screen in a way that reflects the input exactly and thus provides information on the keyboard technique of the musician. The display can also be in the form of traditional music notation and can be printed out as such. The computer can administer aural tests arranging these to suit the strengths and weaknesses of individual students, it can harmonize music in four parts, and can assess student harmonizations. [Tucker, 1975, 1977(a), 1977(b).]

This report will deal with another aspect of the system, namely, computer controlled music performance. To this end a digital synthesizer was designed and built during 1975 and 1976 by R.J. Howarth and R.G. Vaughan [Vaughan, 1977]. This was further developed by the author during 1977 and much of the software required to control the synthesizer was written. These developments are described in this report.

The first chapter is intended as an introduction to music synthesis and the techniques commonly used. This information will allow the reader to understand and to evaluate the system described later in the report. Excellent reviews of the historical aspects of music synthesis are given by both Tucker and Vaughan [Tucker, 1977; Vaughan, 1977] and so are not repeated here. This report deals with the reasons for music synthesis, particularly computer controlled synthesis, and discusses some criteria for evaluating synthesis systems.

Chapter Two is an introduction to the system developed at the University of Canterbury. An overview of the total system is given so that the developments made during 1977 can be seen in context. The philosophy behind these developments is also discussed.

The third chapter describes the functions of the software developments in detail and shows how the facilities available to the user provide him with a flexible means of preparing musical performances.

Chapter Four evaluates the system as it stands at the moment and gives suggestions for future work.

TABLE OF CONTENTS

CHAPTER ONE

ASPECTS OF MUSIC SYNTHESIS

1.1 WHAT IS A MUSICAL SOUND?

It is difficult defining exactly what music is, and modern developments have caused this question to be widely debated. Many sounds that would not have been tolerated five or ten years ago are now becoming acceptable, and it has been suggested that some sounds are not appreciated simply because they have not been heard before. It is well known, for example, that ideas of consonance and dissonance vary between cultures and historical periods.

The noted exponent and composer of modern music, John Cage, points out that nowadays "anything goes." [Cage, 1974, p.7.] "Almost anyone who listens to sound, listens with ease to any sound no matter what overtone structure it happens to have. We no longer discriminate against noise. Also, we can hear any pitch, whether or not it is part of a particular scale, of one temperament or another, occidental or oriental. Sounds that formally seemed out of tune now seem, if anything, more distinguished than those in tune." (p.6.)

Of course many people would not agree and would suggest that there are clearly some sounds which are "musical" and others which very definitely are not. While being inconclusive, this division of opinion does show how far we still are from a clear definition of the boundary between the "musical" and the "non-musical."

While it is electronic music synthesis that has stimulated much of the above debate, we are primarily concerned, in this report, with what might be considered "conventional" music. This is not meant to indicate any limitation within the synthesis system being described, but simply to show where the emphasis has been placed. The music produced by this system is "conventional" in that it consists of individual notes of specified pitch arranged to produce "a beauty of form". [Concise Oxford Dictionary, 1964, p.795.] Each note is a periodic audio signal. This means that it is produced by the regular repetition of a particular waveform for a given length of time. The rate at which it is repeated determines the pitch of the note. The structure of the waveform during one period of the signal is called the waveshape, and this may, theoretically, take a wide variety of forms. (See Figure 1.)

The amplitude of the waveform would normally vary throughout the duration of the note. The function that describes this variation is called the envelope. The envelope will normally increase in amplitude at the beginning of the note (attack), maintain an essentially constant amplitude throughout most of the note (steady state), and then die away at the end (decay). (See Figure 2.)

The quality that distinguishes a note performed on one instrument from the same note on any other instrument is known as the timbre of the note. This is found to depend largely on the waveshape and envelope used to produce the note. In other words, the differences between timbres of various instruments correspond with differences in the harmonics (overtones) present in the sound. [Jacobs, 1973, p.388.] It has been found that there are certain families of envelopes [Moorer, 1977, p.7]. There appear to be characteristic string-like attacks, brass-like attacks and so on.

Two other factors which considerably enhance the quality of a note are tremolo and vibrato. Tremolo is an amplitude modulation applied to an envelope. This means that the volume of the note fluctuates in a regular rapid way. This is the effect obtained by rapid back-and-forth bowing of a stringed instrument. Vibrato, on the other hand, is a frequency modulation of the note. In this case, the fluctuation is in the pitch of the note. This effect is produced by an oscillatory motion in the left hand of a violinist.

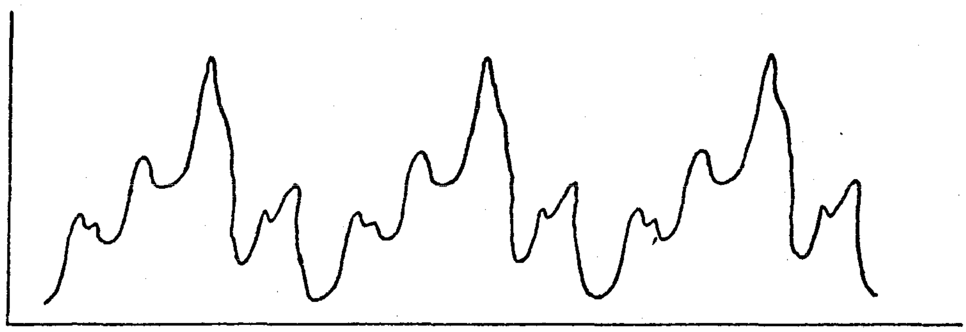


Figure 1. Sample Waveform

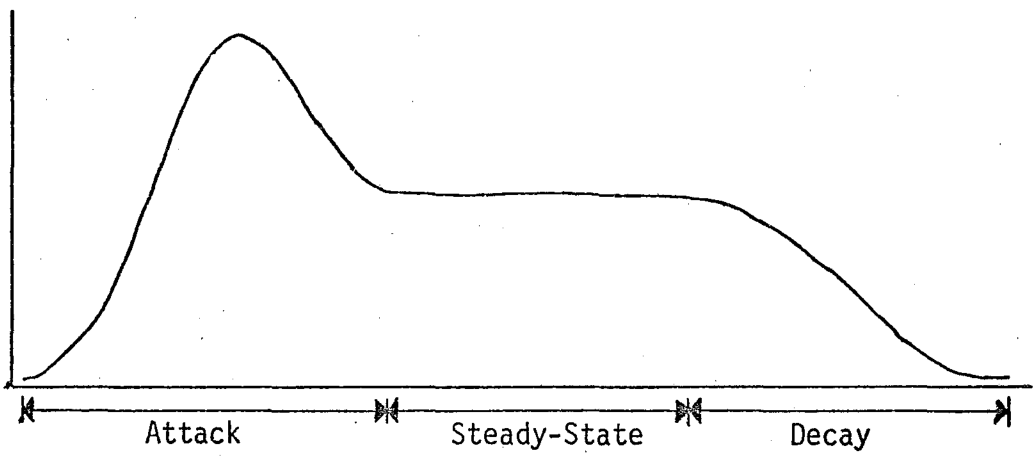


Figure 2. Sample Envelope

1.2 WHAT IS ELECTRONIC MUSIC?

Electronic music is similarly difficult to define and a wide variety of criteria for what constitutes electronic music has been proposed. In the most general sense one could include any music that employed electronic equipment in its production or reproduction. This is obviously too broad a definition to be useful as it would then include any music heard from a radio, for example, as well as music recorded by electronic means.

For the purpose of this report we will define electronic music as any music which is composed or arranged specifically to take advantage of electronic techniques. This may mean that the source of the sound is electronic (e.g. an oscillator) or that the music may be modified by electronic means (as in "musique concrète" which electronically distorts natural sound).

1.3 WHY ELECTRONIC SYNTHESIS?

Over recent years the electronic synthesizer has become a popular means of producing and modifying sounds. A synthesizer is a machine designed to produce for the operator a wide variety of sounds. These may range in complexity from a pure sinusoid to white noise and so cover both (quasi) periodic signals (generally considered musical) and irregular aperiodic signals (thought of as noise). Consequently, it is possible to produce electronically any type of oscillation at any frequency. [Douglas, 1973, p.2.]

Essential to any synthesizer is an oscillator. This is a device that produces a cyclically varying voltage, which can then be converted by a loud speaker to a pressure function in the atmosphere which we perceive as sound. However, before this conversion it is highly likely that the operator will want to modify or enhance the oscillation in some way. For this, the synthesizer will probably include various components such as filters, mixers, ring modulators (multipliers), envelope shapers, amplifiers, reverberators and other specialized equipment as well as monitoring equipment such as VU meters and spectrum analysers.

There are many different synthesizer designs and synthesis techniques, all of which generate and/or modify electrical signals to produce the sounds desired by the operator. These techniques will be explored further in Section 1.5 and later in the report. But the question still remains as to whether this form of music-making has made any significant contribution to music that could not have been made by conventional instruments.

Over the ages techniques of music-making have evolved and developed. Initially all music would have consisted only of vocal sounds which may later have been accompanied by the banging together of pieces of wood. However, as civilisation has progressed, so has the complexity of the music it has produced. Technical developments have been employed to improve and refine musical instruments, and the resultant greater power and extended tonal qualities have been incorporated into the music of the time. This process continued into the nineteenth century when physics could no longer provide improvements to the qualities of conventional instruments.

Composers and musicians therefore had to be content to work within the limitations inherent in the instruments available to them. This meant that music ran the risk of stagnating since it seemed there was no longer any room for innovation. In fact, as recently as 1963, the German conductor, Hermann Scherchen said "that music was a dead art in the sense that Latin was a dead language - that the rich vein of human expression based on modal harmonies and melodies had been exhausted and that no humanly valid alternatives exist, hence no new music could be written." [Mathews, 1974, p.263.]

As an example of the limitations of "conventional" instruments consider the question of temperament. <u>Temperament</u> is a term used to describe the way an instrument is tuned to play only certain pitches which fit into some form of scale. A "natural" scale is one based on mathematical laws, but it is impossible to implement exactly on any fixed interval instrument, if that instrument is to remain physically playable. Consequently, today, all keyboard, woodwind and brass instruments except the slide trombone are equally tempered. This means that the intervals of the natural scale are each lessened or extended slightly so that an octave (i.e. a frequency range of 2:1) can be covered by 12 notes with an equal interval between each pair of successive notes. It is a general rule that two notes sound concordant if their frequencies can be expressed as the ratio of two small integers (e.g. 2:1, 3:2, 4:3 ....). However, with the equally tempered scale in which the ratio of the frequencies of two adjacent notes is $\sqrt[12]{2}$ (1.05946) it is impossible to find notes which have the desired small integer ratios. For example, the closest one can get to $^3/2$ is $2^{7/12}$ (= 1.4983071) and the nearest to $^5/3$ is $2^{9/12}$ (= 1.6817928). This obviously introduces errors and these are compounded in waveshapes rich in harmonics. To have more than 12 notes per octave improves the temperament, but of course makes the instrument very difficult to play. In fact a 53 note octave has been proposed, but as well as being hopelessly impractical, this is still slightly imperfect. [Douglas, 1973, p.32.] From this we can see that most conventional instruments sacrifice the small integer ratios (and hence true concordance) for the sake of easier manipulation. Furthermore, there are only 120 notes in the equally tempered scale, whereas the human ear can distinguish approximately 1400 pitch intervals over the same range. Obviously the use of instruments which are not restricted to the equally tempered scale would make possible a huge number of new musical forms.

Signals of any desired frequency can be generated electronically and these can be sharpened or flattened at will. This gives the composer or performer the opportunity of producing truly harmonious chords etc. which he cannot do with conventional instruments, as well as being able to use many more pitches and combinations of pitches than is otherwise possible.

In virtually all conventional instruments both the power and the quality (harmonic content etc.) of the very high notes is markedly different from those of the very low notes. This limitation is imposed by the size and shape of the instrument and the fact that the energy is provided by the performer who has only limited stamina. Because of these restrictions, the performer cannot exploit the full range of notes available with equal effect; moreover any one instrument is limited to a particular type of timbre and a particular range of pitches. This means that the composer requiring a string tone may have to use violins, violas, 'cellos and double basses to cover the required frequency range, and even if a violin can produce a certain pitch it can never sound like a clarinet at that pitch.

Electronic synthesizers are not restricted in any of these ways and so provide the performer with infinitely greater versatility.

Most instruments depend on resonance to generate the required power. This means that more power is required initially to start the system resonating than to maintain it in that condition. However it is impossible to sharply reduce the energy input once it has been applied and so much of the excess energy is manifested as noise. Thus we are accustomed to excess wind noise, bowing noise or the noise of the impact of piano or xylophone hammers, and the click of valves. These are accepted but with advancing technology this degradation is no longer necessary.

Resonant systems also require a certain amount of time to release their energy and this means that instruments have a natural decay time. This can be reduced by damping, but this often gives the decay curve an undesirable shape and still means that the decay is very much out of the control of the performer. There is no such limitation with electronic systems, which is advantageous as the attack and decay of a note have been shown to be important in its overall effect. Adding all of these constraints together, one can see that there is obvious value in using a system that is not subject to them. Instead of being limited to a particular set of characteristics (only some of which are useful), the composer or performer can "build up" an instrument, using a synthesizer, to his own exact requirements, giving him far greater control and flexibility both within and beyond the scope of conventional instruments.

To summarize, the advantages of electronic systems include :

- greater dynamic (power)range (hence crescendos and diminuendos impossible with conventional instruments)
- greater pitch range
- infinitesimal pitch gradations (hence any degree of glissando or sliding scale, and no pitch inaccuracies as with the equally tempered scale)
- completely new and infinitely variable timbres and musical phrasings
- any extent of echo or reverberation
- arpeggios or similar progressions at greater rates than humanly possible
- far greater control over the above parameters as well as others such as tremolo and vibrato.

1.4 WHY USE COMPUTERS?

A "natural" musical sound is highly complex and is therefore difficult to simulate exactly. Not only must several parameters be controlled at the same time but this control must be fast and accurate. Within the duration of a single note the frequency spectrum of the waveshape can vary considerably and at the same time the envelope, tremolo, vibrato and volume may be changing.

Digital synthesis systems (Section 1.5.1) must produce a sequence of samples to describe the required waveshapes and envelopes. For high quality synthesis each second of a sound is likely to consist of at least 35,000 samples, and the calculation of each individual sample may require several arithmetical operations, taxing even the fastest of the current generation of computers. Of course, this speed is not required if the samples can be calculated before the music is actually heard and stored away until needed. Similarly, time is saved if the samples of any of the waveshapes or envelopes that are to be used repetitively are only calculated once and then stored. This illustrates another of the strengths of the computer that make it useful in this type of application - its ability to store vast quantities of data.

When making music with a conventional instrument many things such as the envelope and harmonic structure of the sound produced are out of the control of the musician. To control these things as well requires far greater dexterity and speed than a human possesses. A computer, on the other hand, can supervise and control all aspects of a musical performance and can make changes with far greater precision than would otherwise be possible. While doing this it can also respond to interaction with the musician which will determine the type of control that is desired and the actions to be taken.

The fact that a computer can do anything for which a suitable algorithm has been written means that the performer has tremendous flexibility and power. The computer can simulate any signal processing function as well as being able to do things which cannot be easily implemented by other means.

However, it is not necessary that the musician write all of the programmes himself. In fact, if a more technically orientated person writes the programmes, with a sensitivity to the needs of the musician, the computer can then act as the bridge between the purely musical and the purely technical. This is an important aspect of our work since it provides the musician with the most powerful tool currently available to him, [Laske, 1974] without demanding that he be familiar with the technical processes being performed. In this way the musician can do the type of things he wishes to do both quickly and easily. This is particularly appreciated if the task is highly repetitive or tedious, and it is in this type of task that the computer excels. A computer can do the same job in the same way any number of times without making a mistake. On the other hand it can make minute and subtle changes in a very precise manner. Here again we see the flexibility that is available to the musician.

Using a computer, the musician is able to perform "micro-surgery" on his music. [Moorer, 1977, p.4.] He is able to account for and control the tiniest detail - a single waveshape sample can be altered, the timing of an individual note can be changed by a few milliseconds. Far from replacing the musician (as is often feared), it becomes obvious that the qualities of a computer and those of a human are essentially complementary. While the computer takes care of the more mechanical and repetitive tasks, the musician is freed from these to concentrate on the creative and emotional aspects to which he is infinitely more suited than any machine.

A computer can give undivided attention to unlimited detail, is precise, reliable and immune to distraction, while being able to carry out the most intricate and lengthy calculations with ease and without tiring. Of course, a computer can do none of these things unless it is furnished with a suitable programme. It is the job of the programmer to transform the machine into the powerful tool it can be, and this is where the effort must be made if a system is to be useful to and easily utilised by a musician.

1.5.0 SYNTHESIS TECHNIQUES

The two main aspects of music synthesis are the actual production of sound and the specification of the parameters that combine to make up that sound. However in both of these areas there has developed a wide variety of techniques and processes. We will look at some of the main classifications into which these fall in order to obtain an overview of some of the ways that have evolved for specifying and producing synthesis music.

1.5.1 Analogue or Digital?

All synthesis equipment is either analogue or digital or a combination of the two (hybrid). Analogue systems, which are by far the most prevalent, are those which deal with signals which vary continuously in amplitude or frequency. In other words the musician works with signals that mirror exactly the pressure function of the sound they would produce.

Digital systems, on the other hand, deal with numbers that represent the amplitude of the corresponding analogue signal at various times. An analogue signal can be converted to a digital one by measuring and recording the amplitude of the signal at regular intervals. This must be reconverted to an analogue signal before the sound can be heard. A digital system can consist of a computer only, a computer controlling some specialized digital signal processing hardware or a non-computerized digital synthesizer. A hybrid system almost always consists of a computer controlling (via digital-to-analogue converters (DAC's)) a collection of analogue signal processing equipment.

An analogue synthesizer is controlled by a set of potentiometers which vary the parameters of such components as oscillators, mixers, filters, amplifiers, etc. The musician "twiddles" these until he obtains a sound he likes. However, unless he tape records it there and then, it is quite likely he will never be able to hear that sound again. Even if he makes a note of every patch (connection between various components) and of every potentiometer setting, he is unlikely to be able to reproduce them exactly. Even a tape recording of the sound is not an entirely satisfactory answer since this is subject to noise such as "hiss" (wideband white noise) and "dropout" (irregularities in the oxide coating). Moreover each time the tape is copied, more and more noise is introduced. This is a serious consideration as analogue synthesis depends on re-recording sounds many times as they are modified or as other sounds are added in. In contrast, digital data can be stored, used, transmitted or manipulated without deteriorating. What is more, digital recording can be extremely accurate and can provide superior sound quality. [Blesser, 1975, p.698.]

As opposed to analogue systems, digital systems are inherently repeatable. Unfortunately the problem of trying to reproduce a potentiometer setting exactly is compounded by the fact that the components of analogue systems tend to drift (i.e. their values change due to temperature and age).

As mentioned, digital data can be manipulated without introducing any inaccuracies. The manipulation is often a lot easier as well. Digital signal processing techniques can be implemented to simulate the operation of any of the analogue components (oscillators, multipliers, filters etc.). The advantage of analogue circuits in this regard is that these processes can be implemented more or less directly, whereas a suitable algorithm is required to do them digitally. Once a digital system is designed, however, it can normally be depended upon to operate more reliably. Digital data can be encoded in some form that makes the manipulation more efficient or as a protection against errors.

Another advantage of analogue systems is that they can be patched in a variety of configurations and give immediate results. In other words the various components can be wired together in any of a large number of ways so that output voltages from some modules can be used as control voltages for others and so on. However, manual patching is slow and so not suitable for rapid changes. Computerized patching has been attempted but is complex and cumbersome. Furthermore, some synthesis programmes for purely digital synthesizers have been written to simulate the patching together of analogue modules (e.g. MUSIC V). [Mathews, 1969, p.37.] This eliminates the problem of timing and is perhaps even more flexible than an actual analogue synthesizer in that it does not suffer from circuit redundancy or insufficient modules.

1.5.2 Turn Around Time

Another important consideration when classifying systems is the amount of time it takes to get from an idea to a sound. The ideal situation would be for a system to be totally real-time. This would mean that the musician could interact with the system and the results of this interaction would be reflected immediately in the sounds heard. However, since a great deal of computation is required for any degree of high quality synthesis, this appears to be an unrealizable goal unless special purpose hardware is employed. [Moorer, 1977, p.5.]

The other extreme is an off-line system and most of the synthesis systems that have been developed so far fall into this category. This is far from ideal, but is often necessary where the computer is not dedicated entirely to music but is shared by many users. Under this sort of batch processing the user usually submits a set of parameters to the computer (often on punched cards) and sometime later the appropriate computation is carried out. The resultant digital samples are either recorded on magnetic tape (requiring later digital-to-analogue conversion) or they are fed directly to a DAC and the analogue signal recorded. Either way the sound is not heard until sometime after the process is started. Although economizing on computer time and not requiring any special purpose hardware, batch processing can result in delays of up to several days.

The disadvantages are obvious. Since much of the human side of music production is entirely subjective, it is often necessary to be able to experiment with various parameters. If each parameter change takes several hours or several days the musician is unable to compare the current sound with the previous one because he cannot remember it. Without immediate feedback it is very difficult trying to 'get a feel for' the effects of various parameters. As mentioned, one of the advantages of computerized systems is their ability to make subtle changes. Unless the effects of the changes can be heard within a few seconds this advantage is lost. It is therefore necessary to have a system with which one can interact freely and which responds in as short a time as possible. Providing the computer is available, it is relatively easy to develop an interactive system but it is more difficult striving for "real-time" operation.

1.5.3 Special Purpose Hardware

The computer is quite capable of producing music without the help of any external hardware besides a digital-to-analogue converter, an amplifier and a loud speaker. The simplest way of doing this is by using a single register bit. A bit is a computer element which can have one of only two values (voltage levels). It can be made to oscillate between these two values by various computer operations, producing square waves of varying frequency. This technique is simple, but very limited.

Pure computer synthesis is the process whereby a computer calculates samples and presents these to a DAC. The output of the DAC is either recorded, as is necessary with batch-operated programmes such as MUSIC IV, MUSIC IVB [Mathews, 1961, p.677], MUSIC 360 [Pierce, 1965] and MUSIC V [Mathews, 1969], or is fed directly to an amplifier to be heard immediately. Programmes such as GROOVE [Mathews, 1970], POD6 [Truax, 1973] and EUTERPE 8 [Smoliar, 1973] employ the second of these techniques. The problem here is that it may require several seconds of computer time to calculate all the parameters of one second of sound.

To obtain real-time operation it is necessary to use special purpose hardware. This equipment is expensive and is usually not immediately transferable to another computer, but it is necessary if one wishes to approach real-time. The advantage is that the extra hardware and the computer can be operating in parallel, thus increasing the degree of real-time interaction possible without losing the flexibility of digital control. There is no reason why the extra hardware cannot itself contain a mini-computer or micro-processor and this approach is being tried more and more. The special purpose hardware should be designed with the needs of the particular system in mind but the general idea is to relieve the computer of the simple but time-consuming tasks, freeing it for more efficient use. Whatever design is adopted it does mean that the tasks are shared and the goal of real-time operation is closer.

1.5.4 Sound Specification

There are three main techniques used in synthesizers to produce sounds. These are (a) additive synthesis, (b) subtractive synthesis and (c) summation formula synthesis. In this section each will be discussed in more detail with particular reference to the parameters that must be specified by the user for each one.

Any periodic waveshape can be thought of as the sum of a series of harmonically related sinusoids. These are known as the Fourier components or harmonics of the waveshape. In other words, any periodic musical sound consists of a fundamental tone plus overtones. It follows from this that any periodic waveshape can be synthesized by adding together a series of suitably weighted, harmonically related sinusoidal signals. This is done and the technique is known as additive synthesis. All that is required is a bank of sinusoidal oscillators and a circuit for summing their outputs. Of course, to be able to obtain different waveshapes one must be able to specify various values for the amplitude of each harmonic. However, in practice it is found that sounds are more interesting if the amplitudes of the partials vary within a note forming what is called a quasi-periodic waveform. A typical expression for the value of the n^{th} sample of a waveform would be :

$$X(n) = \sum_{k=1}^{M} A_k(n) \sin \{ nT [k\omega + 2\pi F_k(n)] \}$$

where

	$X(n)$	is the sample value at time nT
	n	is the sample number (time index)
	T	is the time between consecutive samples
	ω	is the fundamental frequency of the note (radians)
	k	is the harmonic number
	$A_k(n)$	is the amplitude of the k^{th} harmonic at time nT (assumed to vary slowly in time)
	M	is the number of harmonics
	$F_k(n)$	is the frequency deviation of the k^{th} harmonic at time nT.

[Moorer, 1977, p.9.]

To be able to evaluate this function at any time, the computer must be supplied with values or functions for each of the parameters ω, M, T, $A_k(n)$, $F_k(n)$. This is a powerful technique since, as mentioned, any quasi-periodic waveform can be realised. It also has the distinct advantage that the values that must be specified for the parameters given above can be estimated from analysis done on existing musical sounds. This has been found to produce "synthetic sounds with remarkable similarity to the original sound." [Moorer, 1977, p.5.] Of course a synthesizer can be used to produce sounds other than those produced by conventional instruments but the fidelity with which it can reproduce known sounds is a useful bench-mark as to the quality of the technique used. The disadvantage of this type of synthesis is that it is only useful for producing "periodic" tones of nearly constant frequency.

Another widely used technique is subtractive synthesis. This uses an harmonically rich waveform such as a square wave or pulse train (periodic) or white noise (aperiodic) as an excitation function which is modified by a time varying filter. That is, one starts with many harmonics, and attempts to filter out those not required. This technique is used extensively for speech synthesis as well as music. The parameters that must be specified include the nature of the excitation function (periodic or aperiodic), the excitation function fundamental frequency (if periodic), the overall gain and the filter coefficients. Once again, these can normally be determined from analysis of actual waveforms.

It is thought that this method could be even more flexible than additive synthesis since the harmonic content (determined by the filter) and the pitch (determined by the excitation function) are independent and can be varied independently. To date however, subtractive synthesis has not produced the same high quality sounds as additive synthesis despite the use of very accurate high order filters. This is thought to be because insufficient is known about modelling the excitation function and good tools for generating this are not available. [Moorer, 1977, p.16.]

The third main synthesis technique is different from the previous two in that its parameters cannot be readily derived by analysis. Consequently, results are only obtained by empirical means. One must develop an intuitive feeling for the parameter variations that produce particular types of sounds. This technique is classified as summation formular synthesis. By evaluating particular formulae it is found that the resulting function (if converted to a voltage) produces sounds which, under certain circumstances, are "musical". The most important and widely used of these formulae is the frequency modulation equation. [Chowning, 1973.] This can be represented as :

$$X(n) = A(n) \sin \{2\pi f_c nT + I(n) \sin (2\pi f_m nT)\}$$

where $A(n)$ is the amplitude of the waveform (i.e. constitutes the envelope)

$I(n)$ is the modulation index (and is usually a simple, slowly varying function).

This can be expressed as a summation of a series of sinusoids, the frequencies of which are separated by the frequency, f_m. What is more, these make up a harmonic spectrum if f_c and f_m are related by a rational number. The amplitudes of these harmonics are sums of Bessel functions which are themselves functions of the modulation index. Thus by specifying f_c and f_m and varying the modulation index the spectrum can be changed dynamically, creating rich and colourful sounds.

We have spoken mainly about the specification of waveshapes because it is here that the harmonic components are of prime importance. However, in the specification of envelope values it is the time domain representation rather than the frequency domain representation that is important. This illustrates the importance of being able to specify shapes in either domain. Specification in the time domain is more easily implemented since it only requires the user to tell the computer the values of the samples it is to use. It must be remembered that methods should be developed that make this as easy and as natural for a musician as possible.

1.5.5 Quality of Sound

The two biggest factors influencing the fidelity of digitally synthesized sound are the number of bits per sample and the number of samples that are calculated for one second of sound. It is not difficult selecting appropriate values for these, but they must be kept in mind when designing a system. Each sample is a numerical representation of the amplitude of the signal at a particular time. The more bits there are to a sample the greater the number of voltage levels that can be described by that sample. Consequently, the accuracy with which any voltage is described increases if more bits are used. Any inaccuracy introduced by the sampling process is called quantization noise. If the number of bits to be used (n) is known, the rms signal to quantization noise ratio can be calculated by using the following expression :

$$S/N \doteq \sqrt{3/2}\ .2^n$$

The maximum number of bits which can be used is often determined by the availability of a suitable DAC. However, DAC's that can handle 16 bits are now obtainable and this is quite sufficient for high quality sound production. $S/N = \sqrt{3/2}\ .2^{16} = 80264 = 98\ dB.$

Even 12 bits (74 dB) are sufficient except when very soft sounds are generated. [Mathews, 1969, p.6.]

The frequency range that can be obtained from a synthesizer depends very much on the sampling rate. It can be shown mathematically that a signal (if it is to be faithfully reproduced) must be sampled at at least twice the frequency of the highest frequency component of that signal. Since humans can hear frequencies up to about 16 kHz, sampling rates in the range of 35 kHz must be considered. This gives us some idea of the rate at which we are asking the system to perform. At this rate, a three minute performance would require an output of 6.3 million samples. It is obviously necessary that the hardware can present samples to the DAC sufficiently quickly. If a slower sampling rate must be used, the frequency range of the system is limited.

The next chapters describe the work of the author in relation to an actual system.

To control the synthesizer from the Decwriter, the user must learn the functions of the various commands available to him. This is no more unreasonable than expecting a musician to learn the rules appropriate to any instrument. Every effort has been made to make the system easy to use, and so the commands have been kept simple, while the jobs performed by them are powerful and suited to the needs of a musician.

The synthesizer system is only part of a very much larger system of computerized aids for musicians and is compatible with other aspects of that system. [Tucker, 1977.] This greatly extends the flexibility available to the musician as, for example, the notes which are to be played through the synthesizer can be recorded, replayed, edited and printed out, until the user is completely happy with them, before a timbre is associated with each one. In other words, while other aspects of the system are concerned with the notes themselves (i.e. their pitches, start times and durations) the synthesizer system is concerned with the timbre associated with each note.

2.1.2 Specifying a Sound

In developing a timbre the user must select a waveshape and an envelope that produce the sound he desires. By specifying a particular shape (for use as a waveshape) the user is automatically specifying the number and amplitudes of the Fourier components (harmonics) that make up that shape. Consequently, he does not have to worry about adding more in or filtering some out as is the case with the additive and subtractive techniques. (Section 1.5.1.)

There are various ways one can specify a shape, but they fall into two main categories. One can either specify the amplitudes of the components that make up that shape (i.e. in the frequency domain) or one can actually describe the form of the shape (time domain representation). The various routines that have been written to make shape specifying quick and easy are described more fully later in this chapter and in Chapter Three.

Once a shape has been produced it is stored in a catalogue of shapes and given a catalogue number. By referring to the catalogue number, any shape can be retrieved for use or for further modification. A shape that is not quite as the user requires can be modified in a number of ways.

Any catalogued shape can be used at any later time, as either an envelope or a waveshape. The catalogue system is a similar concept to that used with MUSIC V [Mathews, 1969, p.35] and GROOVE [Mathews, 1970, p.715].

The timbre can be further enhanced by the specification of a particular type of tremolo. The user has control over the waveshape of the tremolo function, its amplitude and its frequency, and therefore has at his disposal a wide variety of tremolo functions which can be applied to any of the notes played. These functions include signals that are themselves at audible frequencies and which produce interesting effects by beating with the note. The volume of the individual notes can also be controlled.

Each note played is either <u>percussive</u> or <u>non-percussive</u>. A percussive note goes through the attack portion of the envelope and, omitting the steady state portion completely, goes straight through the decay and stops. A non-percussive note stays in the steady-state portion for as long as is desired by the user, before beginning the decay. A note played on a piano is percussive since it dies away and stops even if the player keeps his finger on the key. An organ, however, will continue playing a note as long as a key is depressed.

As well as controlling whether a note is percussive or not, the user can also determine how long the attack and decay portions take to be completed. They can be short, producing very abrupt (staccato) notes or they can take several seconds so that the note builds up and decays slowly (legato).

It would be possible to include the gain information and the attack and decay rates into the envelope, since the gain is simply a multiplication of the amplitude by a constant factor, and the attack and decay rate simply a multiplication of the time scale by a constant factor. However, there are distinct advantages in maintaining complete independence of timbre parameters. The most obvious advantage is that, with independent parameters, the shape of the envelope is the important factor and not the units used to describe it. Therefore the envelope can be used with any note irrespective of the gain to be applied to it. Far fewer envelopes need to be specified than would otherwise have been the case. As well as this, all shapes can be standardized to a maximum amplitude of 8 bits. Parameter independence also greatly increases the resolution obtainable. An independent gain factor, for example, increases the number of different amplitude levels from fewer than 256 to 65236 (256^2). This finer resolution is particularly noticeable during soft sounds. The independence of the gain variable from the envelope also gives rise to the possibility of creating new envelopes by varying the gain during the note rather than just between notes. Because of these advantages, all of the timbre parameters can be varied independently by the computer or by the user via the computer.

An important aspect of this synthesizer is the idea of a <u>voice</u>. A voice is essentially a channel through which a sound is made. The number of voices determines how many different sounds can be heard at any one time. Each voice is completely independent of the others and each contains all of the elements of the final composite sound produced. For example, the tremolo on each voice can be controlled, and each voice can be producing different tremolo signals at the same time. When varying timbre parameters, the user can specify to which voice the change is to apply. The individual elements of a single voice will be described more fully in Section 2.1.4.

A note can be played over a particular voice only if that voice is "free" (i.e. it is not already being used to play a note). When playing individual notes, the user can select the voice over which the next note is to be played. If a sequence of notes is being played by the computer, it must be continually monitoring the states of the voices in order to know which ones are available at any time.

A multiple-voice system has the advantage that two notes originally recorded as non-overlapping, can be made to overlap in playback while still starting at the correct times. This is the situation that arises, for example, when a piano key is struck and the note continues to be heard for some time even though subsequent notes are being played. The first note can still be heard because of the time it takes to complete the decay part of the envelope and this effect can be simulated on this synthesizer by specifying a slow rate of decay (i.e. a slow envelope clock). This is the same effect as is produced by the "sustain" stop on an electronic organ. The disadvantage of a system consisting of a fixed number of voices is that if, at any time, the total number of notes required exceeds the total number of voices available, some must be delayed or ignored.

2.1.3 Using the System

The ultimate goal of the user is likely to be the performance of a piece of music consisting of a variety of notes. However, in working towards this, he will probably wish to experiment with individual sounds and to vary any of the timbre parameters whilst leaving the rest unchanged. Provision is made in the software for this.

A performance consists of a score and an orchestration. The score contains the information about the notes that form the piece of music (i.e. the pitches, start times and durations) and the orchestration contains the information on the timbre. In the context of this system, the timbre is made up from the waveshape, envelope, tremolo waveshape, tremolo amplitude, tremolo frequency, envelope clock rate and the volume associated with each note. The score provides the skeleton for the performance while the orchestration puts the flesh on the bones.

As we have mentioned, other aspects of the music system are admirably suited to score production and modification. [Tucker, 1977.] The synthesizer system is primarily concerned with the orchestration of the score. This basically involves associating with each note the timbre parameters required.

The software modules written fall into categories that reflect the major steps that lead to the production of a performance. These classifications are :

1) specifying and modifying shapes
2) producing individual sounds and experimenting with the timbre parameters
3) preparing and playing sequences of notes (i.e. the performance).

In Section 2.4 we will look at the range of commands available and how these can be combined to form a useful system.

Although the system is designed so that even the most inexperienced user can get results, he will benefit from practice and experience as he would with any other instrument. As the musician learns more and more about the capabilities of an instrument he is able to use it to greater effect, and to ask more of it. This is particularly true of a synthesis system as there are many more possibilities to be explored than with a conventional instrument. To develop this rapport between musician and instrument, it is necessary to provide the user with feedback within a short time that demonstrates the effects of his actions. The ideal situation would allow the user to vary parameters continuously and for the sound heard to vary at the same time, reflecting the changes. This is real-time interaction and will be discussed more fully later.

2.1.4 Background to Synthesizer Hardware

Vaughan (1977) documents the synthesizer hardware as it stood at the beginning of 1977, but a brief overview will be given here so that the developments that have been made can be seen in context.

Figure 3 is a schematic representation of the various components of a single voice showing how the timbre parameters are incorporated into the final sound.

The most important elements of each voice are the two shift registers which are recirculating digital memories and which hold the samples up the envelope and waveshape to be used on that voice. Each shift register contains 128 eight-bit samples. For high quality music production samples of 8 bits would be insufficient (see Section 1.5.5) but for experimental purposes they are adequate. Standardizing the samples for both the envelope and the waveshape to 8 bits also simplifies the shift register loading hardware and thus reduces the cost. Before a note can be played, both shift reigsters must be loaded with the appropriate samples, which are calculated by the computer and loaded via the B.D.I. into the shift register of the appropriate voice.

To play a note the waveshape shift register is recirculated and the samples cyclically presented to a DAC. Each of the 128 samples must be presented during one period of the note and so the recirculating clock frequency is 128 times the frequency of the note. This clock signal is obtained from a commercially available integrated circuit, which produces the frequencies of the twelve equally tempered notes of the top octave. The notes of the other four octaves (61 notes are available) are derived from these by successive divisions by 2. The

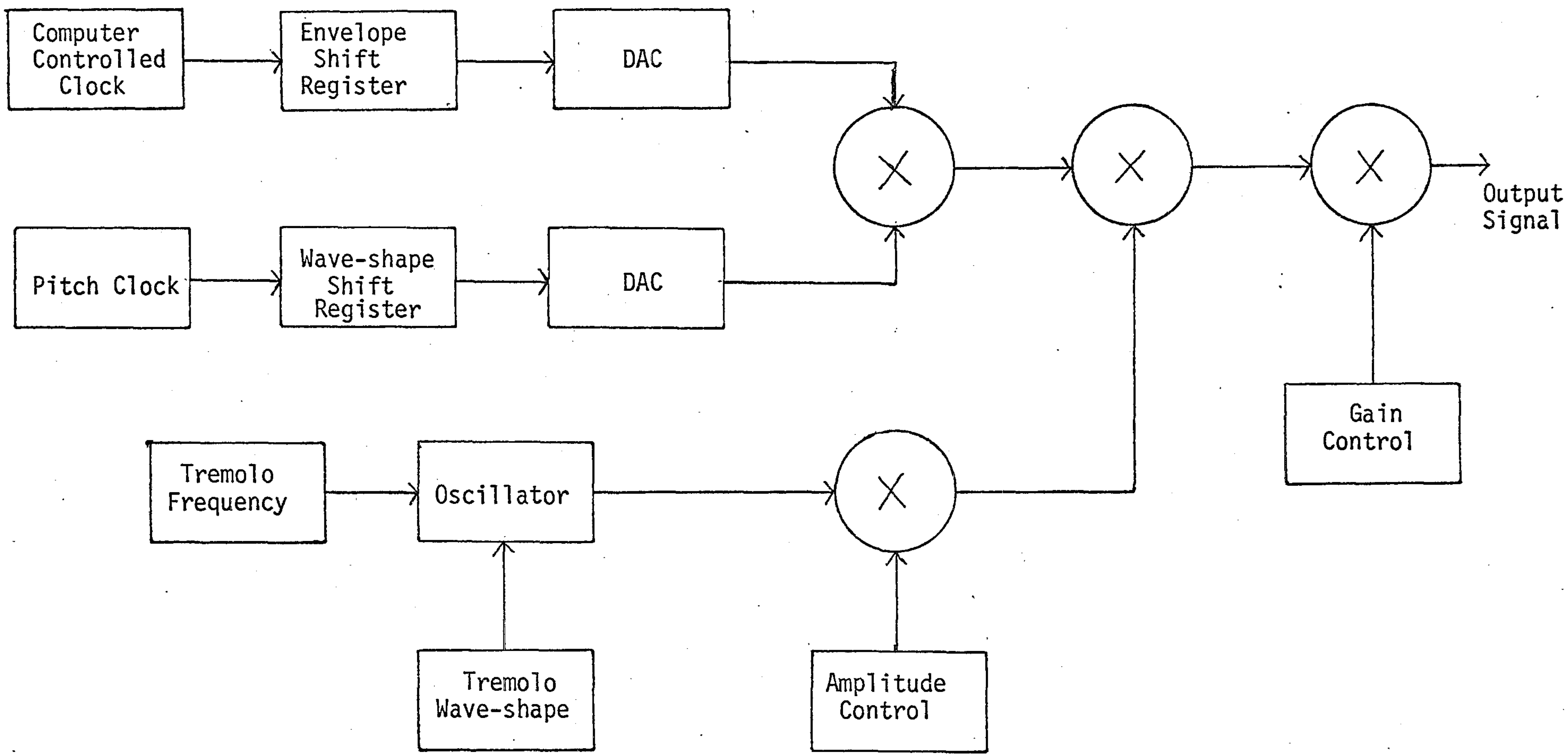


Figure 3. Voice Components

integrated circuit is driven by a crystal oscillator and so the resultant pitches are highly accurate and stable. The generation of the pitches of the lower octaves by frequency division also means that the relationships between the various pitches is stable and no tuning is required. While these equally tempered frequencies are required for the production of conventional music, there is also provision in the system for the utilization of a continuously variable clock to drive the waveshape shift register. This allows for the production of music not limited to the equally tempered scale.

The envelope shift register contains the attack and decay of the envelope, but not the steady state portion. When a note is started, the shift register cycles through one half of the available samples (64) and then stops. The envelope then remains constant until the decay is initialized, when the shift register presents the remaining 64 samples to the DAC. Consequently, while the attack and decay take a certain length of time, the steady-state portion can be of any duration and the envelope shift register recirculates exactly once no matter what the length of the note. To give the user control over the attack and decay rates of the envelope, this shift register is also driven by a variable frequency clock. This is described in Section 2.2.4.

The inclusion of tremolo and gain, and the implementation of computer control over all of the timbre parameters is also discussed in Sections 2.2.3 and 2.2.5.

The main tasks of the synthesizer are to control the playing of notes at particular pitches, incorporating any combination of the timbre parameters and to route all timbre data (including the waveshape and envelope sample) and control signals to the appropriate section of any selected voice. The data channel is time-shared rather than parallel hardware and so is common to all voices. This arrangement avoids circuit redundancy and expensive duplication of hardware.

Both the hardware and the software have a modular structure which, as well as making them flexible to use, means that both can be expanded easily by adding new modules that are compatible with existing ones. As far as the hardware is concerned, this modularity means that all components of the final sound are available and could be processed or modified by any new modules added to the system. The main advantage in the software is that the modules can be used in almost any order allowing the user to perform a wide variety of tasks in the way he finds easiest and most natural.

2.2 HARDWARE DEVELOPMENTS

2.2.1 Voice Extension

Work has been done by the author to extend the synthesizer from a two voice instrument without computer controlled tremolo, gain and envelope clock, to one of four voices with the above control. This means that four sounds can be heard simultaneously instead of only two, thus reducing the chances of the timing of the notes being disrupted, because of notes having to be delayed until a voice becomes free.

This system was originally designed to control sixteen voices, and it would be extremely flexible if all of these were implemented. It was decided, however, that this number of voices was not necessary for experimental purposes, and the cost was not warranted. Nevertheless, the fact that the system was designed to control sixteen voices does show that this is a feasible and worthwhile goal.

2.2.2 Multiplication

We saw in Section 2.1.4 that the envelope and waveshape samples are converted to analogue signals before being multiplied together. (See Figure 3.) The multiplication of the resultant signal with the tremolo signal and gain factor is also done in the analogue phase. This introduces some inaccuracies due to drift and noise susceptibility (Section 1.5.1) but there are also definite advantages. Digital multiplication is possible and accurate but hardware multipliers of sufficient speed are only just becoming available [Snell, 1977, p.38], and are very expensive. If this was not the case, digital multiplication would be preferable and this may become feasible with advances in technology and reductions in costs.

To perform these multiplications, an analogue module was added to the system to take the place of the analogue computer which had previously been employed. A schematic diagram of this can be seen in Figure 4.

2.2.3 Tremolo

The tremolo signal is generated by another module which employs a voltage controlled oscillator (V.C.O.) in the generation of sine and square waves at varying frequencies. Figure 5 shows how the waveshape, frequency and amplitude are specified in digital form by the computer, thus providing quick, accurate and repeatable control.

This module has also been incorporated into the system and provides four types of tremolo waveshape (no tremolo, sinewave tremolo, squarewave tremolo and the sum of a square and a sine wave tremolo) at frequencies from 0.1 Hz to 1 kHz. These frequencies are divided into two ranges (0.1 Hz to 50 Hz and 10 Hz to 1000 Hz) and the amplitude can be varied from 0% to 100% modulation. This module replaced a manually controlled sine wave generator that was used to produce the tremolo signal.

It would be possible to produce a digital tremolo signal but this, once again, would require digital multiplication. It cannot be superimposed on the envelope within the shift register because no

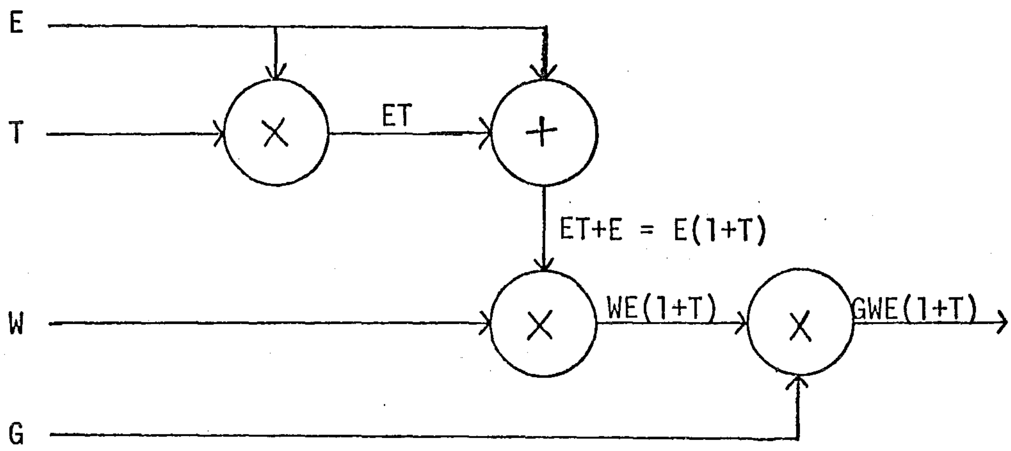


Figure 4. Multiplier Module

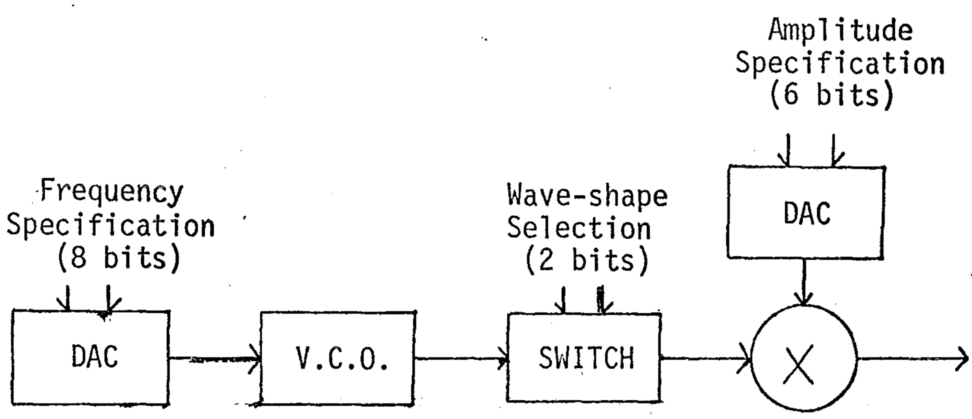


Figure 5. Tremolo Generator

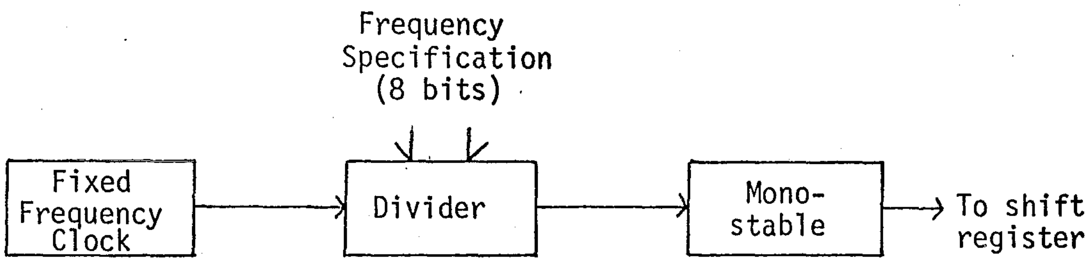


Figure 6. Envelope Clock

tremolo would be heard during the steady state portion of the note, when the shift register is not being recirculated.

Both the multiplier module and the tremolo generator were designed and built by David Fitzgerald, under the partial supervision of the author, as a Third Professional Year project and are more fully documented by him [Fitzgerald, 1977].

2.2.4 Envelope Clock

Another addition to the hardware was a computer controlled clock for the envelope shift register of each voice, providing the musician with control over the attack and decay rates of the notes. It was necessary to be able to vary the attack and decay times from several milliseconds to several seconds and this was achieved by designing a programmable frequency divider circuit driven by a fixed frequency clock as can be seen in Figure 6.

Using this method a maximum/minimum frequency ratio of 256 is obtained which allows the duration of a note to vary from, say 40 msec. to 10 seconds. This circuit was implemented in TTL logic and is documented in Appendix A.

2.2.5 Computer Control

As a result of these additions, all of the following parameters could be applied to each note played :

Tremolo waveshape	2 bits
Tremolo amplitude	6 bits
Tremolo frequency	8 bits
Envelope Clock rate	8 bits
Gain	8 bits

To be able to change these quickly and accurately, computer control had to be implemented. This involves routing the timbre data to the relevant latch of the appropriate voice. It was decided to modify the synthesizer control unit slightly so that eight of the unused voices (i.e. voices 5 to 12) could be used to route this data instead of the waveshape and envelope data that would have been required had these voices been implemented. This also required new software. All data had to be latched because of the time shared nature of the control unit and so latches and level shifting circuitry had to be added to the synthesizer.

Computer control was implemented by the author and the hardware additions and modifications are documented in Appendix A. All software is held in the Electrical Engineering Department, as is documentation produced by the author, for most of the rest of the synthesizer.

2.2.6 Real-Time Interaction

In an attempt to provide the musician with real-time feedback, a control unit has been designed to interface to the synthesizer and to vary any of the parameters on any voice in response to the user's interaction. This is being built by Susan Frykberg and will be discussed more fully in Sections 2.4 and Chapter Three.

2.3 GENERAL PHILOSOPHY BEHIND SOFTWARE

It is with software of a system that the user must interact. Therefore, it is the software that must bridge the gap between the discipline of computer technology and the discipline of the user. It must appear completely natural to use and must do the sort of job the user wants done.

In the music system the user interacts with the software via the Decwriter. Any software module can be initiated by the use of a simple command. At this point a certain operation begins. If more interaction is required, the computer will ask questions or give information or advice. In this way the user is led through the process until the task is completed and another module can be used. The action taken by some modules depends on sub-commands entered once that module has been initialized. These sub-commands must be learnt but this is not difficult as they are all short (not exceeding one character) and many similar operations within different commands are performed by entering the same sub-command. As well as this, almost all commands and sub-commands are abbreviations of the title of the operation subsequently performed. For example, both the command #S and the sub-command 'S' mean "show". This will become clear as each command is discussed in the next chapter.

Often questions are asked which require only "yes" or "no" as an answer. In this case, the user must enter "Y" (yes) or "N" (no) at the Decwriter. To make it even easier, the computer will, in certain cases, accept just a "carriage-return" and interpret that as meaning "no". Other questions require one of a list of options to be entered.

Experienced users do not want their time wasted while messages containing information they already know are printed out. A sense-switch option has been built into the software so that the computer prints out only the prompts and questions, and not the more informative messages and lists of options. This saves a considerable amount of time and paper.

It has already been mentioned that the modular structure of the software means that the various functions can be used in any order. As well as this, some modules can be operated while the user is within other modules. This provides for nesting of modules and thus increases the power of the system. Sometimes this is done automatically without the user being aware of it, but at other times the nested modules must be initialized by the user. For example, it is possible to write comments on the display screen while within the module which calculates the waveform to be loaded to a shift register, or at any other time between other modules (command #A).

All of the commands associated with the synthesizer are preceded by a hash (#) sign to distinguish them from commands used for other aspects of the computerized musicianship aids system. This is followed, in all cases, by a letter which determines which module is required. This may or may not be followed by further parameters which specify more precisely the action to be taken (for example which voice is involved or which waveshape is to be modified). Allowing for these parameters to appear in the original command speeds up the process, but makes the structure of the command more complex. In order to preserve both simplicity and speed, the computer will in some cases accept the command with or without subsequent parameters. If the parameter is not given, the computer will ask for the relevant information. Some other commands will, if not provided with a particular parameter, simply employ a default option. This is only done in cases where the user still has the opportunity to verify or alter that choice. Both of these devices mean that the same result can be obtained from a variety of commands.

The rigidity of input required by computers is a common complaint amongst users who cannot see why the machine cannot tolerate slight mistakes or variations. This problem is partially overcome by the use of questions and default options. As well as this, a musician who is uncertain how to operate the system can be guided through it, and before too long, achieve results.

All communication between the computer and the musician is expressed in musical terms, rather than technical ones, to make the system even easier to use.

2.4 SOFTWARE DEVELOPMENTS

In this section we will look at the general requirements of the software. Details of the individual modules are given in Chapter Three.

Let us imagine that the user is ready to prepare a performance which can be heard at some later time. We will assume that he has a score (recorded at the keyboard) and that he now must develop the timbres he requires and fit these into the orchestration as seen in Section 2.1.3.

2.4.1 Shape Specifying Routines

The user will probably want to begin by creating his catalogue of shapes. The shapes can be specified in either the time or frequency domain. For the user who wants to make sure that certain harmonics are present in certain proportions, the best way is to actually specify these in the frequency domain. One software option (Section 3.1.1) allows him to specify the amplitudes of up to 64 harmonically related components. A Fourier Transform is then performed and the corresponding shape drawn on the screen. In this way, a particular spectrum is described which can then be used with any note. Consequently, as with subtractive synthesis, the spectrum and the frequency are independently specified and can be varied independently. This process gives the user complete flexibility in the frequency domain, but it is only useful if he knows the harmonic content of the sound for which he is looking. This information is easily obtained from analysis of known musical sounds, but this is not always helpful as the musician may not be interested in mathematical analysis, or he may be searching for a particular sound that he has never heard before.

To increase the control the user has over the frequency domain, another option (Section 3.1.6) allows him to vary the amplitude of any of the individual harmonics of a shape. The amplitude of any component can be increased or decreased and each time a change is made, the new shape is drawn on the screen and the new sound heard (at any desired pitch). This technique can be used to vary either the waveshape or the envelope of the sound. In providing both visual and auditory feedback, it is hoped that the user will, with experience, develop a feeling for the correspondence between the spectrum, the shape and the resultant sound.

To further help the user to be able to think in terms of both the time and frequency domains, another of the software modules (Section 3.1.3) draws the frequency spectrum of any of the catalogued shapes on the screen. Thus the user can go from the time domain to the frequency domain and vice versa.

The harmonic varying routine was written to provide almost immediate feedback reflecting a particular parameter change. Although the user interacts with it via the Decwriter at present, it was this programme that was written to service the real-time parameter varying hardware mentioned in Section 2.2.6 and this will be the case as soon as the hardware is ready to be incorporated into the system. However, there is no reason why this should be restricted to varying only the harmonics of a waveshape or envelope. In fact, the other timbre parameters are even easier to change. The hardware is general purpose and the software can be extended to be similarly general. The harmonic varying routine does however show the usefulness of this type of technique and the exciting results that can be obtained from real-time interaction.

There are various routines for specifying shapes in the time domain too. The simplest of these (Section 3.1.1) allows the user to specify square waves of varying mark/space ratios and triangular waves of which the positive and negative slopes can be specified. This is a quick method for generating standard shapes, but complete flexibility is obtained in the time domain by the use of an interactive graphics system that has been developed (Section 3.1.2). Using this routine the musician can draw any shape by positioning a cursor on the screen using a joy-stick. Points can be added or removed from the drawing, and the shape formed by a linear interpolation between the points can be viewed at any stage. This process is repeated until the user is satisfied with the shape. Existing shapes can also be graphically edited by using this routine.

Any existing shape can be modified by filtering, rotating or reflecting. In all cases the user has the option of cataloguing either the original and the modified versions or just the new shape.

The facility to increase or decrease individual components has already been discussed, but it is also possible to apply a software, low-pass ideal filter to a shape (Section 3.1.5). To do this the user must specify what percentage of the components are to be removed. This process can be repeated over and over again, leaving fewer components each time, until the user is happy with the sound produced by the shape.

In the time domain, a shape can be reflected about a vertical axis to give its mirror image (Section 3.1.7) or it can be rotated (Section 3.1.8). To avoid hearing "clicks" the first (left-most) sample of each envelope is always of zero amplitude. However, since any shape can be used as an envelope, all shapes catalogued have a sample of zero amplitude in the first position. When a shape is rotated, all of the samples are shifted along until a sample of zero amplitude again lies in the first position. Mirror imaging can be used, for example, to reverse the attack and decay characteristics of an envelope, while rotating can transform an envelope that would produce a loud note with two quiet echoes into one producing a quiet note with a loud and a quiet echo. Diagrams illustrating these processes can be found in Chapter Three.

The above facilities not only allow the quick production of new shapes from old ones, but also allow the user to investigate the effects of progressive changes. For example, the use of successively filtered shapes would demonstrate a progression from harmonically complex tones to pure ones.

Any catalogued shape can be recalled and drawn on the display screen as a reminder to the user of the contents of his catalogue (see Section 3.1.9).

Whenever a shape or a frequency spectrum is drawn on the screen, a "hard-copy" version of this can be obtained for future reference. It is often valuable to be able to write comments on the screen that will also appear on the printed version. These comments may include information as to the type of sound produced, the method of production, a title and the date. An option is available (Section 3.1.4) to write comments on the display screen after they have been entered at the Decwriter.

2.4.2 Producing Sounds

The musician can at any time play an individual note at any of the 61 available pitches on any voice (Section 3.2.1). He can control whether the notes he plays are percussive or non-percussive (Section 3.2.3) and he is able to change this at any time.

Of course, no sound will be heard unless an envelope and a waveshape have been loaded into the appropriate shift registers. Any of the catalogued shapes can be loaded out (Section 3.2.2) to any of the voices as either an envelope or a waveshape.

At this stage the user can generate any waveshape he likes and listen to it being played with any envelope. While it is playing the other timbre parameters can all be altered independently. One routine (Section 3.2.4) allows the user to vary any parameter in real-time and to listen to the result. Consequently he can "play around" with sounds until he finds one he likes or he can strive after a particular sound, varying each parameter slightly until it is reached.

When playing a complete performance, the computer has control of all of the timbre parameters and so must know which ones the musician wishes to be associated with each note. A table must be built up containing all of this information, but for the user to go through each note and specify the seven parameters that go with it, is potentially the most tedious aspect of the whole procedure. To overcome this, there are eight different ways in which this timbre table can be edited. These allow the user to edit only specific notes or parameters within the table or to edit whole blocks of notes at once. These editing options are more fully described in Section 3.3.1. It is sufficient to say here that they make the editing of the timbre table a relatively painless procedure. Here again we see the computer being used to do the tedious parts of the work, so that the musician can feel at ease with the system.

To select the waveshape and envelope, the catalogue number of the appropriate shape must be entered in the timbre table. There are four choices for the tremolo waveshape and these are :

0 - no tremolo
1 - sine wave tremolo
2 - square wave tremolo
3 - sine + square wave tremolo

The tremolo amplitude ranges from 0% to 100% modulation and the frequency from 0.1 Hz to 1000 Hz. The duration of the envelope of a percussive note ranges from 40 ms to 10.24 sec in steps of 40 ms and there are 256 different volume levels.

Once the timbre table has been set up, the complete performance can be heard. Following this, the musician can change anything at all and hear the performance again.

the programmes with which the musician must interact must be carefully written to overcome this.

The general problem of making the computer approachable and useful to people who have no computer training, is one occupying a large number of researchers throughout the world, as computers are being used in an ever-growing range of disciplines and applications. Although not the main motivation for our work, this has been a very important aspect and one in which considerable progress has been made.

During 1977 work has been done to convert the synthesizer from a system with a large potential, but which was difficult to use, into one which can be used easily, and thus the potential realised. As well as this, however, the technical capabilities of the system have been considerably extended.

CHAPTER THREE

SYSTEM SOFTWARE STRUCTURE

In the previous chapter we saw how each command available to the user fits into the total synthesizer system and how they could be used in various combinations to produce a performance. In this chapter we will examine each of the software modules in more detail to see exactly what facilities are available and how these are implemented.

Each module is initiated by a short command, but its actions may depend to a large extent on the sub-commands used. For example, the user can "escape" from several of the routines by entering 'N'. Some routines enter a loop in which they ask for an option and then perform an action repeatedly. If 'N' is entered instead of one of the valid options the loop is broken. Similarly if a module is entered by mistake this sub-command may be used.

A summary of all of the commands and a brief description of their functions can be found in Appendix B.

3.1 SHAPE SPECIFYING ROUTINES

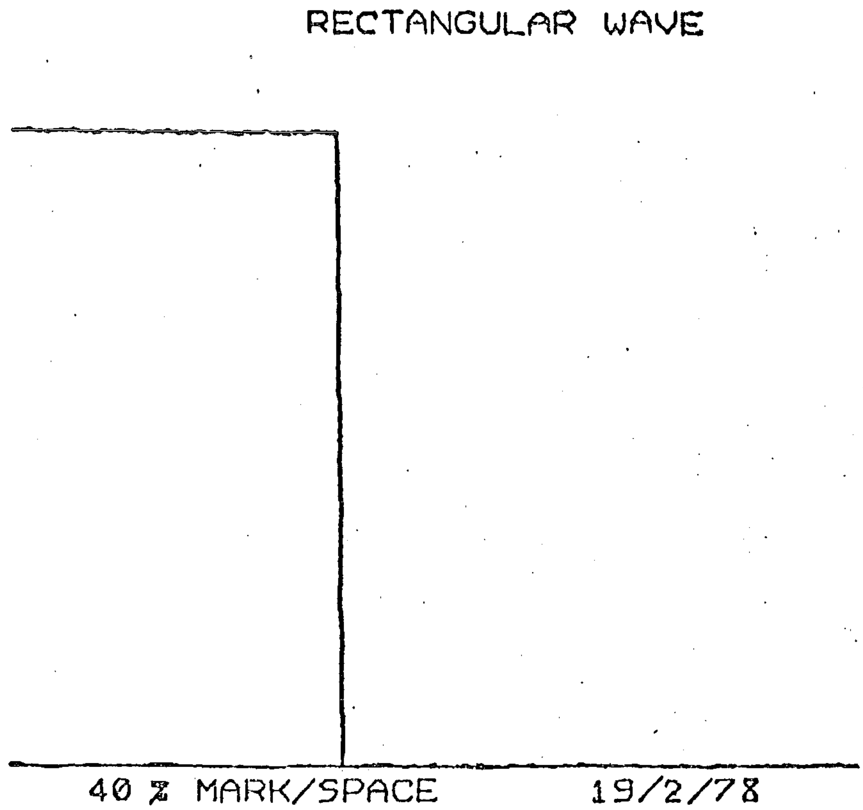
3.1.1 Shape Calculation (#C)

Any shape that can be classified as either a rectangular wave, a triangular wave, or the sum of a series of harmonically related sinusoids of zero phase can be calculated using this routine. In all cases, once the shape has been fully specified by the user, it is calculated by the computer and drawn on the display screen. This gives the musician the opportunity to see the shape and to consider whether or not it is as required. The computer asks for a catalogue number and as soon as this is supplied stores away the samples at the appropriate position in the catalogue. Once a shape has been catalogued it can be used to produce notes within a sequence (performance)(see Section 3.3), or a single note while the musician experiments with sounds (see Section 3.2). It can also be modified in a number of ways or completely replaced by another shape.

Once the shape calculation routine is initialized, the user is asked which family of shapes he requires. Providing sense switch C is not depressed, the options are listed on the Decwriter and the computer waits for the musician to respond. As soon as the user has entered either '1' (rectangular), '2' (triangular) or '3' (sum-of-sines) the computer asks for the parameters required to fully specify that type of shape.

The rectangular and triangular options offer quick but useful "standard" shapes which require only one parameter to specify them completely. If the user requests a rectangular wave, the computer asks for the mark/space ratio. This is expressed by the percentage of the total period for which the shape is "high". Two examples are shown in Figure 7. A similar parameter specifies a triangular wave. In this case, the parameter describes the percentage of the total period for which the slope of the shape is positive. (See Figure 8.) Triangular shapes are convenient approximations for many useful envelopes. For example, a "20%" triangular shape approximates the envelope of a percussive instrument such as a piano.

Because of the greater flexibility afforded by the "sum-of-sines" option, more than one parameter must be specified. The user must specify how many harmonics are to be included, and then enter the relative amplitudes of these. This type of shape is calculated using the Fast Fourier Transform (FFT) algorithm [Brigham, 1967, p.63] in order to minimize computation times. This algorithm is ideally suited to this system. Because each shift-register contains 128 samples a maximum of 64 harmonics can be specified since the waveshape must be sampled at



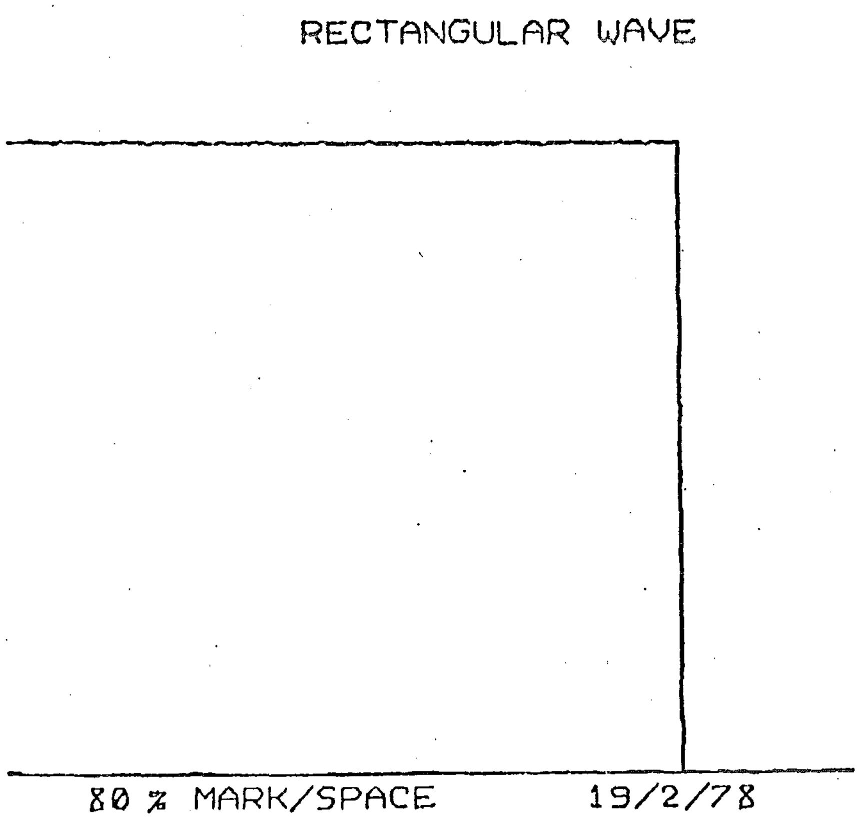


Figure 7. Sample Rectangular Shapes (#C)

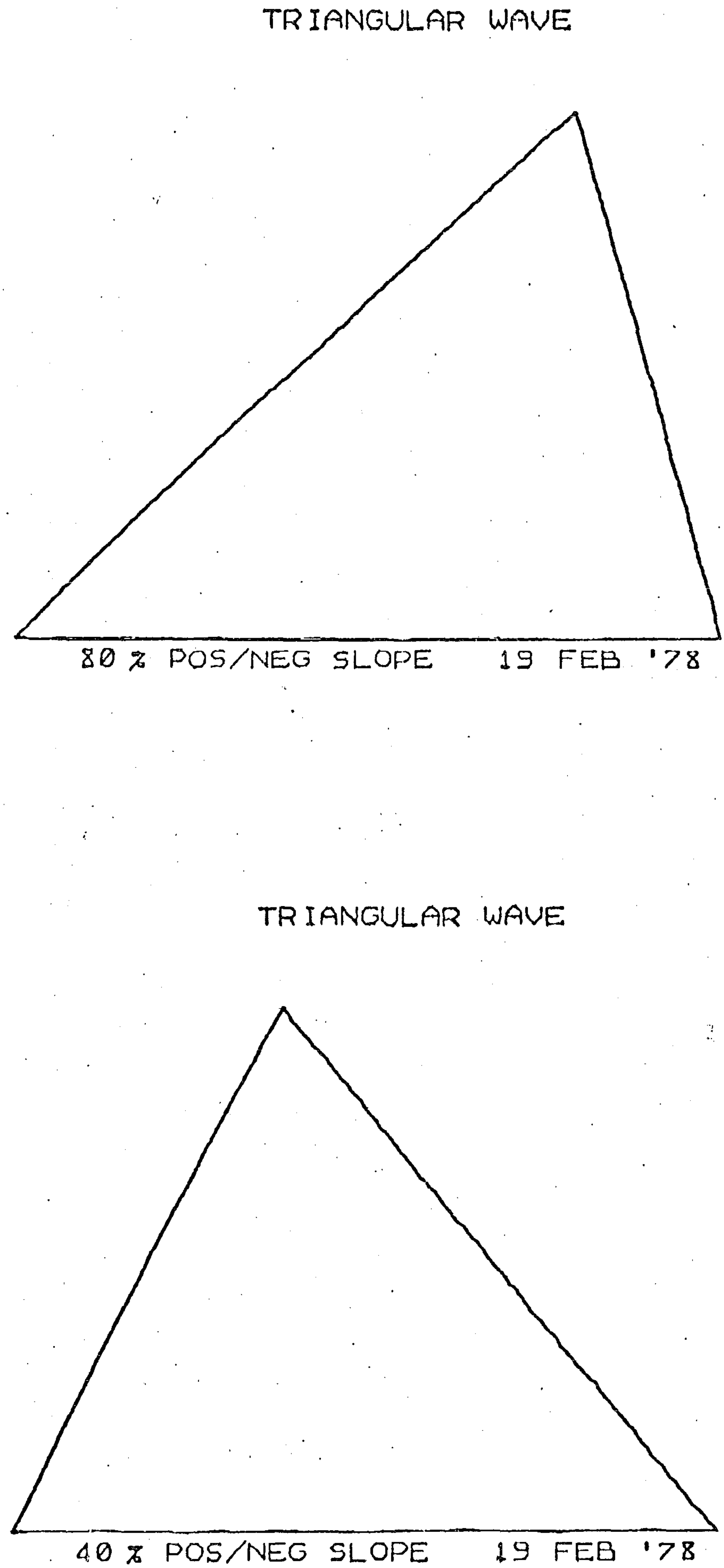


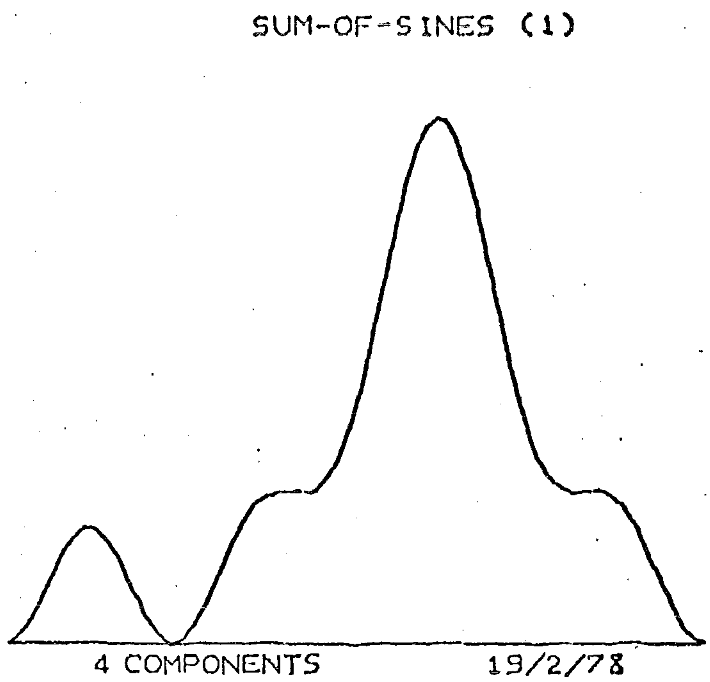
Figure 8. Sample Triangular Shapes (#C)

at least twice the frequency of the highest component. If this is not done, "fold-over" will occur which will introduce unwanted and unexpected frequencies. Using the FFT, only harmonically related components can be specified but this is not a serious restriction. It is of course possible to write a programme that will sum sinusoids of any frequencies, but this not only requires more computation time, it also takes the user a longer time as he must specify the frequency as well as the amplitude of each harmonic. A programme such as this was used before 1977 but the implementation of the FFT has reduced the computation time for a 64 component shape from about 90 seconds to less than one second and thus is obviously worthwhile. It will be noticed that the user does not have to specify the phases of the components either. Although this would, once again, have been possible to implement, it has been found that the effects of phase differences are only just detectable aurally, and therefore are of virtually no importance in comparison with other factors. [Cabot, 1976, p.568.]

If one is trying to produce a particular shape for use as an envelope the phase of the harmonics is important, but one would normally work in the time domain rather than the frequency domain to do this. Consequently, another shape specification option may be more appropriate. (For example, see Section 3.1.2.)

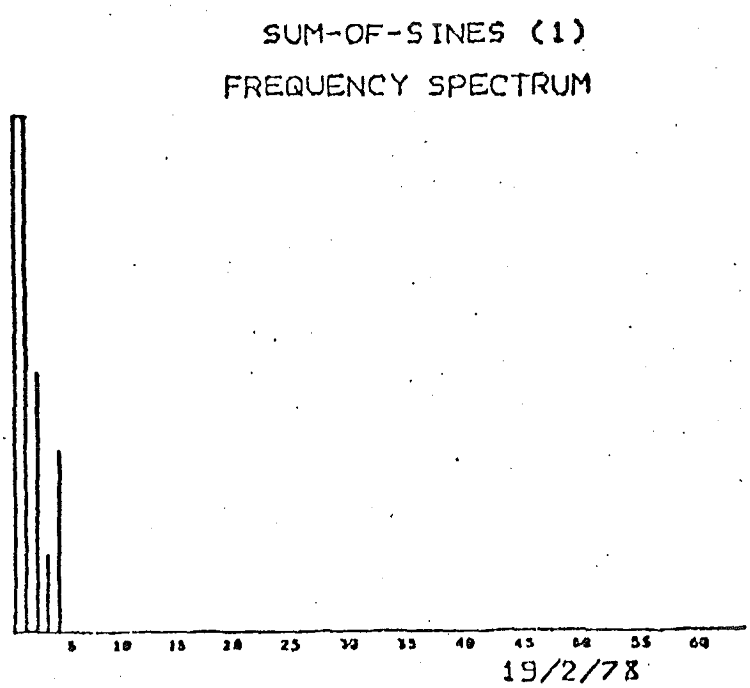
After a "sum-of-sines" shape has been calculated and drawn on the screen, the corresponding frequency spectrum is also drawn. (See Figure 9.)

It can be seen from Figures 7 to 9 that whenever a shape is drawn by the computer, all of the parameters required to reconstruct that shape are also listed. These include the mark/space ratio for the rectangular and triangular waves, and the number and amplitudes of the components for "sum-of-sine" shapes.



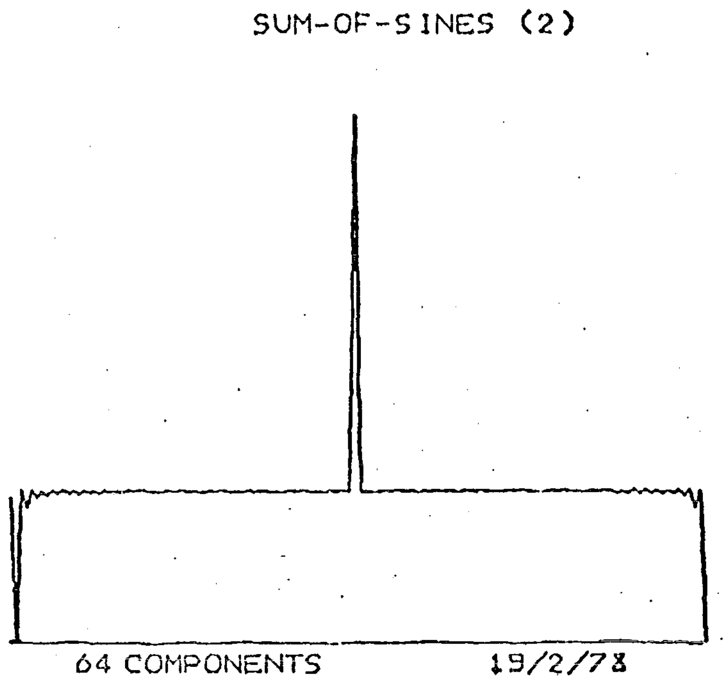
20 10 5 7

NOTICE COMPONENT AMPLITUDES
LISTED ABOVE

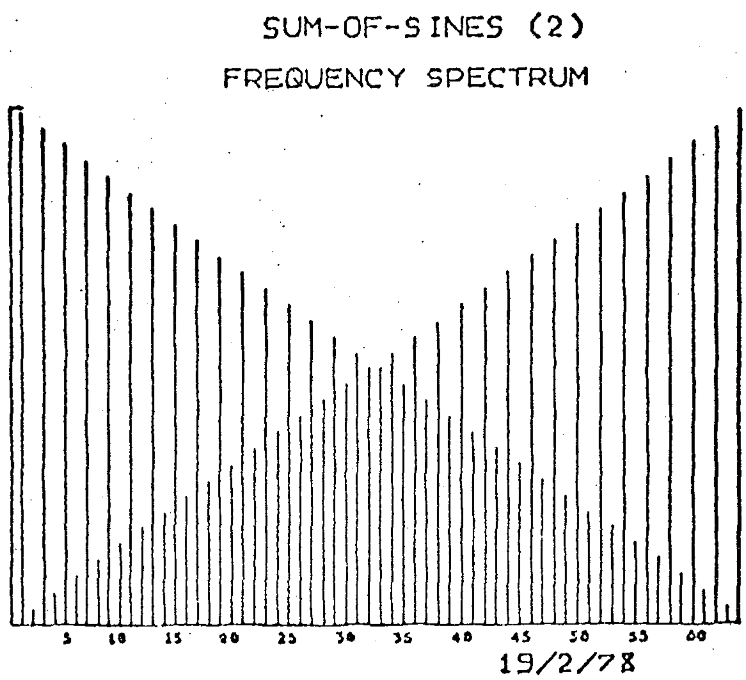


20 10 5 7

Figure 9. Sample "Sum-of-Sines" Shape (#C)



32	1	31	2	30	3	29	4	28	5
27	6	26	7	25	8	24	9	23	10
22	11	21	12	20	13	19	14	18	15
17	16	16	17	15	18	14	19	13	20
12	21	11	22	10	23	9	24	8	25
7	26	6	27	5	28	4	29	3	30
2	31	1	32						



32	1	31	2	30	3	29	4	28	5
27	6	26	7	25	8	24	9	23	10
22	11	21	12	20	13	19	14	18	15
17	16	16	17	15	18	14	19	13	20
12	21	11	22	10	23	9	24	8	25
7	26	6	27	5	28	4	29	3	30
2	31	1	32						

Figure 9. (Cont.) Sample "Sum-of-Sines" Shape (#C)

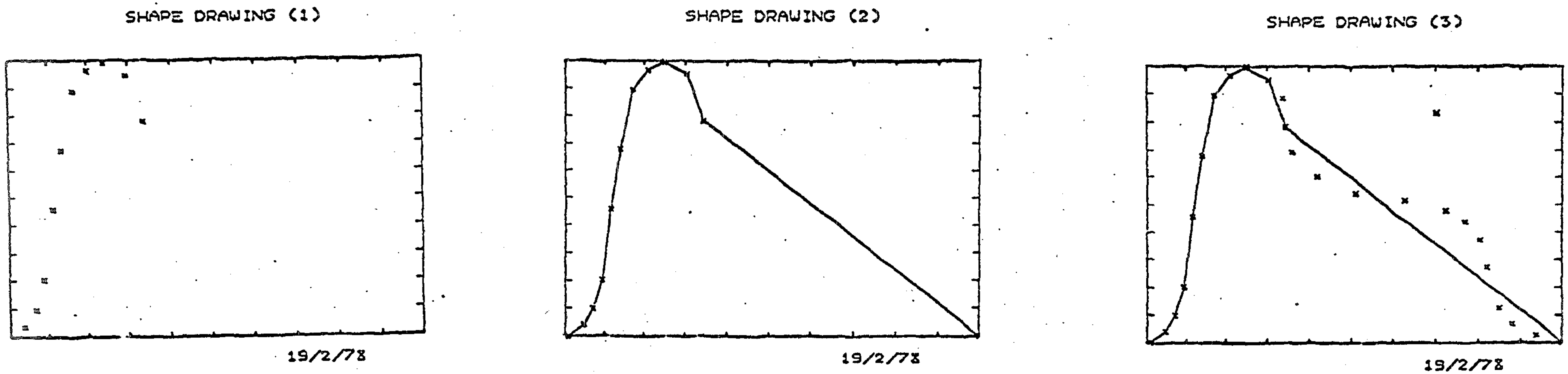
3.1.2 Shape Drawing (#D)

To give the user complete flexibility in the time domain an interactive graphics system has been developed which allows him to quickly specify shapes by drawing them. This replaces a hardware "specifier" [Vaughan, 1975] which consisted of a perspex tablet on which the required shape was drawn, and which was passed under an array of phototransistors. The shape was thus transformed into digital data and transferred to the computer. This method was cumbersome, slow and unrealiable. In contrast computer graphs are quick and easy.

The user interacts with the computer by manipulating a joy-stick which moves a small spot of light around the display screen. It is possible to draw a shape by moving the joy-stick while the computer keeps a record of its path. However, this produces very jagged shapes due to difficulties in co-ordinating the spot on the screen and the hand on the joy-stick, and also due to small involuntary hand movements. If the line is drawn immediately on the screen it cannot (being a storage scope) be updated quickly if a drawing error is corrected. If not drawn immediately, the user cannot see what he has done until he has finished.

To overcome these difficulties a system was designed whereby the user could position the joy-stick before telling the computer to record its co-ordinates. Therefore, by positioning the joy-stick and entering a "carriage-return" on the Decwriter, the user can ensure that only those points which are in exactly the right place are recorded. This eliminates errors due to unintentional hand movements. For each point recorded, the computer draws a small cross on the screen. This feedback allows the user to position subsequent points in relation to those already recorded (see Figure 10(a)).

A linear interpolation between each successive pair of points is used to create the final shape. This produces an essentially smooth



POSITION POINTS ON SCREEN

(a) 1

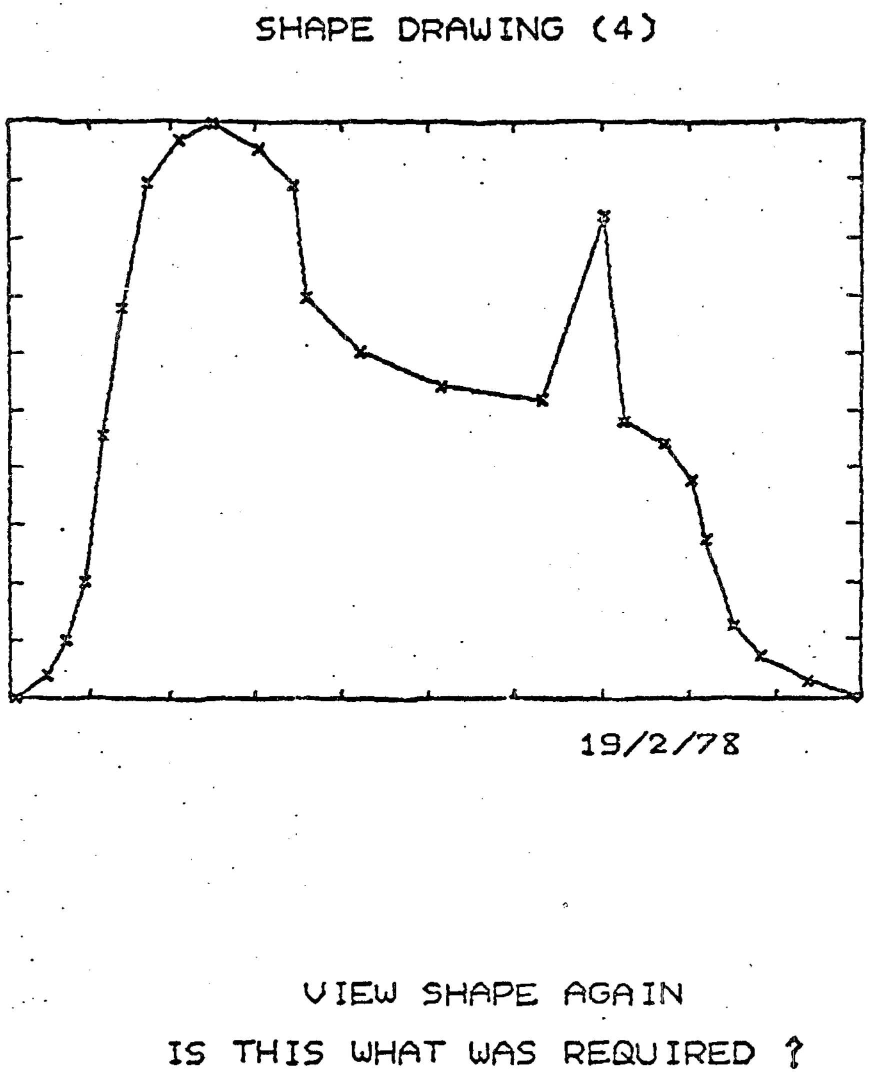
VIEW CURRENT STATUS OF SHAPE
FORMED BY LINEAR INTERPOLATION

(b) 2

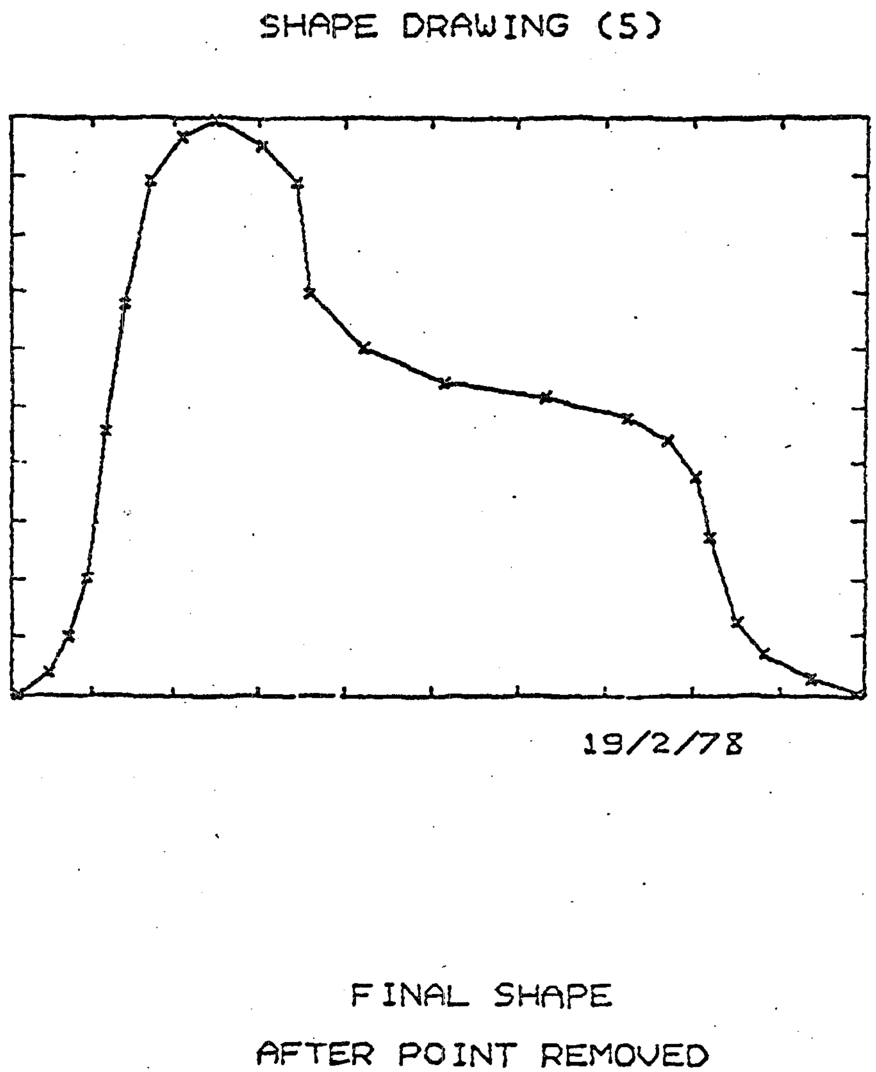
CONTINUE ADDING POINTS

(c) 3

Figure 10. Steps Required to Draw Shape (#D)



(d) 4



(e) 5

Figure 10 (Cont.) Steps Required to Draw Shape (#D) curve compared with the jagged path taken by the joy-stick. The user can view the curve at any stage by entering the sub-command 'S' (show) (Figure 10(b)), and can then, if he wishes, continue adding points (Figure 10(c)).

Points can also be removed, as can be seen by comparing Figures 10(d) and 10(e). To do this, the joy-stick is positioned near the point to be removed, and an 'R' is entered at the Decwriter. As the point nearest to the cursor is removed, the user does not have to indicate the offending point exactly.

The sub-command 'A' allows the user to write comments on the screen (as described in Section 3.1.4) at any stage in the drawing process.

When the user is satisfied with his shape, the sub-command 'N' causes the computer to calculate the values of all of the samples between the specified points and to store the shape in the catalogue.

The 'D' command can also be used to graphically edit existing shapes. The user can specify in the initial command (see Section 3.1.10) that he wants a particular catalogued shape to be drawn on the screen when this module is initialized. As long as this is retained on the screen, it can be used as a guide for drawing a new, similar shape.

3.1.3 Spectrum Calculation (#U)

The Fourier option of the shape calculating routine allows the user to specify a shape in the frequency domain and to see the time domain representation drawn on the screen. To go from the time domain to the frequency domain the musician must use another command which will analyse any of the catalogued shapes and then draw the corresponding spectrum on the screen. Some examples of shapes and their spectra, found using this routine, can be seen in Figure 11. The shape in

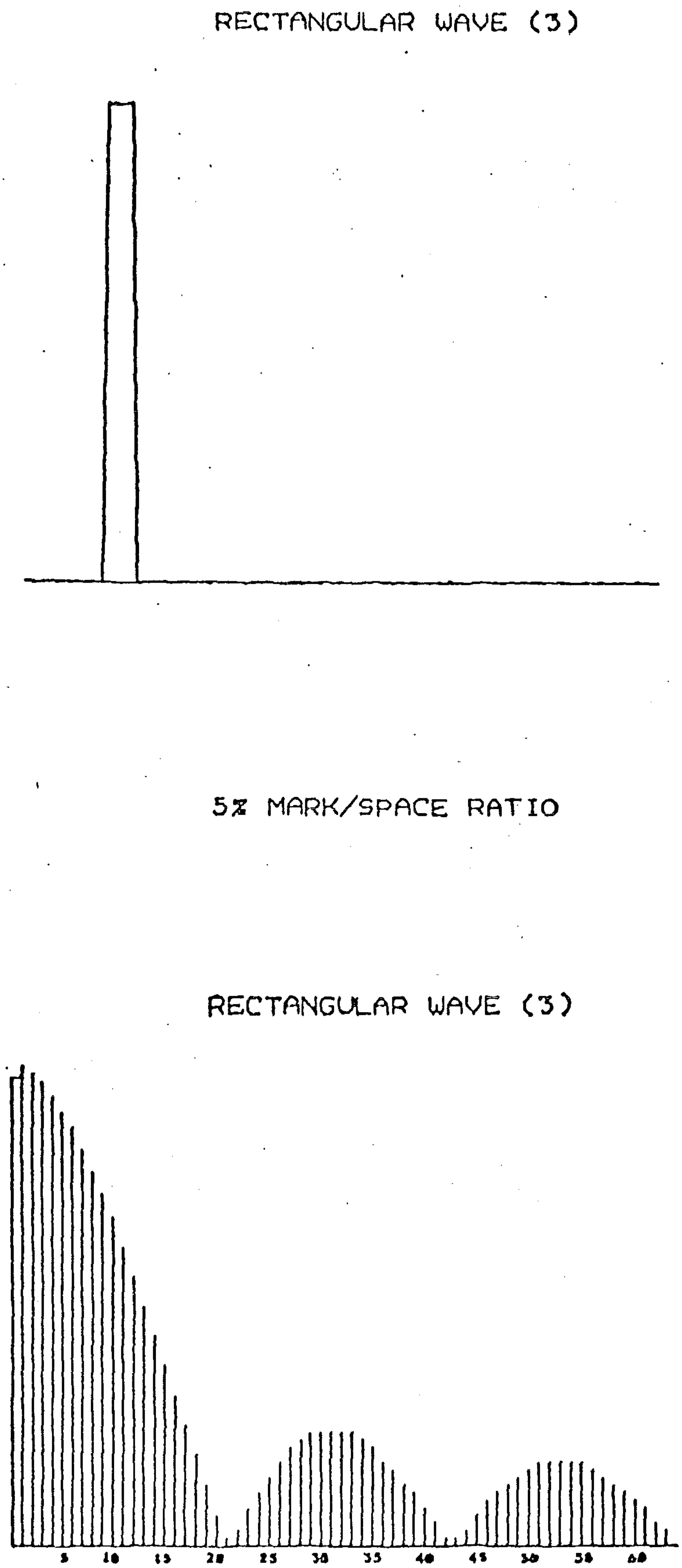
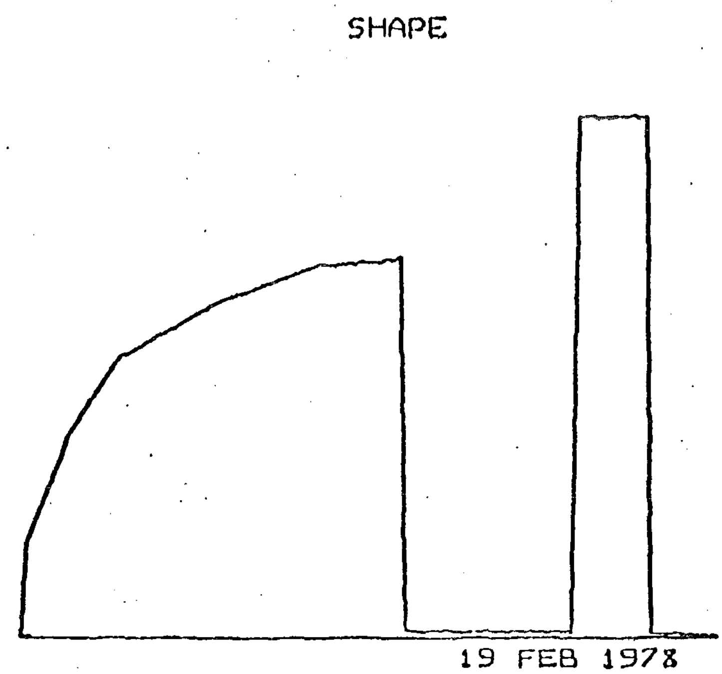


Figure 11(a). Calculated Shape and Spectrum (#U)



PRODUCED USING DRAWING ROUTINE
NOTICE STRAIGHT LINE SEGMENTS

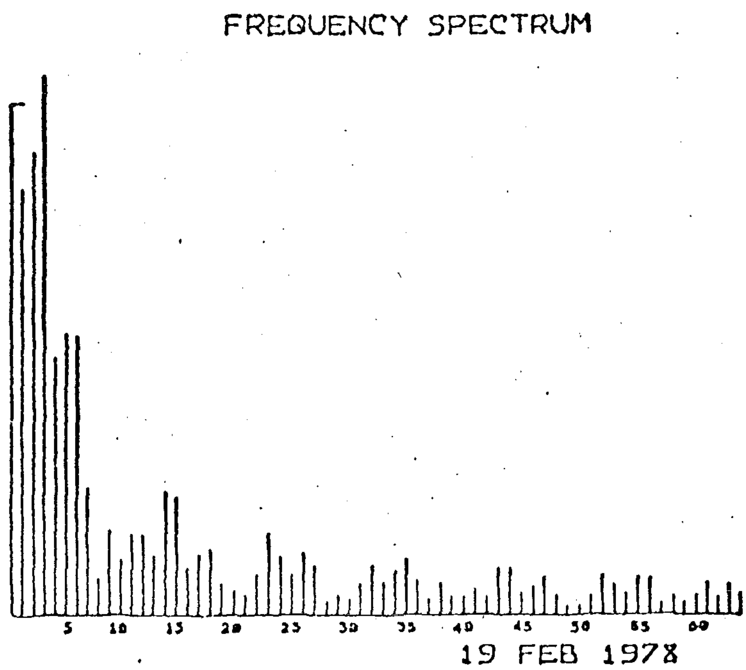


Figure 11(b). Drawn Shape and Spectrum (#U)

SPECTRUM OF CATALOGUED SHAPE
SHAPE PRODUCED BY DRAWING ROUTINE

Figure 11(a) was produced using the shape calculating routine (Section 3.1.1) and that in Figure 11(b) by using the shape drawing routine (Section 3.1.2).

As well as helping the musician to appreciate the correspondence between the time and frequency domains, this routine allows him to analyse a shape he is using, so that he can, for example, change it slightly using the component varying routine (Section 3.1.6) or create a new, similar one using the shape calculating routine (Section 3.1.1).

3.1.4 Routine to Add Comments (#A)

It is quite possible that a user may wish to keep a record of those shapes (and their spectra) that produce particularly interesting sounds. Whenever a shape is produced it is drawn on the screen along with all relevant information such as the mark/space ratio or the amplitudes of the components. A printed record of this can be obtained, but the user may also require other information such as the nature of the sound produced by the shape, when used in a particular way. Comments of this nature can be typed at the Decwriter and transferred to pre-specified positions on the screen. Nothing is written on the screen until a "carriage-return" is entered, terminating the comment. In this way the user has the opportunity to correct errors before they appear on the screen.

Positions are allocated on the screen for a title, two lines of comments, the date and a reference number. The computer writes out prompts on the Decwriter indicating what information is required, and tells the user what the maximum number of characters allowed in any message is. If this number is exceeded, no action is taken and the prompt is repeated.

Each comment is centred within the space available to it on the screen to maintain symmetry irrespective of the length of the message. Figures 11(a) and 11(b) show examples of these comments.

3.1.5 Shape Filtering Routine (#I)

There are various ways in which the user can modify existing shapes, in both the time and frequency domains.

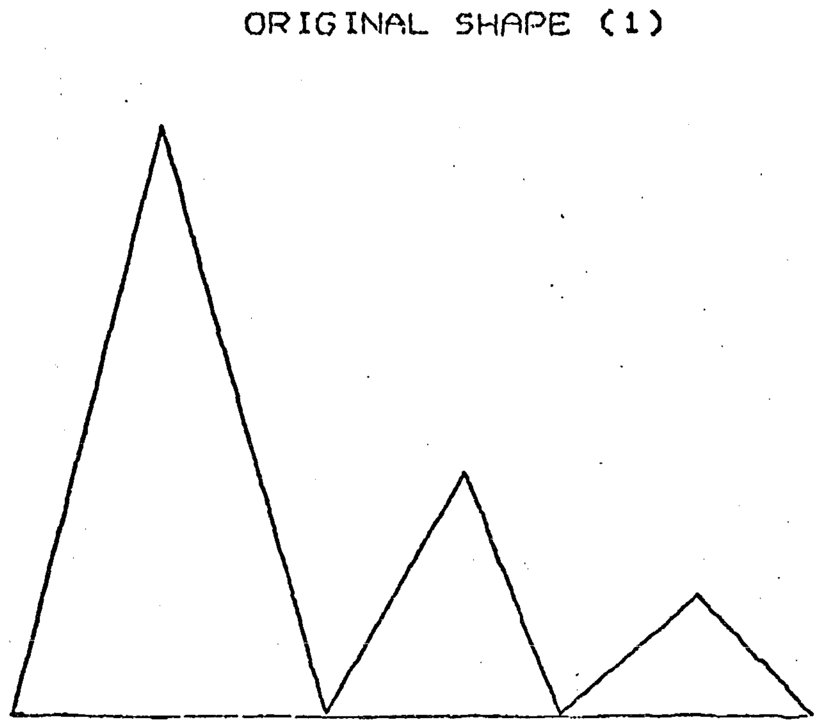
A software, ideal low-pass filter can be applied to any of the catalogued shapes. The user must specify what percentage of the harmonics are to be retained. Using the FFT algorithm the rest of the components are removed and the new shape is calculated from the remaining harmonics and displayed on the screen. If the shape has not been sufficiently filtered, the process can be repeated until the user is satisfied. This routine was not written by the author and has only been modified to the extent that any of the catalogued shapes can now be filtered, and not just the last one calculated, as was the case previously.

Some examples of filtered shapes can be seen in Figure 12 along with the originals. This routine is particularly useful in removing the corners from shapes that result from the drawing routine.

3.1.6 Frequency Component Variation (#V)

As well as removing harmonics, the musician may want to add in components or increase or decrease the amplitudes of those already present. The routine that makes this possible has the added advantage of allowing the user to vary any component, and not just the high frequency harmonics, as with the filtering routine.

After each change, the new shape is drawn on the screen and a note is played (at any specified pitch) using the new shape. Both the envelope and the waveshape can be varied in this manner. Thus the



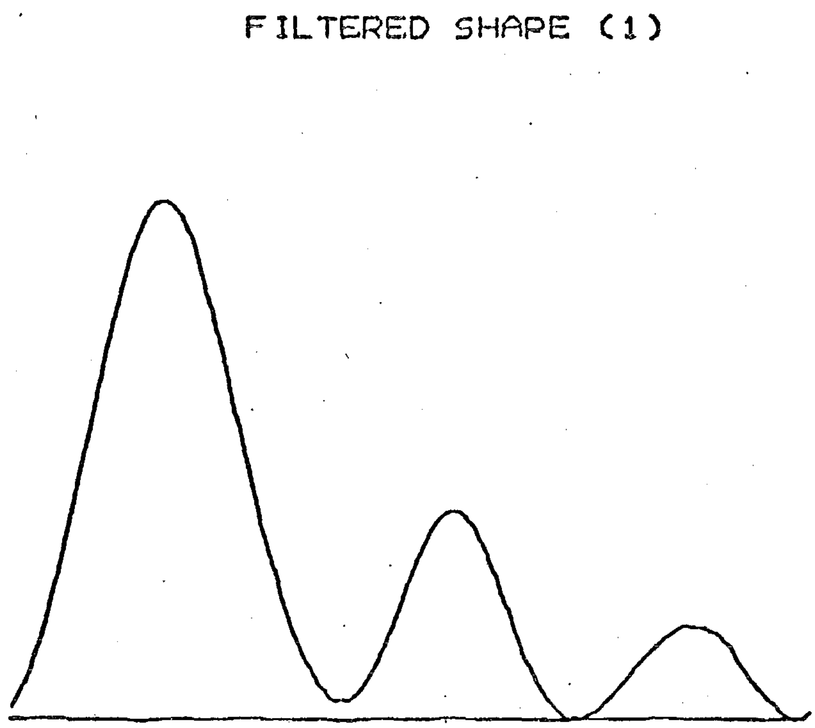
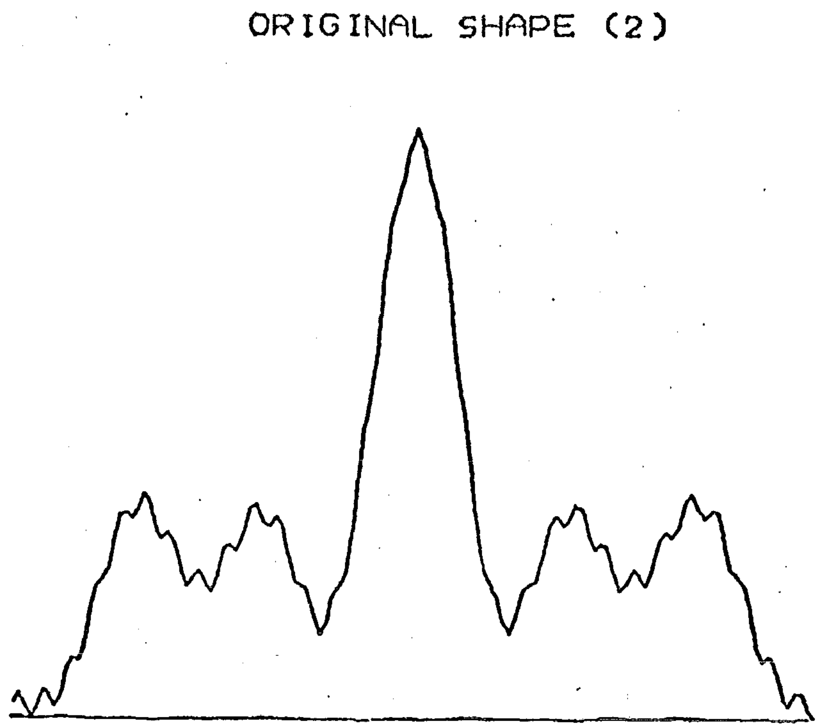


Figure 12. Filtered Shape (#I)



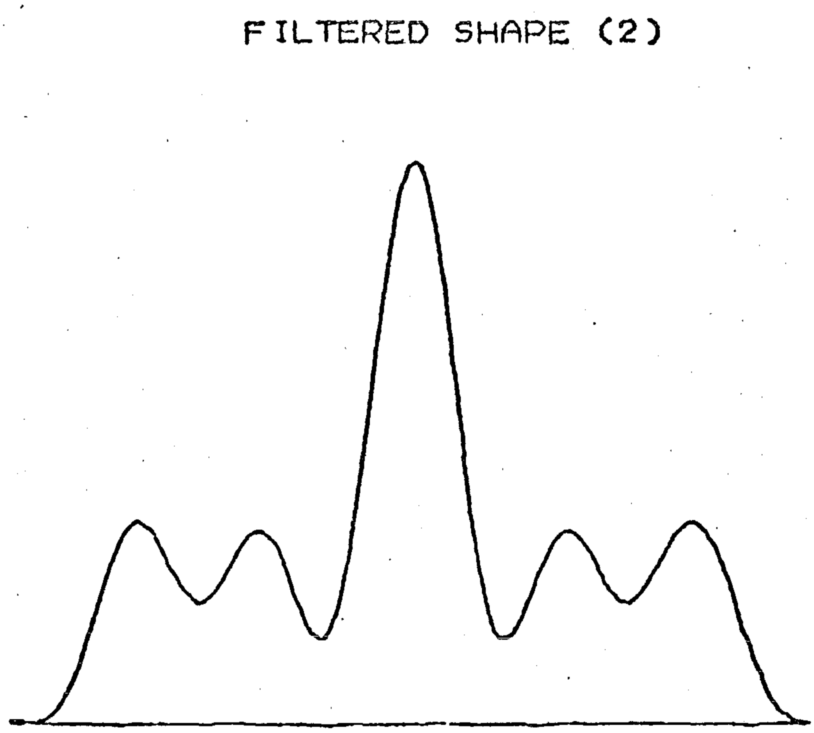


Figure 12 (Cont.) Filtered Shape (#I)

user has almost immediate visual and auditory feedback, illustrating the effects of essentially continuous changes. Figure 13 illustrates how a shape composed of first harmonic only can be slowly varied to be composed of third harmonic only.

The user must specify whether the shape is to be considered as a waveshape or an envelope, which harmonic is to be increased (or decreased) and the amount by which it is to be varied.

The amount by which a component is varied is defined as a percentage of the maximum amplitude of a shape (which is normalized to 8 bits). For example, an increment of 100% means that a sine wave of the required frequency and of an amplitude equal to that of the present shape, is added to the present shape. The new shape, composed of equal quantities of the old shape and the new harmonic, is then scaled to maintain the standard maximum amplitude. It would be easier to keep track of the amplitudes of the components present if the amount of change in any component could be expressed in terms of the current amplitude of that component, rather than the sum of the amplitudes of all the components. However, it has been found that the musician can more easily think of the amount of change in relation to the total sound heard than in terms of one component of that sound. Furthermore, to have made changes on the basis of the amplitudes of the individual components would have required (because of the scaling of the shape after each change) two FFT's per variation. This would have taken too long, and meant that the feedback was far from real-time. Under the present scheme, one FFT is performed when the routine is initialized in order to calculate the phases of all of the components in the original shape, and then a sine wave of the appropriate frequency is simply added to the existing shape each time a change is made. To avoid calculating the sine values each time, these are calculated only once, when the routine is initialized, and are stored in a table for later use. This makes the process even faster.

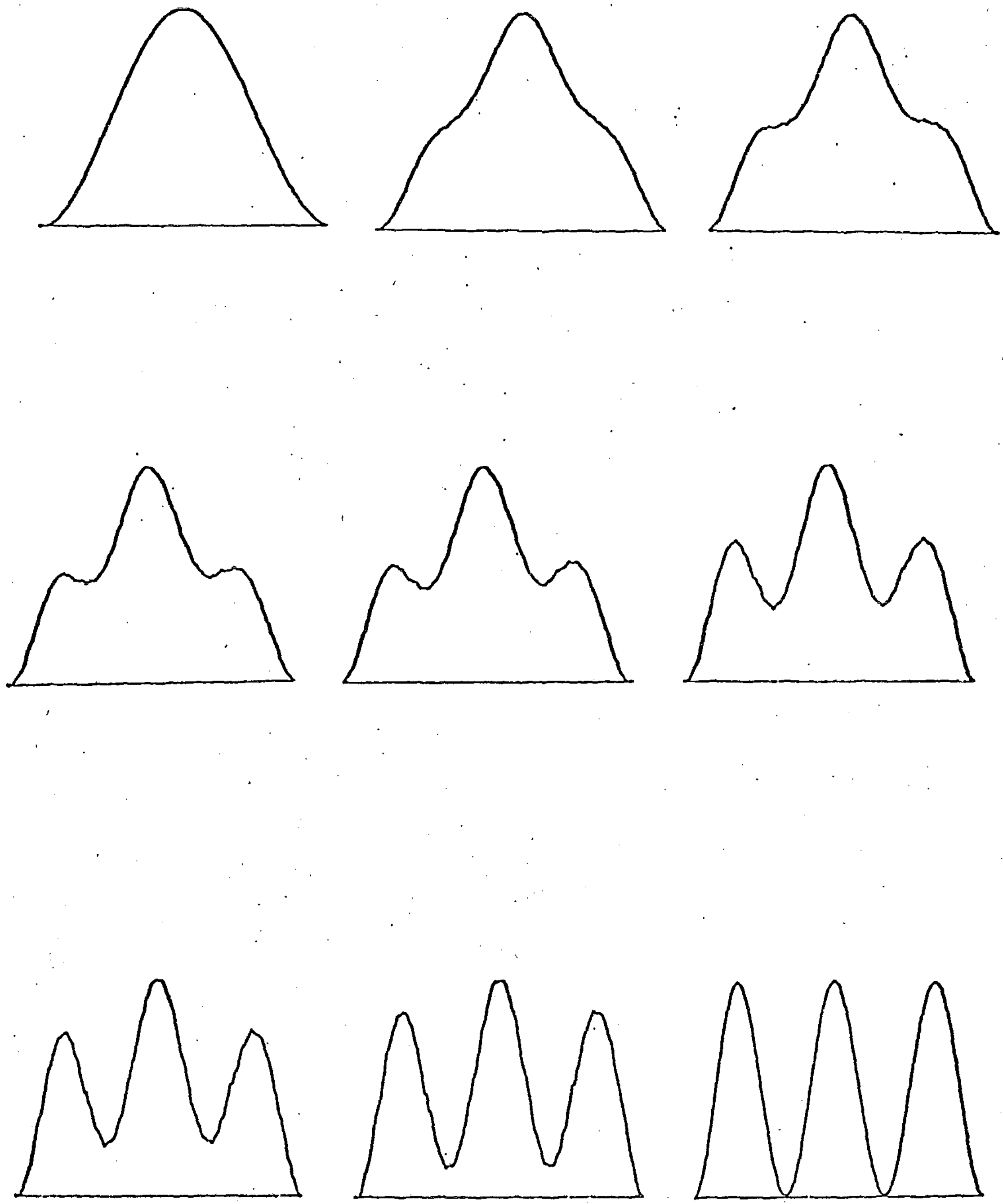


Figure 13. Harmonic Variation (#V)

The phase of each sine wave added is taken to be the same as the phase of that component (if present) in the original shape. If this was not done, the amount of change would be indeterminate. As new components are introduced, the values of the samples change. When the first sample becomes non-zero the shape must be rotated automatically (see Section 3.1.8) until it again begins with a sample of zero amplitude. The phase relationships of the components must be maintained even when this is done.

This routine can be used to vary existing shapes as well as producing completely new ones.

This facility, enabling the musician to hear immediately the effects of changes in one parameter, has been found to be useful and exciting. The immediate feedback encourages the musician to experiment with sounds. There is no reason why the other parameters of the sound cannot be varied in the same way and programmes to facilitate this would be worthwhile.

3.1.7 Shape Mirror Imaging Routine (#M)

In the time domain a shape can be modified by either reflection or rotation. An example of a reflected shape can be seen in Figure 14. The mirror image of the shape is produced and thus the characteristics of the attack and of the decay are interchanged.

Since the harmonics of the shape are not altered by reflection, the effect of this modification will only be noticed when applied to envelopes and not to waveshapes. Nevertheless, it does give the user the ability to quickly develop new envlopes from existing shapes.

3.1.8 Shape Rotation (#R)

Rotating a shape is also only useful in the production of envelopes. The computer scans the shape to be rotated from left to

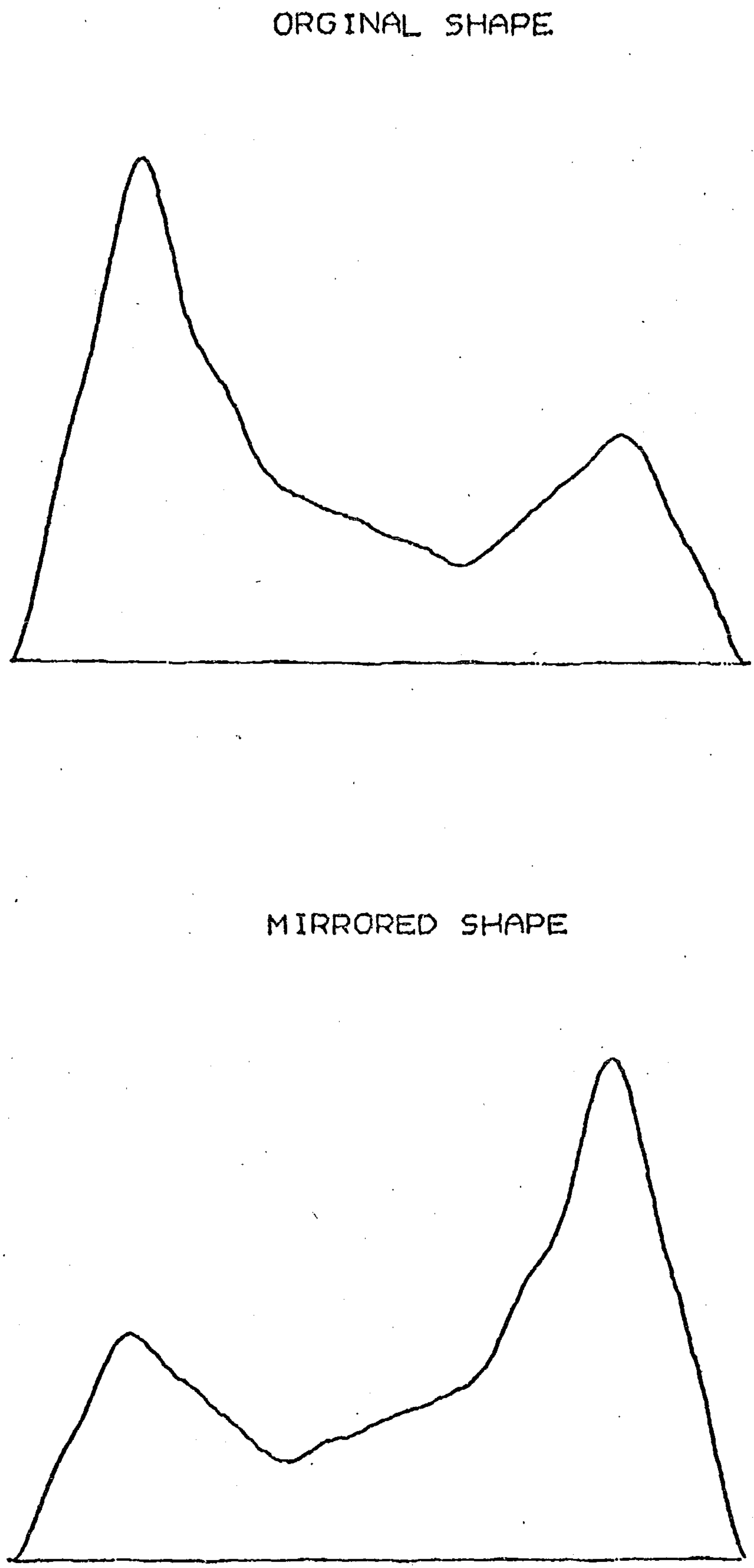


Figure 14. Reflected Shape (#M)

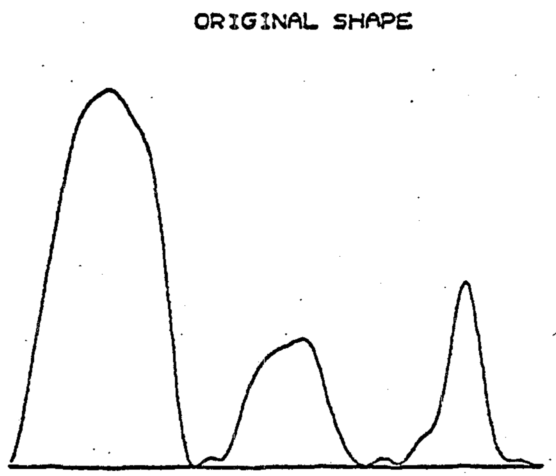


Figure 15. Rotated Shape (#R)

right until a sample of zero amplitude is located. This is then transferred to the position of the first sample and all of the other samples are moved to maintain their original position relative to the moved sample. In other words, all of the samples are shifted left, until one with zero amplitude lies in the left-most position. Figure 15 shows the results of this process and illustrates how a variety of envelopes can be created from the one original.

3.1.9 Shape Viewing (#S)

To remind himself of the contents of a particular catalogue location, the user can have the shape in that location drawn on the screen by entering the #S command. The information regarding the mark/space ratio and Fourier component amplitudes is not listed, as this information is not stored by the computer. The user can, however, still add comments to the screen.

Like the filtering routine, this one was not written by the author, but has been modified so that any shape can be redrawn and not just the most recent one produced.

3.1.10 Shape Specifying Routine Parameters

Most of the routines discussed so far produce new shapes which must be stored in the catalogue, and many of them can be used to modify existing shapes. In either case, the catalogue number of the existing shape and/or the catalogue location that is to contain the new shape can be specified in the original command.

The general form of these commands is"#Y,C1,C2" (where Y represents any of the shape specifying routines, C1 is the catalogue number of the existing shape and C2 specifies the location in which the new shape is to be stored). Several variations of this are available.

Consider, for example, the shape drawing command. If one simply wants to draw a new shape the easiest thing to do is to enter just "#D", in which case the computer will ask where the shape is to be stored after it has been drawn. If, however, he wishes to modify an existing shape, this must be specified in the command. When "#D,C1" is entered, the computer draws the shape already stored in catalogue position C1 on the screen and stores the modified version in the same location when the modification is complete. If, on the other hand, two similar shapes are required, one being a modification of the other, the user can enter "#D,C1,C2" whereupon the old shape would stay in position C1 and the new one would be stored in position C2.

The component varying routine is similar in that it always produces a new shape, but may or may not use an existing shape. The filtering, mirror imaging and rotating routines are slightly different in that they must operate on a shape that has already been catalogued. In these cases if C1 is not specified, it is assumed that the shape to be modified is the last shape that was used. In this way, a whole sequence of operations can be applied to a single shape without having to specify it each time.

The shape showing and spectrum finding routines are different again in that they must have an existing shape on which to operate, but they do not produce new shapes. In these cases C2 is not required, and C1 may or may not be present as in the previous example.

The shape calculating routine is a special case in that it cannot use an existing shape in any way. The two valid forms of this command are therefore "#C" and "#C,C2". If "#C" is entered, the computer will ask where the shape is to be catalogued.

The variations permitted within these commands give the user considerable freedom in the manipulation of shapes and allow him to specify exactly what he wants to do.

3.2 SOUND PRODUCTION

3.2.1 Note Playing (#P, V, P)

In order to be able to experiment with sounds, the user must be able to play any note he wishes and vary any of the associated parameters. To play a note the user must specify the voice on which it is to be played and the pitch. Any voice (1-4) can be used providing it is free (i.e. it is not being used already by another note). The pitch is designated by a number from 16 ('C' two octaves below 'middle C') to 76 ('C' three octaves above 'middle C'). The larger the number, the higher the pitch. 'Middle C' is designated by the number 40.

When the command is entered the note begins and, providing it is not percussive (see Section 3.2.3), it continues playing until told to stop by the user. A note is stopped by repeating the #P command with the pitch set to zero (i.e. #P, V, Ø).

This routine existed before 1977.

3.2.2 Shape Loading (#L, V, C)

Since two of the main timbre parameters are the waveshape and the envelope, the musician must be able to load any of the shapes he has developed into the shift registers of the synthesizer. In fact, unless both the waveshape and envelope shift registers of a particular voice are loaded with a shape, no sound can be heard from that voice.

Any shape can be loaded out to any voice as either a waveshape or an envelope. When the command, which contains the shape catalogue number and the voice number, is entered by the user, the appropriate shape is drawn on the screen and the computer asks if it is to be used as an envelope or waveshape. The user responds with an 'E' or a 'W' (any other character causes the question to be repeated) and the shift register is immediately loaded. The waveshape and the envelope are the only timbre parameters that cannot be changed while a note is playing - they must be loaded out before the note begins.

3.2.3 Envelope Status Control (#E)

Whether a particular note played by using the #P command is percussive or non-percussive is controlled by the user. When the command"#E,P" is entered, all of the subsequent notes are percussive. When "#E,N" is entered the following notes are non-percussive:

This control is more important that it might at first appear. The effects of different envelopes and of varying the envelope clock rate can often only be heard if the note is played right through without a pause. On the other hand, a long steady-state portion is needed if one wants to listen to various tremolo combinations. With non-percussive notes, it often appears that the waveshape used is the prime factor in determining the timbre, but as soon as percussive notes are heard, the effect of the envelope becomes more noticeable.

3.2.4 Timbre Parameter Variation (#T)

This routine is used to vary all of the timbre parameters except the waveshape and the envelope. Having been established, the timbre parameters remain unchanged until altered by the user. Therefore, although they can be varied within a note, they can also be set up before a note is played and will apply to all subsequent notes.

When the user enters the routine, he is asked on which voice the changes are to be made, and then which parameter is to be varied. He may enter :

T - tremolo waveshape
A - tremolo amplitude
F - tremolo frequency
E - envelope clock rate
G - gain (volume)
V - change voice number
N - no more.

If 'N' is entered, this routine is terminated and the computer waits for the next command. If 'V' is entered, control returns to the initial question so that the user can change to another voice. If any of the other options is entered, the computer asks what value is to be loaded out for that parameter. When the loading has been completed, the computer asks for another value for the same parameter, allowing the musician to experiment with a single parameter without having to specify that parameter each time.

When the musician wants to vary another parameter he enters 'P' instead of a value, the computer asks which parameter is required, and the process is repeated.

The fact that all parameters are independent and can be varied independently means that any combination is possible. There are theoretically 2.8×10^{14} $(2^8)^{16}$ combinations (including waveshapes and envelopes) available at present, although it is unlikely that all of these are distinguishable. This does, however, give us some idea of the flexibility of the system and the amount of control available to the user. This total can be extended to 6.9×10^{16} if we remember that any of these combinations can be applied to notes of any of the 61 available pitches and one, two, three or four voices can be used simultaneously.

3.3 PLAYING NOTE SEQUENCES (#Z)

Sequences of notes (and hence, complete performances) can be played and controlled by the computer which will automatically change all of the timbre parameters with speed and precision. However, before this can be done, the user must decide what parameters are to be associated with each note. Even before this can be done, the user must have his score and his catalogue of shapes. The score would normally be obtained by recording input played at the organ keyboard. This recording is performed by using one of the commands of another of the aspects of the computerized music system [Tucker, 1977], and the notes can be edited to eliminate any errors or to introduce particular effects. An alternative means of specifying sequences of notes is to enter the required pitches and durations from the Decwriter at the beginning of the note sequencing routine, but this facility would never be used for long or serious performances. Its purpose is to allow the user to experiment with particular combinations or sequences of notes. We have already discussed how the catalogue is established.

3.3.1 Establishing the Timbre Table

The only remaining task, before the performance can be heard, is to fill in the timbre table. This contains the values for all of the timbre parameters associated with each note. A print out of a timbre table can be seen in Figure 16.

To have to type in every entry in this table would be extraordinarily tedious, especially if the performance consisted of several hundred notes. To overcome this, there are various options available which allow the user to edit all or parts of the table in a number of different ways. These will be described individually but it should be remembered that values are entered in this table not only when it is being set up originally, but also when the musician wishes to change parts of it.

The most obvious, but as mentioned the most tedious, way to fill in this table, would be to go through every note and type in the seven parameters required. This is made possible by option one. As with all of the other options, the computer writes the number of the note to be edited in the first column and then the Decwriter moves across to each of the other columns in turn, and waits for the appropriate parameter to be entered. If an invalid character or value is typed in, the computer simply outputs a "line-feed" (i.e. moves the paper on one line), returns the Decwriter to the same column and waits again.

NOTE	W/S	ENV	TREM TYPE	TREM AMP	TREM FREQ	ENV RATE	VOLUME
1	9	2	0	63	12	254	0
2	34	2	0	63	12	254	10
3	9	2	2	63	174	254	20
4	34	2	0	63	12	254	30
5	9	2	0	63	12	254	40
6	34	2	0	63	12	254	50
7	5	3	2	20	212	254	60
8	34	3	2	20	212	254	70
9	5	3	2	20	212	254	80
10	34	123	2	20	212	254	90
11	5	123	0	20	8	254	100
12	34	123	2	20	212	254	110
13	5	123	0	20	1	254	120
14	34	123	0	63	12	254	130
15	9	2	0	63	12	254	140
16	34	2	0	63	12	75	150
17	9	2	0	63	12	75	160
18	34	2	0	63	12	75	170
19	9	2	0	63	12	75	180
20	34	1	3	63	47	180	190
21	78	1	3	63	47	180	200
22	78	1	3	63	47	180	210
23	78	1	3	63	47	180	220
24	9	2	0	63	12	254	230
25	9	2	2	63	210	254	240
26	34	207	1	40	20	46	250
27	9	2	0	63	12	254	99
28	9	2	0	63	12	254	31

ANY MORE EDITING ?

Figure 16. Sample Timbre Table

Editing is made more convenient if the parameter values for only a block of notes, rather than the whole performance are entered. Option three allows the user to enter the numbers of the first and last notes in the block, and the computer will then step through these notes in exactly the same way as in option one.

If the notes to be edited do not form a block of consecutive notes but are scattered throughout the piece of music, these can be edited by using option two. This allows for the editing of individual notes. The user must enter the number of the note and the Decwriter will then step across to each of the columns and accept the parameter values. It then returns and waits for another note number. When the user has finished with this option, he enters 'N' instead of a note number and can then select another option.

One very useful option (option four) can be used to edit blocks of notes when the same changes are to be made to each note within that block and the parameters therefore only need to be entered once. An example of its usefulness is its ability to enter values for every note in the performance (if the user defines the block to be edited as containing every note). Of course, all of the notes would have the same parameters, but those which were to be different could then be changed using other options.

If the same changes are to be applied to several notes, but once again the notes are scattered and not in a continuous block, option seven should be used. This allows the values to be typed in and then the notes to be changed to be numbered by the user. This would be useful if, for example, the musician was orchestrating a choral piece and wanted all of the tenor notes to have the same timbre. The tenor notes might be notes 2, 4, 7, 12, 13, 16 . . . . . ., and these numbers could be typed in after the parameters had been entered. It is envisaged that in the future the notes to be edited could be selected by pointing to them on the screen using the joy-stick. This is not possible at the moment but as a compromise the computer helps the musician select the correct notes. When a note number is typed in, the computer writes out the pitch of that note. If this is confirmed as the right note by the user, the editing is performed. If not, he must try again.

As well as wanting to select individual notes, the user may want to edit individual parameters. For example, he may only want to alter the tremolo frequency or the volume. Option six allows the user to select which parameters are to be changed. This must then be followed by one of the other options which operates in the normal way except that only the selected parameters are affected. For example, option six followed by option four could mean that the tremolo amplitude and envelope clock rate of all of the notes from note 16 to note 41 would be changed to the same values. All of the other parameters of these notes would be unchanged.

Option five causes blocks of the timbre table to be printed out on the Decwriter so that the user can see the current state of the table.

3.3.2 Playing the Performance

As soon as the timbre table has been filled in, the performance can be heard. During the performance the computer must start and stop the notes and load out the timbre parameter values at the correct times.

Before a performance starts, it is not known on which voice a particular note will be played. This depends on which voices are free at the time that the new note is to be started. To complicate matters, a voice may still be busy (after the computer has turned off the note being played through it) if the envelope clock rate for the note is low and the decay takes some time to be completed. Consequently, when playing a sequence of notes, the computer is constantly testing the status of the voices so that it knows which ones are available and can be used to play the next note. If more than one voice is free, the one with parameters most similar to the parameters of the new note will be selected. To save time, only those parameters which must be changed are loaded out. If no voice is available, the note is delayed until a voice does become available.

To reduce delays in starting a note, the tremolo parameters can be loaded out after the note has been turned on, but all of the other parameters must be loaded out beforehand.

Provided there are no delayed notes, the musician need not be concerned about which voices are selected. He can simply listen to his music being performed and evaluate it. If there are aspects of the performance he does not like, the appropriate notes of the timbre table can be re-edited and the performance re-heard until it is perfect.

3.4 MISCELLANEOUS ROUTINES

3.4.1 File Manipulation (#F, #G)

One of the major advantages of a computer based system is its ability to store all of the data required by the musician. Thus for example, an unfinished task can be left and picked up again at some later date without the user having to repeat all of the previous work.

There is obvious value in being able to store all of the data associated with a performance so that it can be re-heard or further modified at any later time. The filing (#F) routine stores the score, catalogue and timbre table away on disk. This can then be "dumped" onto magnetic tape for permanent storage. Each file must be supplied with a name so that it is distinguishable, and so that it can be called back into memory when required. Files can be retrieved by the #G command ("file getting"). All data is restored to its original position in memory so that performances can be re-heard or the musician can continue where he finished last time. A message can be stored with each file and this is written out on the Decwriter whenever the file is recalled.

3.4.2 Command Listing (#X)

The routine that does the least is probably the most important to remember, especially for inexperienced users. This writes a list of all of the available commands on the screen, along with a brief definition of each. Thus, if the user forgets what command is required to do a particular job, he can look it up.

3.5 CONCLUSION

These commands do allow the musician to perform a wide variety of operations and to achieve any of the possible combinations of timbre parameters. Thus he can produce and use any of the available sounds. However, some improvements might still be possible to make them easier to use, especially for less technically minded users. Possible future developments will be discussed in the next chapter.

CHAPTER FOUR

CONCLUSIONS AND SUGGESTIONS

4.1 CONCLUSIONS

The two main objectives behind the development of the synthesizer system have been to produce a system that has all of the technical capabilities required to create music and control musical performances, and to make the system easy to use. If a system is not powerful enough to perform the tasks the musician requires or if he feels he has to waste time learning new techniques or procedures, then he will be discouraged from using it and the project would be a failure. However both of these objectives have been achieved in this system. Both the hardware and the software meet the specifications laid down for them and the system is convenient to use. This means that, for the first time, the entire system is unified to the extent that the synthesizer, the computer and the musician can communicate easily and in a meaningful way.

The project is now, therefore, at the point where the next objective must be a serious and rigorous evaluation. Part of the assessment process is being carried out at present by Susan Frykberg who is preparing short performances to explore and demonstrate the various aspects of the sounds that can be produced. Some interesting results, illustrating the effects of variations within the timbre parameters and particular combinations of parameters, have been produced.

It is true that there are still aspects of the musical sound (such as vibrato and continuous pitch) that are not under the control of the musician and the system should be extended to incorporate these. The software too is still a little clumsy to use in that it is not yet the ideal interface between the musician and the computer. Suggestions for future developments are given in the next section.

It has been found that there is an appreciable learning process involved in using this system, in the same way that it takes time and practice to learn to play an instrument well. However, every effort should be made to make this learning as easy as possible. More real-time interaction would reduce the time involved in learning about the particular characteristics of this system and the effects of the various timbre parameters. Interaction helps the musician to develop an intuitive feel for the system since he can experiment with aspects of his music and hear the results of the changes he makes.

The only way the user can interact in real-time at present is by using the harmonic varying routine (Section 3.1.6). This has been found to be one of the most exciting programmes to use, which helps to illustrate the value of interaction, but the musician should not be restricted to being able to vary only this one aspect of his music. Ideas for extending this facility are given in the next section.

4.2 SUGGESTIONS FOR FUTURE WORK

Although the system meets all of the specifications stipulated, this does not imply that it is either perfect or complete. In fact, there is plenty of room for both extension and refinement. It often seems that every improvement that is made suggests two others that should be made and one can always think of ideas that one would like to try.

Because contributions have been made by a number of different students the synthesizer described in this report has grown in a piecemeal manner. It is the opinion of the author that it is now time to stand back and re-evaluate the system in order to formulate a unified policy for future work. It was suggested to the author, by

Barry Vercoe[Vercoe,1978], that research should be carried out into the psychoacoustical aspects of music-making so that it is known what parameters of a musical sound are important. In this way it could be determined what factors make music "interesting", what factors the musician will want to control, what aspects interrelate with each other, and so on. It would seem valuable that this information be known before a system design philosophy is developed.

The current synthesizer would be useful in generating particular sounds that could be evaluated in the course of this research and could be extended to incorporate new techniques developed. The experience gained from developing the present system would also be valuable and many of the techniques used (such as the shift registers) could be duplicated for other aspects of the music.

Electronic music is often criticized as lacking character compared with the sounds of more conventional instruments and this is usually attributed to the perfect regularity with which electronic music is produced. [Von Foerster, 1969, p.133.] This apparent lack of character may just be due to the fact that we have come to expect particular types of sounds because these are the only ones we have heard [Von Foerster, 1969, p.122] but it is worthwhile trying to eliminate from any system any sources of irritation. Various methods have been suggested to overcome the regularity of most electronic music and these should be tested to see if they produce the required effects. Most of these techniques endeavour to introduce timbre changes within a note (i.e. "time varying timbres") and they often depend on the introduction of a degree of randomness. For example, the tremolo frequency can vary randomly within certain limits or slight random variations can be applied to the pitch of a note. Another way of introducing timbre variations is for the waveshape to change slightly between each period. This is brought about by applying an envelope to each of the harmonics of a sound rather than to the complete waveshape.

With the current system it is possible to vary all of the timbre parameters, except the waveshape, while a note is playing. It is not yet known how much effect this will have but it is one of the aspects that must be evaluated in the light of the psychoacoustical requirements of the musician and listener. The current software cannot be used to vary the timbre parameters during a note but there is no reason why this facility should not be added as no changes to the hardware would be required. To implement this the computer could be used to output the necessary parameter values at the appropriate times. However, this would require a large amount of storage as the computer would have to know the value that was to be loaded out and the time at which this was done, as well as which parameter it was to apply to and which voice was to be affected. The computer would also be occupied for a lot of its time supervising the loading out of the parameters, especially if there were to be several changes per second for each of the parameters on all of the voices. One way to overcome the timing problems would be to use shift registers to contain the various values for each parameter of a particular note. This would, of course, involve hardware modifications. To be able to specify the parameter profile for a complete note the shift register would have to be clocked at such a rate that it recirculated exactly once each note. This would be possible but would require considerable calculation and accurate clocking hardware, as the length of the particular note depends on the specified duration and the rate of decay.

The reason that the waveshape (and hence the spectrum of the sound) cannot be varied in real-time is that the shift register must be reloaded every time a change is made and this cannot be done while it is being recirculated (i.e. playing a note). If it is desired to be able to vary the waveshape (as well as the other timbre parameters) to produce an aesthetically appealing sound, this could be done using two shift registers in parallel so that one could be loaded while the other was being recirculated and vice versa. A counter would also be required to ensure that one shift register stopped recirculating, and the other one started, only at the completion of an entire period. A multiplexer and decoder would also be needed in order to select the correct shift register each time.

To increase the number of timbre parameters under the control of the musician and to introduce further means of real-time variation of the timbre, both the continuous pitch option and vibrato should be included in the system. The value of being able to produce notes of any pitch was described in Section 1.3 and the continuous pitch option, for which there is already provision in both the hardware and the software, should be developed. Vibrato is more difficult to implement in the context of the present system but possible methods for this are given by Vaughan [Vaughan, 1977] and more thought should be given to this.

As mentioned in Section 4.1 more real-time interaction should be included to reduce the time necessary to learn to use the system effectively. It is the opinion of the author that top priority should be given to developing interaction in all aspects of the sound. Special attention should be given to interaction during the playing of performances (sequences of notes). This would mean that the one basic performance could be reproduced in a vast number of variations reflecting the nuances of various performers.

Interaction could be facilitated by a variety of devices. One could use the instrument described in Section 2.2.6 or, perhaps, particular processes could be initiated from the organ keyboard; a bank of "stops" similar to those of an ordinary organ could be developed or a bank of potentiometers could be used. Here again it is necessary to decide exactly what is to be varied and how this is best achieved before any hardware is developed. There is no theoretical difficulty in implementing real-time interaction and most of the software that would be required already exists.

One of the major problems facing anybody working to develop this system is the amount of computer storage available. Already the synthesizer system is divided into six "phases". This means that the whole of the memory must be overlaid six times if every sub-routine is to be used. This is satisfactory except that it takes some finite time to perform a phase overlay and any applications which required more than one phase would be slowed down considerably as the phases were swapped back-and-forth between the disk and the memory. Minor savings of memory could be achieved by reorganizing some of the data areas and writing all of the programmes in Assembler instead of the Fortran-Assembler mixture that is used. The samples of the catalogued shapes could, for example, be packed two samples per computer word instead of one sample per word as at present. While this should be done, the memory saving will not be significant. To convert all of the programmes to Assembler would be time consuming and tedious. The obvious solution is more memory for the computer but if this is not possible future workers may have to seriously consider using the disk as auxiliary storage for data areas such as the catalogue and timbre table although the swapping in and out of areas of core will, once again, slow down operation.

One very important consideration that should be taken into account is the possibility of incorporating a micro-processor into the synthesizer hardware. This could help relieve the pressure on the computer memory and improve the real-time aspects of the system. The ways in which a micro-processor could be used are numerous but the system could, for example, be designed so that it could operate independently of the main computer or in co-operation with it when interfaced. In this way a performance could be prepared and all of the relevant data calculated by the computer but the micro-processor could supervise the playing of the performance (which requires very little calculation). This would mean that the synthesizer could play performances when not interfaced to the computer and could thus be used as a recorder so that performances could be heard at any suitable location.

The development of the software is very open ended. One simple improvement would be for the computer to modify the timbre table in some predetermined (or perhaps even random) manner and then repeat a given performance using the new table. Variations on the basic theme would then be produced and may suggest particularly pleasing arrangements to the musician.

More thought should also be given to the problem of making the system easily used by a person with no computer background. A discussion of the measures that have been taken was given in Section 2.3 but there is still room for some streamlining. For example, even though an experienced user can suppress many of the messages that would otherwise be printed out, there are still occasions when he has to wait while messages he already knows are typed. Perhaps a second sense-switch option could be included so that abbreviated versions of the messages are written (as is already done in some cases) or the message is not written at all if the user is familiar with the sequence and can anticipate the next question.

Of course there are many developments that can still be made and each new user and researcher will have his own ideas. The project can only benefit from this.

4.3 SUMMARY OF SUGGESTIONS FOR FUTURE WORK

In this section the suggestions made in Section 4.2 are summarized for ease of reference along with several other suggestions that do not require any discussion.

(1) Evaluate the sounds that can be produced by the present system.

(2) Remove the loud speaker from the Computer Laboratory so that the sounds can be evaluated without a lot of background noise.

(3) Develop the continuous pitch option.

(4) Incorporate vibrato.

(5) Change the upper limit of the lower tremolo frequency range from 50 Hz to 15 Hz to give greater resolution over this range.

(6) Investigate which parameters of a sound are important to that sound (psychoacoustical research) and develop techniques for incorporating these aspects.

(7) Modify the software so that the timbre parameters can be varied during the notes of a sequence and evaluate these parameter changes as a means of producing time varying timbres.

(8) Investigate and implement a method of real-time waveshape variation.

(9) Investigate the use of random variations in the timbre parameters.

(10) Investigate the inclusion of a micro-processor as discussed in Section 4.2.

(11) Incorporate real-time interaction with all aspects of the music (using potentiometers, switches (stops), the organ keyboard or special purpose hardware).

(12) Introduce computer modification of the timbre table between performances.

(13) Improve the timbre table editing options so that the notes can be inserted or deleted.

(14) Modify the timbre table editing options so that notes can be selected (using the joy-stick) while drawn on the screen instead of the user having to type in the note number.

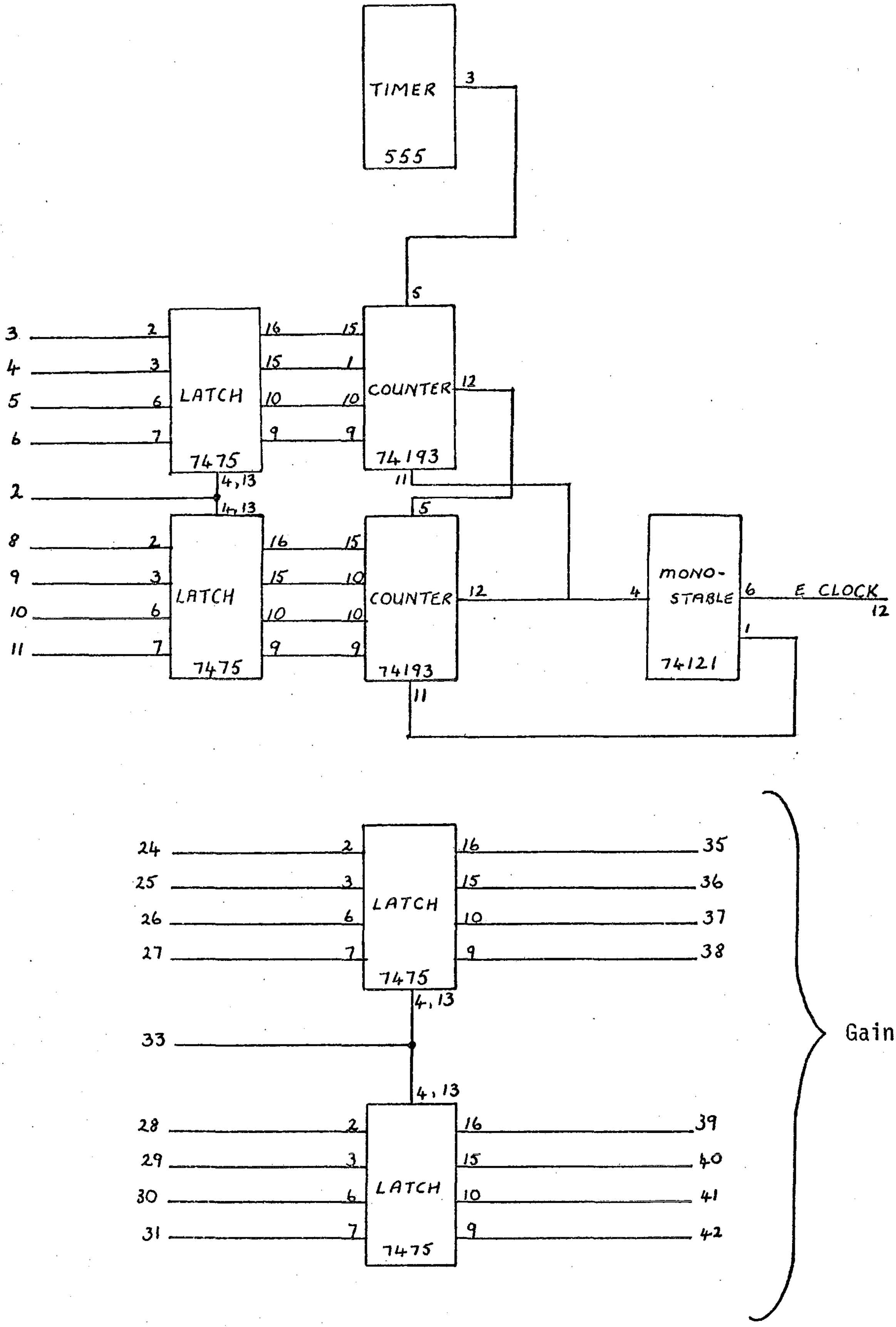
(15) Minimize the storage required for data areas and programmes (e.g. pack the shape samples two/word).

(16) Modify the sequence playing routine so that notes can be turned off before the duration has been completed so that the notes are not lengthened by the decay period.

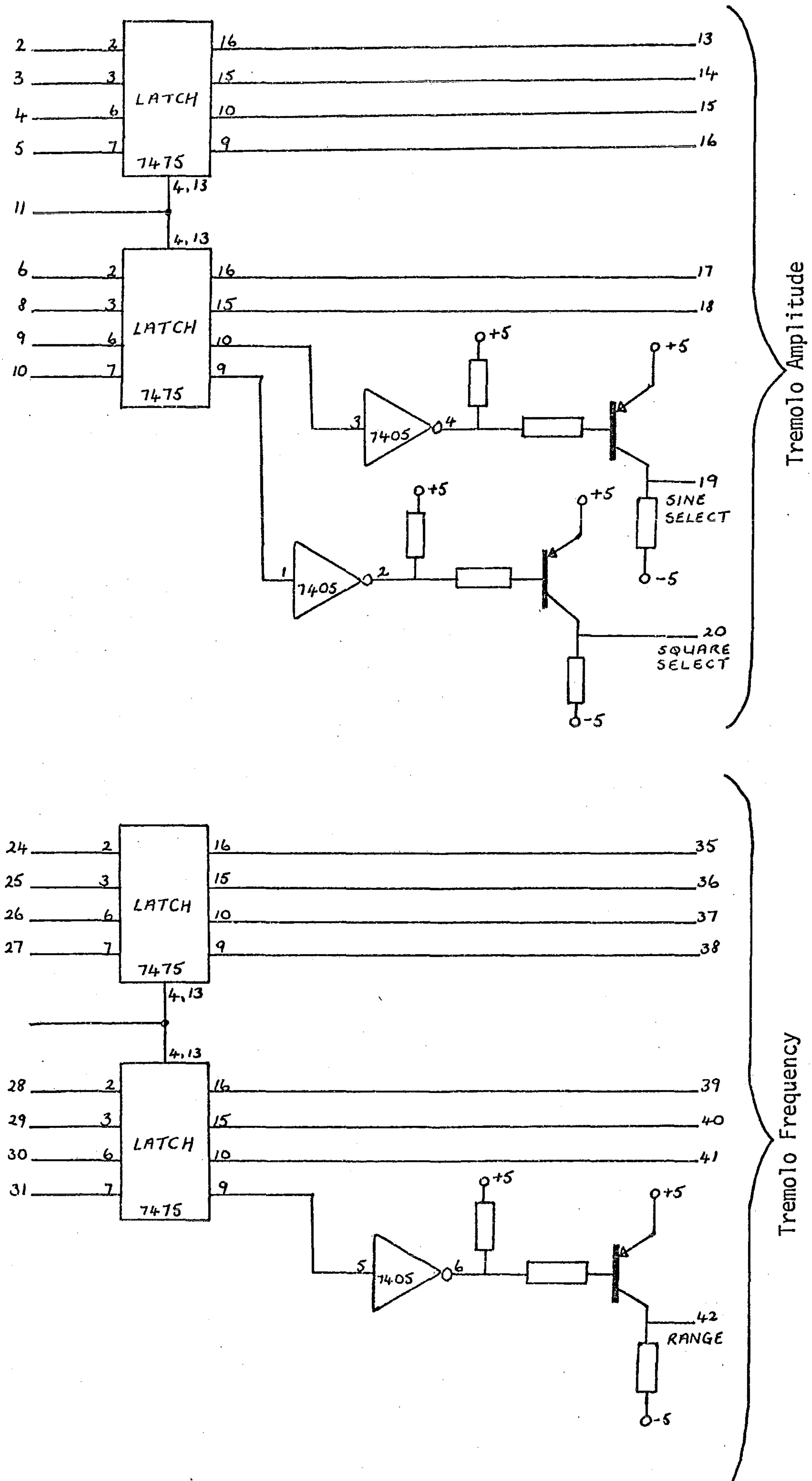
(17) Modify the software so that short sequences of notes can be combined in any order to produce longer sequences.

APPENDICES

APPENDIX A



Envelope Clock Generator and Gain Latches



Tremolo Amplitude and Frequency Latches

APPENDIX B

SUMMARY OF COMMANDS

This is a summary of the commands available to the user showing the scope of the software modules. A manual for users is held in the Electrical Engineering Department giving full details on how to use these commands and explaining the arguments and sub-commands that can be used. Some details are also given in Chapter Three.

#A	-	Add comments to screen
#C	-	Calculate shape (rectangular, triangular or sum-of-sines)
#D	-	Draw shape on display screen
#E	-	Set envelope status (percussive or non-percussive)
#F	-	File performance to disk
#G	-	Get filed performance from disk
#I	-	Low pass filter catalogued shape
#L	-	Load out waveshape or envelope samples to synthesizer
#M	-	Mirror catalogued shape
#P	-	Play a note
#R	-	Rotate catalogued shape
#S	-	Show catalogued shape
#T	-	Vary timbre parameters
#U	-	Calculate spectrum of catalogued shape
#V	-	Vary harmonics of envelope or waveshape
#X	-	List commands on screen
#Z	-	Set up timbre table and play sequence of notes

REFERENCES

Blesser, B.A., Baeder K, and Zaerski R. (1975). "A Real-time Digital Computer For Simulating Audio Systems". Journal Audio Eng.Soc., Vol.23, pp.698-707.

Brigham, E.O., Morrow R.E. (1967) "The Fast Fourier Transform". IEEE Spectrum, Vol.4, No.2., pp. 63-70.

Cabot, R.C., Mino, M.G., Dorans, D.A., Tackel, I.S., Breed, H.E. (1976) Journal Audio Eng.Soc., Vol.24, No.7. p.568.

Cage J. (1974). "The Future of Music". Numus West, No.5.

Chowning J.M. (1973). "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation". Journal Audio Eng.Soc., Vol.21, pp. 526-534.

Douglas, A. (1973) "Electronic Music Production". Pittman, London

Fitzgerald, D.I. (1977) "Timbre Control for Computer Generated Music". Third Professional Project Report. Electrical Engineering Department, University of Canterbury.

Jacobs, A. (1973). "A New Dictionary of Music". Penguin Books.

Laske, O.E. (1974) "Towards a Centre for Studies in Musical Intelligence". Numus West No.5.

Mathews, M.V. (1961) "An Acoustic Compiler for Music And Psychological Stimuli". Bell Systems Tech.Journal, Vol.40, pp.677-694.

Mathews, M.V. (1969) "The Technology of Computer Music" M.I.T. Press, Cambridge, Massachusetts.

Mathews, M.V. and Moore, F.R. (1970). "GROOVE - A Program to Compose, Store and Edit Functions of Time". Communications of A.C.M. Vol.13, pp. 715-721.

Mathews, M.V., Risset, J.L., Moore, F.R. (1974) "Computers and Future Music". Science. Vol.183, pp.263-268.

Moorer, J.A. (1977). "Signal Processing Aspects of Computer Music : A Survey". Proc.IEEE, Vol. 65, pp.1108-1137.

Pierce, S.R., Mathews, M.V., Risset, J.C. (1965) "Further Experiments in the Use of the Computer in Connection with Music". Granesaner Bläiter, No. 27/28, pp.72-97.

Smoliar, S.W. (1973) "A Data Structure for an Interactive Music System". Interface, Vol.2, pp.127-140.

Snell, J., (1977) "High Speed Multiplication" Computer Music Journal, Vol.1, No.1., pp.38-45.