MACROMUSIC FROM MICROS: A DIFFERENT ANIMAL

1704(E-8)

Laurie Spiegel
New York, NY

Presented at the 67th Convention 1980 Oct. 31/Nov. 3 New York

AES

This preprint has been reproduced from the author's advance manuscript, without editing, corrections or consideration by the Review Board. The AES takes no responsibility for the contents.

Additional preprints may be obtained by sending request and remittance to the Audio Engineering Society, 60 East 42nd Street, New York, New York 10017 USA.

AN AUDIO ENGINEERING SOCIETY PREPRINT

Macromusic from Micros: A Different Animal

Laurie Spiegel
New York, New York

::

ABSTRACT: When people use a new technique, they tend initially to be unable to bring out its strengths because they limit the ways they use it to conform to their past experience. Attempts to realize the musical potential of microcomputers are discussed in light of concepts inherited from 3 predecessors: instrumental music, analog synthesizers, and large mainframe computer music systems.

::

PART I: THREE INTRODUCTORY QUESTIONS:

It is appropriate that this discussion of microcomputers in music be part of the session on micros in audio, rather than part of the regular music session. Micros in music are likely to be used by people outside of the professional music world, at least at first, and to gain their first broad acceptance as a new kind of folk instrument. It will probably be some time before more than a handful of professional musicians feel that the microcomputer is a suitable instrument for their work.

1. Who is involved in microcomputer music?

Unlike large expensive music synthesis systems, micros are most likely to be used by home computerists who are neither expert nor professinal in either music or computer techniques. Of the two, it is more likely that the micro-musician have a well-developed technical than a strong musical background, as deciding to use already-present technical skills to make music would be more satisfying than abandoning musical techniques already mastered to try to achieve a similarly rewarding musical level with computer skills that must be learned.

The musician who wants to go over to computers is likely to have in mind some specific effect (s)he wishes to achieve and can't get any conventional way. This is probably a small group compared to the growing number of computerists looking for some way to gain more personal meaning and pleasure from skills and equipment already possessed. The largest demand for microcomputer music systems is likely to come from people who have wanted to be able to do music, lacked the extensive traditional training necessary, and hope that technology will enable them to substitute the use of their intelligence and sensativity for years of laborious acquisistion of reflexes and an early start.

People who have not done music before, either with traditional musical techniques or with larger computer music systems, who are coming to music entirely through small computers, posses an ignorance which may lead to beneficial, new, and untried approaches. But the work may also be impeded by lack of experience. The worst result of lack of experience is not loss of time through re-invention of previous techniques. It is the unquestioning implementation of previously developed techniques in situations where they may be inappropriate or even counter-productive, as a result of insufficient research and understanding.

The greatest disadvantage of education and experience is that it is habit forming, not by addiction to the acquisition of knowledge, but in the habituation or entrenchment of frames of mind, methods, attitudes which - though once productive - now prevent progress.

2. Where are they getting their ideas?

In the current development of microcomputer-based music systems there are 2 main sources of questionable preconceptions, both of which are creating problems. These are the two areas most likely to be studied by microcomputerists endeavoring music. One consists of the traditional meothds whereby music has been created in the non-electronic past. This is a body of technique and method requiring many years of study, and easily subject to oversimplification and misunderstanding, as well as ignorance of musicians' dissatisfactions with same. (A goal of computer music should obviously be, not duplication of past methods, but improvement on them.) The other precedent area of study consists of methods in common use with large computer music systems, concerning sound synthesis techniques, scientific versus artistic data types and methods, types user interaction, and other such questions.

A third available source of preconceptions, which does have valuable ideas to offer, but is unfortunately often overlooked, is analog music synthesis. Digital versus analog thinking have unfortunately overspecialized away from each other to the point where analog and digital system design are considered almost irrelevant to each other. The chief characteristics of the analog music synthesizer which may be seen as potentially beneficial to, but underutilized by, small digital music systems include:

1. modularity
2. hybridization
3. parallel processing
4. random access to (of) all signals, variables, controls
5. completely flexible signal routing
6. subtractive synthesis

3. What would be the advantages of using micros to make music, versus older methods?

Disadvantages of traditional instrumental and paper music:

1. People want to begin making music later in life than childhood, and music is one of very few techniques which are dependent on such lengthly study that it must be started in childhood or the boat will generally have been missed. Computers promise the substitution of intelligence and sensativity for coordination and reflexes.

2. Late starters don't want to have to learn, and trained composers don't want to be limited by, the traditional written notational language and its constraints. (e.g. notation's bias toward discrete rather than continuous changes - every language has inherent conceptual biases)

3. Composers want to work directly with sounds, rather than symbols for them, including sounds that could never be played by the traditional composer's tool, the piano. (to work on the work itself, as a painter does, instead of composing a set of symbol instructions for someone else to use in realizing your actual piece)

4. People want to make sounds either unmakable or indescribable by traditional means.

5. Composers want to automate the time consuming drudgery of such tasks as copying scores and parts, and such standard pattern transformations as transposition, inversion, retrograding, as well as resequencing, insertion, deletion, and other such editing operations as are cumbersome on paper.

6. People without instrumental performance skills want to be able to create their own renditions, performances, interpretations, of standard repertoire.

7. Improvisers lacking adequate technique in written notation want to be able to record their creations in a format that can be edited and altered. (unlike tape)

8. Musicians of all types want to shorten the turnaround time between conception (or impulse) and realization.

Dissatisfactions with analog systhesizers:

1. They are expensive.
2. They are large and cumbersome.
3. They are difficult to calibrate exactly.
4. They often drift and are unsteady.
5. They have no memory.

a. It is not possible to return to exactly the same calibration.

b. It is not possible to store and replay a series of user interactions.

c. It is not possible to store and replay either a specific sound or an entire piece.

d. All editing and recording must be done with tape, which is noisey and also prohibits selective alteration of parameters or individual component sounds.

6. The patterns of control available are both limited and simplistic (cyclic repetition, randomness, sampling, and keyboard control).

7. Unlike software routines, hardware modules are fixed in function and finite in number.

Disadvantages of large computer music systems:

1. Too expensive (for private ownership).
2. Too large (not portable).
3. Generally not owned by the user, with consequently limited access.
4. Often require specialized knowledge of both traditional musical techniques and computer techniques, rather than reducing the amount of knowledge required to create music.

Other reasons for microcomputer music:

1. Microcomputer freaks want to be able to do music on their computers regardless of whether it would be easier (or better music) to do the same music by another method. (afterall, they love their computers the way musicians love their instruments)

2. For various reasons, people who have micros are always looking for things to do with them (in order to feel better about having gotten them in the first place?).

3. Music itself has suffered from the exclusion of many highly musical individuals by virtue of their lacking conventional means of creating it. The musical community has generally been smaller than those of writers, painters and other such occupations due to the elaborate technique which must be learned, the early age at which it generally must be begun, the non-musicality of criteria for selection of individuals able to pursue music (physical coordination and mastery of a symbolic written language), and the scarcity of access to tools of production (an orchestra of 100 trained individuals in a large space, each playing a handcrafted precision instrument and reading hand-copied notation, preferably mic'ed for 24 track recording, is neither common not cheap). Micros will hopefully reduce the exclusivity attached to older musical techniques.

4. Instrumental, analog, and large-computer musics are all relatively expensive to produce, and all require access to scarce means of realization (performers, studios, equipment).

5. Computers offer a unique and as yet unfulfilled potential for synchronization of music with non-musical media, for which it is often created (film, TV, etc.)

6. Because they are there.

PART II: Casting aspersions on questionable assumptions which underlie microcomputer music system designs?
!!

Micros differ from the large mainframe computers on which established computer music techniques have been evolved. They are smaller and slower. They are easier to modify as hardware and easy to interface with other equipment. Carry-over tendencies / preferences / assumptions from traditional paper music and large mainframe computer music systems which are being applied to micros but deserve to have a few aspersions cast on them include:

Assumption : Musicians can't program, and shouldn't have to.
Aspersion : High level languages are probably easier to learn than the musical notational language. This prejudice results in a tendency to create large inflexibleintegrated music software systems rather than music languages consisting of libraries of single function modules (macros) which can be interrelated via a higher level language. (shameful in this era of structured programming)

Assumption : Conventional staff notation on microcomputers is wonderful.
Aspersion : What functions has this difficult to learn, difficult to use, uneconomical, traditional musical technique served in the past? Are they still necessary functions given computer technology? Is there any better way to serve these functions on a micro than there was on paper? Visible instructions to players are unnecessary, as the computer plays the music. The user can tell if the music is right by ear, and doesn't have to read it. Notation permits dealing with more material than can be held in detail in the human memory, but a tiny graphic window showing a paltry few notes is insufficient. Notation is used for dealing with relationships more complex than can be simultaneously conceived or described, but displays only events, not relatonships per se. Notation is not needed for communication of musical material in the absence of sound, as the equipment which displays the computer notation can also play the music it represents. Tape recording has been invented, too. Most people don't read notation anyway. What is your reason for implementing it?

Assumption : Musicians want to know exactly what something is going to sound like before they hear it, and to be able to describe exactly, in advance, what they are going to make audible.

Aspersion : Scientists may prefer exact quantitative description, and work best with total preconception and definition of sounds and sound complexes, but music generally involves a process of real time empirical discovery, interactive alteration of a more qualitative than quantitive nature ("Play that passage with a gentler sound" rather than "attenuate partials number -, -, and -, by -- DB"). Often composition consists in large part of recording, ex post facto, what has been "discovered" during a session at the keyboard (or console). In the past, due to inadequacies of traditional musical technology, it has been necessary to specify music in advance of hearing it if one wanted to make music beyond what one could play oneself. This has made composition a difficult art, and need no longer be the case.

Unlike scientific research, music commonly uses a "black box" technique. The instrumentalist does not need to know what the specific content of a sound is or to understand the exact mechanism of its production. Through practise in a continuous feedback situation, a musician learns how a sound will change when interaction with the means of production changes. Unlike the scientific researcher's need for exact quantized description of phenomena, music makers need a clear and delicate correlation between an input and an audible response. This is a relationship of analogy, effectively implemented in analog synthesizers, as derived from conventional musical instruments.

Assumption : Additive sinewave synthesis is best.

Aspersion : Adding sinusoidal partials in exact proportion may be an educated scientist's ideal and gives the greatest control over timbre. However, this technique requires extensive knowledge, and greatly magnifies the number of variables that must be dealt with. Consequently, it is likely to increase the time delay between musical impulse and sound. Large, fast processors are required, especially if any time, space, and attention is to be devoted to good user-interaction, a good editor, and other such system characteristics. Use of this technique has exhibited a tendency toward music of preconception rather than spontaneity, as it pre-dates real-time synthesis.

FM synthesis (Chowning) generates harmonics on a fundamental more economically, though sacrificing some timbral range. FM synthesis using complex waveforms (as opposed to sinewaves) has not yet been adequately explored. Using sinewaves, this technique is still likely to require too much processor speed for micros, unless it is implemented in modular plug-in hardware. One of its chief advantages to the music maker is the reduction of a large number of relatively weak timbral variables to a small number of relatively powerful ones. Music has too many varaible already, and too much detail, which distracts from overall musical flow and shape.

Toggling bits in the machine to make square waves, considered far and wide to be intolerably limited in its timbral potential, but which is the natural, and most efficient way to produce audio signals on computers, has yet to be adequately explored, either through the sophisticated machinations of Walsh functions, or through the analog-synthesizer-derived method of complex waveform FM and subtractive filtration, which is tried and true.

The effectiveness of subtractive synthesis has been hard for mainframe mentalities to perceive because good digital thought holds as an ideal the generation and processing of only that information which one ultimately wants. Analog experience has found that putting out a lot more sound information than you want, and then filtering it down (adjusted by ear) is musically effective, easy to learn and control, and economical. I have personally found microcomputer squarewave-squarewave FM with hybridized (analog) filtration to be musically useful, but apparently, otherwise unexplored.

Assumption : It is somehow better and purer to do everything digitally, within the central computer itself, no matter how small it is, avoiding hybridization, particularly avoiding analog modules.
Aspersion : This kind of thinking advocates the implementation of software oscillation on systems with mere 1 megahertz clocks, as opposed to interfacing external oscillators which need only be updated by the computer. It means tying up most of RAM and of processor time in dedicated sound generation, and severely restricting what's available for other uses (control functions, user IO, etc.)

Conceiving of analog processes and hybrid systems as mere compromise stopegaps to be avoided if at all possible, misses two points. First, hybridization results from the selection of the most efficient method and components for the specific task. This involves breaking down or modularizing the complete process to be implemented, and then implementing each stage the best way. Second, hybridization, typically of such tasks as oscillation, filtration, or attenuation, is actually a form of parallel processing. Parallelization is an extremely important principle for optimizing many kinds of systems. Music, a highly parallelistic phenomenon, is likely to benefit from relatively high modularity and hybridization.

Assumption : Music is made up of notes.
Aspersion : Here we find a common unnecessary assumption implied by both large-computer music systems and (upon superficial study) traditional musical technique. Traditional musical scores are composed by building up a total work out of many specific event descriptions. That is a function of its primative technology, and is conceptually similar to building up a single sound by exact description of each sinusoidal component. As composers start with themes, motives, chord progressions instead of notes, analog and instrumental techniques of timbral construction start with the creation of textures of sonorous material which are then continuously and qualitatively (not quantitatively), refined. In the compositional example, what is produced is not so much a group of notes as a musical pattern module designed for ease of extension. In the second example, the resultant timbre possesses a desired subjective quality.

The idea to which this leads is that music does not consist of individual sound events, any more than timbre consists of individual component sine waves. Music, as subjectively perceived, has never been objectively differentiated from non-music. Music and timbre consist, not of events, but of patterns, systems, complexes, configurations, which vary qualitatively according to principles as yet unformulated. The definitions of these principles are to be found by study of the interaction of our perceptual and cognitive systems with accousticly encoded patterns of information.

A fitter study of timbre than analysis of conventional instrument sounds, with a goal of resynthesizing them, would be the study of how the orchestral composer's use of timbral variation correlates with the subjective effect that is to be created. Probably in the long run, it will be more economical to program such basic principles than to store and assemble data necessary to recreate the tools of an earlier practise. (e.g. to heighten contrast, smooth the existing transcients in some of the current waveforms, and exaggerate them in others -- substituting a concise algorhythm for an extensive database representing a full orchestral palette)

Music consists of PATTERNS of sound (rhythms, melodies, themes, chords, sequences, ornaments, sonatas, rondos, canons, fugues, etc.) and of transformations of theses patterns (repetition, variation, transposition, inversion, substitution, superimposition, interpolation, augmentation, diminution, etc.). To the extent that the system user is required to be concerned with specific parametric values of each sub-event or accoustic component, it becomes that much more difficult to focus on the music (larger-scale architecture). One reason that there have been few great composers is that few minds are able to transcend all that musical detail to maintain adequate focus on large-scale patterning.

Sound is not music. Even organized sound (rivet guns, lawnmovers) is not music. Particular patterns of organization are music. The patterns in question are those from which the human perceptual system is able to derive meaning. The focus of computer music systems, unprecedented instruments for the manipulation of patterns of information, should be optimization of the congruence between patterns in the sound they generate and patterns from which our cognitive and perceptual systems are able to interpret meaning. It is both unmusical and a waste of the computer's potential to deal with music as a non-patterned collection of individual events, separately described one by one.

The achievement of already-established musical phenomena and methods (staff notation, instrumental timbre resynthesis, etc.) may be impressive technical accomplishments, but they hold no particular value for music itself. It is only the ways in which electronic generation of music differs from established musical methods that will have real or lasting value. The value of such differences will not be so much in the creation of radically new and different musics, but of better ways of creating music as powerful as the best music that has been done in the past. This can only be done with better understanding of how such music has been made, how it effects us, and of what insufficient techniques and premises have prevented its being so well made more often.