New Computer Music Facilities at the University of Illinois at Urbana-Champaign

James Beauchamp and Soot Aurenus
School of Music and Department of Electrical and Computer Engineering, UIUC, Urbana, IL 61801 USA

ABSTRACT

Two new computer music situations have been established at UIUC. One consists of the Computer Service Office's VAX 11/780 system to which a stereo 16-bit DAC has been attached. Users can either compute samples on the VAX or bring digital tapes from other campus computers such as the Cyber 175 or IBM 3744 (where much analysis and synthesis software has been developed over the past 15 years). The VAX runs Unix Version 7 (4.2 BSD) and has limited accessibility, since it is shared with many other campus users. The second computer music system consists of a Logical MicroComputer (LMC) 8500-based microcomputer, which is attached to a private sound conversion/storage system (SCSS) and is housed by the School of Music's Computer Music Project (James Beauchamp, Director). The LMC/SCSS system is dedicated to music composition and research and is available to qualified persons on a continuous basis.

The School of Music computer system was purchased for a total budget of $28,000, required no special space preparation, and generates less than 50 dbm noise. The LMC computer is contained in a 15 shuttled cabinet and includes one 5016 cpu, 2388 floating point coprocessor, and the 3288 memory management chip; a 5 Myra floppy disk; a 42 Myra hard disk; 2 Myra RAM; and 6 plp serial output.

It runs Unix Version 7 (4.2 BSD), and for computing-intensive programs it runs 2-3 times slower than the VAX 11/780 (without 4ps). Connected to the LMC are a Siler M5007 graphics terminal (1024x768 pixels), a Siemens PT-89 low noise printer, a 1200 baud modem, and the digital audio processor subsystem.

Software is currently being developed for sound file management, some synthesis and analysis, and graphics. For example, the LMC is used to create sound synthesis programs utilizing user-written or existing software. The LMC/SCSS is operational for several months. Much of the software is first tested on the VAX and then mailed to the LMC. Most users interact with the LMC via the dialup port unless they need its special graphics or sound capabilities. Thus, program development is limited to those who find it convenient to work directly on the LMC. The sound conversion/storage system (supplied by the UIUC CSEbL Music Project) has the following characteristics: a 768k processor, 25 Myra RAM, and 85 Myra hard disk and is connected to the LMC via high-speed interface. Since the SCSS is devoted to handling all disk-audio transfers, it was not necessary to program the LMC to handle these real time operations. The LMC computer and delivers (or records) samples to (from) the SCSS and initiates PLAY and RECORD without fear of inhibiting its normal multitask response.

1. A Brief History of Computer Music Facilities at UIUC

A complete detailed history of our computer music facilities would look like a summary of the history of the British Empire.* In the past we have relied on the services of other departments' facilities on campus for both computation and digital-to-analog conversion. While in general our campus' main-frame computer services have been reliable and steadfast, since 1965 we have been forced to change DAC facilities or an average of once every three years, and in practically every case have had to contribute plenty of our own time and money to make other groups' systems work right for music, always with the promise of continued access. These situations almost always necessitated the use of digital tape for transfer of sample data from campus mainframes to the "current conversion facility" (generally a relatively small computer with DAC attached). Of course, time was wasted in staring each new DAC facility, and whenever DAC became unavailable at UIUC, we were forced to send digital tapes to other universities for conversion to analog form. It was obvious that we should have our own DAC facility except that until recently the recitalist's initial and continuing cost of an appropriate system made this impractical.

While computer music at UIUC can be traced back to 1958 with the experiments of Leibergen Miller in computer-assisted composition, the first sounds did not emanate from a computer at UIUC until 1965 when David Freedman installed both an ADC and a DAC on the Illiac II computer. Since that time we have used six different DAC systems on campus. Most recently (1983-85) Herbert Brun and Jerry Kravis carried out a project to install a DAC on the UIUC Computer Services Office's VAX 11/780 computer. It became operational in May, 1983, but it has limited access. Unfortunately, we have just been informed that CDC will remove their VAX 11/780 next year.
Finally, in 1983, Beaubourg, Bruno, Nelly, and Trevor Tipping proposed to the National Endowment for the Arts and the GIDC Research Board to configure a computer music system inside the GIDC School of Music. This was installed in November, 1983 except for the sound conversion/output system, which will be installed during July, 1985. This self- contained computer music system is the basis of the Computer Music Project, which we hope, funding permitting, will spur our problem with dependencies on other groups on campus.

Despite our problems with DAC, we have always enjoyed access to excellent computation services. Computers have been IBM 7944 and IBM 7080/7 (in the 1976-78), and the IBM 4311 and 360/65 (in the 1979-80). Sound synthesis has been accomplished by packages such as Music 5, Music 48, and Music 390, which is very efficiently on minisystems. Sound synthesis has been accomplished by Beaubourg's TONEA package.

Our computer music facility history would not be complete without mentioning that during 1978-79 Beaubourg directed a hybrid computer music project utilizing a TRN-12, a custom synthesizer and interface, and the GIDC PLATO system for control. This project gave us some valuable experience with interactive composition and graphics using a special musical language called FLAMENCO (Me- ray et al, 1979). In 1981 the FLAMENCO system was relayed by a Synthesizer system, under the direction of Scott Wyant.

In terms of research and creative work, our group has done well. John Melby has realized an average of two finished pieces (usually for one or more instruments plus computer) per year and has received awards for his work. Herbert Stone has recorded many of his com- poses, which were performed as a recital at the University of Illinois in 1985. Jeanne Beaubourg has pub- lished numerous papers in sound analysis and syn- thesis and is a member of the Audio Engineering Society. Trevor Tipping's compositions using MIDI are frequently performed in the USA and abroad. Sava- ior Missieres is a well known composer who has recently spent several months in Israel working on computer music and to perform his custom digital synthesizer at the University of California at San Francisco. Among many successful students, Brad Alexander (1976) directed the studio project at U. of Md. at Tallahassee, Florida, and Sam Lomaha (U. of Michigan, 1979) is director of the audio curriculum at U. of Miami and is a prolific composer for such periodicals as DM Magazine and Digital Audio.

I. GENERAL ROLE

Our objective is to establish a center to satisfy the diverse needs of a large group of composers and researchers. Indeed, the needs of each of the Com- puter Music Project original groupswas quite different. For those who wish to utilize the computer and expand their skills on such periodicals as DM Magazine and Digital Audio.

2. FACILITY REQUIREMENTS

The first requirement is that the system be dedicated to music synthesis/composition and, as a corol- lary, be affordable. Almost as important, the sys- tem should be "general purpose," i.e., capable of realizing any sound synthesis or analysis technique in software, and hardware should be provided for playing from or recording into made simple files (of up to 20 minutes stereo duration) which can be connected to the system/synthesis analyzer software. A system satisfying these requirements with suitable performance characteristics could not be purchased for less than $30,000 until very recently.

Other important requirements are as follows: The computer should satisfy the needs of a diverse group of composers and engineers; for our use in the future, sound processing, graphics, sound input and output, and text-complex languages (for- tran and C) should be available. The system should be multi-tasking and multi-user, i.e., when high data rate DAC or ADC activity in taking place, specifically, the system should exhibit "good response" when up to four people are using the sys- tem simultaneously. The system's software should be compatible with an accessible reference machine (in our case, a Macintosh) and a number of the computer music facilities in the U.S. and abroad. The computer's operating system and language should contain a large variety of utilities, allow a great deal of flexibility, and pro- vide an interface for third party support. Learning to use the system should not present a problem. The computer's storage should be fast, have a large address space, support (the sharing of data from the right kind of L/C (sound, graph- ics, and text) does just that. The system should also provide the tools necessary for a variety of approaches to problem solving, stimulate an open, intellectual, non-competitive atmosphere. Finally, it should strongly encourage the production of music for public performance and the publication of research results.

In addition to providing tools, we will be respon- sible for education in the use of project facilities. The first course designed to teach how to use the UMC system to accomplish sound synthesis will be offered in fall, 1985, with a follow up course planned for spring, 1986. The course is for composition majors and interested computer engineering majors. (The latter are encouraged to write new software for the UMC.) Also, we are contemplating an intensive workshop for summer, 1986.

ICMC '85 Proceedings 408
Unit to the operating system since it allows multi-tasking and is the overwhelming favorite operating system for computer music groups with which we would expect to share software. Also, during the past few years comparatively low-cost "Unix boxes" have become available which satisfy most of the hardware criteria mentioned. One problem, however, was that most of the systems in our price range did not support hardware floating point operations. The LMC Magn tro computer, based on the National 32016 cpu and 32081 floating-point chips, was an exception. LMC's box was compact and low noise and contained 15 multi-bus slots, only four of which were initially filled. Their operating system was a 16-user version of Unix (Dennis), and 8 serial ports were standard. While several competing computers were clearly faster in general, the LMC won out according to the all-important floating point benchmarks (by a factor of 10 to 20 compared to 68000 systems without floating point units).

4. COMPUTER MUSIC PROJECT HARDWARE: EXISTING AND PROPOSED

Because of our restricted budget ($28,000), we decided to limit our first version of the system to one which would handle only a very few users but which would demonstrate a number of different options for user-friendly program development, graphing, sound synthesis and analysis, and sound output and input. However, the system was to be an expandable one. Our initial configuration, based on the LMC Magn tro is shown in Figure 1. All components were installed on November 1, 1984 save the Hasselblad 942 terminal (on loan) and the sound conversion/storage system (installed during July, 1985).

Figure 1. Computer Music Project System

4.1 The LMC Magn tro Computer

The LMC is in general only a moderately fast Unix time-sharing system, but for floating-point execution it is comparatively fast. For one-compilation language its compilation timings were about the same as the PDP 11/34. For 8080 and 68000 systems, the other (68000, no fp) comparison machines by factors ranging from 10 to 20. Within the constraints of the LMC's architecture there are several avenues for future speed improvements: a) optimize the operating system; b) increase the system clock (now 5 MHz); c) upgrade hard disk and more intelligent disk controller; d) upgrade to intelligent serial interface.

As a representative of a new class of "superpower computers", the LMC features VAX-like performance (although scaled-down) in the sense that it is multi-user and that its operating system (Unix Version 3, 4.1 BSD) is bundled with a rich variety of utilities. It is also very reliable (practically no downtime in the field). In addition, the site preparation (10 - 30 C, 10 - 80% humidity, consumes about 1000 watts # 120 volts), and is in the 107 cubic meter volume and quiet (less than 60 db noise). And at a cost of $19,000 (with educational discount), it was affordable.

The LMC's box has four internal spaces for mm storage devices, with a choice of hard disk (48 Megabytes each), floppy disk (18,000 tracks each) or cartridge tape. We were able to afford only 1 Quantum hard disk and 1 Epson cartridge for these reasons: there is no cartridge drive available for backup of the hard disk since it takes six hours to backup just the operating system (Even high density floppies)

The LMC is very easy to expand. First, there are eight RS-232 serial lines up to 19,200 baud, allowing a large variety of peripherals. Second, it uses the standard Multibus (IEEE-196 bus, which is quite fast, and for which a large number of firm manufacture memory and peripheral controller boards. If one wants to add a tape unit, for example, he has a large field of choices for controllers and actual physical systems. The pitfall is that one also needs a software driver that is compatible with the LMC operating system and hardware. For our initial system only 4 out of 15 available slots were used, leaving a lot of room to expand. Generally all one has to do is plug in the unit using a simple cable and make small changes to a couple of flags in the operating system.

Finally, the LMC is an example of a "generic" machine. While use machine itself will eventually become obsolete, we expect that the Unix operating system and the C and Fortran languages will be around for quite a while. If 1985 were the year to choose a computer, we might then choose a Unix machine (Masscomp and Microvax currently look better)
interesting), but we had to make our choice in 1984. The software we have been developing is highly portable (most of it also runs on the VAX 11/780), and I feel some point it behoves us to switch to a different processor, it should be easy to adapt it to the new system. Moreover most of the hardware should be easily transportable to a different processor, particularly if that processor uses Multibus I.

4.2 Graphics Terminal, Printer, and Nodes.

Several of the Computer Music Project grants had established an interest in graphics. While there were several supermuto computers on the market (e.g., Sun, CADAM, Apollo), which featured excellent built-in high resolution bit-mapped graphics, none of them were within our price range. We needed relatively inexpensive graphics with hard copy. The solution was to purchase one of several available Tektronix 4010/4013 terminals which also provided options for graphics screen dump to a bit matrix printer. We chose the Selenor Himes 100 terminal because of its good monochrome graphics resolution (256 x 188 pixels) and its ability to screen dump to an MG-80 graphics compatible printer. We also chose it because it offered a standard VME terminal emulation with cursor control (VT52), and because of its reasonable cost ($1850). The Tektronix model is useful because of a great deal of software is available which uses the Tektronix protocol. It is also used effectively with remote as well as local graphics terminals.

One of our highest priorities has been to keep a lid on the computer equipment noise problem. This is especially important in an environment where music listening will take place. For this reason we have been investigating the new low-noise technology, the Siemens ST-95. The absence of print head impact implies extremely low noise and comparatively long life and mean time between failures. Moreover, since the PT-95 has a serial interface option and emulates the MG-80 graphics standard, it works in both text and graphics screen dump mode with the Selenor terminal. It is also a fairly fast printer, being rated at 150 cpm. The cost (at $3600) considering that many cheaper printers are available today with more varied performance possibilities.

A mass serial switch (Cliktronix) has been installed so that the printer can be either attached to the graphics terminal for screen dump or directly to the LMC for printing of entire files. There are three screen dump modes: test screen dump, graphics screen dump, and text pass-through. In the latter case, the terminal printer can be used as a hard-copy terminal. When connected to the LMC, the user can print files using the terminal as a part of the system or by using the command 'printer' which apport the printer output and prevents collision between units.

Perhaps the most effective investment we made was to purchase a 300/1200 baud serial/isolated/receive module, the DS Robotics Password Nodes ($300), which is connected to one of the LMC's serial ports. We use this for dial-in from home terminals, dial-out to other computers, and for mailing software to and from the CCR VAX 11/780. About 80% of the LMC's use has via modes dialup from home terminals or microcomputers. With remote terminals users can answer mail, update the system, and try out programs, including those which use Tektronix 4013-styled graphics. For mail to be passed to and from the LMC, the VAX 11/780 calls the LMC periodically.

4.3 Sound Conversion/Storage System

Hypervoice is a real-time real-time transfer system in a nontrivial job. A direct memory access interface may be necessary, and frequently notifications to the Unix kernel are required. (This assumes that the Unix computer's own hard disks are going to be used to hold the sound samples for conversion.) Even so, this's response to interrupts is typically very sluggish, on the order of tenths of a second. One solution would be to use a large amount of buffering in a parallel interface, a technique incorporated in Micro Technology Unlimited's Digibound I a 16-bit stereo input box which sells for approximately $3500. However, MUV did not deal with the problems of Multibus interfacing (it operates from a generic parallel interface) and special disk-to-LMC software for the host computer.

Another solution is to have a completely separate computer with its own memory and hard disk, dedicated to the DAC/ADC conversion process. Data would then be transferred from the LMC to the satellite system in non-real time, and there would be no problems of critical timing. Such an approach was proposed to our group by Lippold Haken and Kurt Hebel of the University of Illinois CCL Music Group in summer, 1984, and we decided to allocate $9000 for them to build this system.

Some of the attributes of the Sound Conversion/Storage System are:

1) 16-bit stereo in/out at 32k sample rates up to 100 kHz total, audio quality.

2) Mixed sample files can be recorded into or played back from the host system, deleted, and manipulated in various ways.

3) Sample files contain headers giving information about the files.

4) Feedback and record operations will not place a load on the host system bus.

More details on the Sound Conversion/Storage System are given in the UCSC ICML Proceedings paper by Kurt Hebel (1985).

4.4 Proposed Additional Hardware

Of course, the system just described is in just the beginning. Much more equipment is needed to make a completely versatile system. Some additions are essential and are highly desirable for future development. Recently we were allocated $16,000 from the UCSC Research Board, which we will use to purchase two essential items: audio equipment and a digital tape unit. Also, although it is not part of the Computer Music Project, an IBM PC AT system will be configured this fall with funds provided by the UCSC campus EXCEL project and IBM Corp.

Besides the two essential items, we will discuss here a number of desirable components to add to our system. However, the exact order that we will
acquire these components will depend on funding and on negotiations among those involved in the grant proposals (so far these have been James Beauchamp, Herbert Brun, John Melby, and Sever Tjp). This is complicated by the fact that these people have very diverse interests.

Another factor to keep in mind is the rapid rate of development in digital technology and the consequent obsolescence of existing equipment. At certain junctures we will have to decide between the purchase of new equipment to supply new functions and the replacement of old equipment in order to improve already established functions.

4.1 Audio Equipment. This equipment was not included in the original proposal as we knew that it could always be configured as a provisional system from spare parts. Now that we have a little more money, we will purchase a high-quality stereo amplifier, loudspeakers, and a conventional tape recorder. Another high priority item is a digital encoder (e.g., the Sony PCM-1015/RX combination), which is probably the most cost-effective way to make wide dynamic range (96 dB) stereo recordings.

4.2 Tape Unit. A conventional 9-track start-stop tape machine is essential for at least one of our composers (John Melby) because of his need to use the synthesis program MUS (which only runs on the computer IBM 4341). A 4500 ft., 1600 BPI tape will hold about 14 minutes of 16 bit, 44 KHz 32 sound. Similarly, 3000 BPI will hold about 24 minutes and 2500 BPI will give 47 minutes. The tape unit will be useful for backing up music from the sample disk as well as for regular file storage.

4.3 Increased LMC and SCSS Storage. As more users are added and project scopes build up, it will be necessary to add more memory. Fortunately, prices are decreasing rapidly. More RAM in the LMC will improve its performance when larger programs are run and more people use it simultaneously. Also, more LMC hard disk capacity will be needed as more users and programs are added. The present 85 Megabyte capacity of the SCSS sample disk is too small to satisfy more than a handful of users and we hope to increase this to 500-400 Megabytes in the near future.

4.4 SCSS Upgrades. One project upgrade is to add two more channels of I/O to complete a four-channel system. Another intriguing possibility is the UMass Music Group's Fast Audio Signal Processor (designed by Lippold Raskin), which can be added to the SCSS for approximately $5000. With suitable software, the FASP will allow us to increase synthesis throughput by a factor of 100 or so.

4.5 Laser Graphics Printer. Several of our users feel that high-quality hard copy is essential. Herbert Brun is well known for his computer-generated line drawings as well as his music; both Sever Tjp and John Melby need music notation output, both conventional and avant-garde; James Beauchamp uses graphics extensively in his work in analysis and synthesis of sound. High-resolution (400 ppi) laser graphic printers are available now at fairly low prices (around $3000) with quality approaching that of fine human draftsman.
The symbol, structure-oriented nature of MHC makes it easy to change the association between score parameters and instrument parameters, independent of the instrument definitions. This facilitates experimentation and re-orchestration in compositions. The technique uses structure templates called "stencils", resulting in addressing mechanisms which incur less execution overhead than traditional array-oriented methods.

Scores for MHC can vary a great deal in size and complexity. Experimenter sketches might use a few simple voices; finished compositions could involve several complex voices and control parameters. For these reasons, MHC allocates much of its storage dynamically, that is, as the program executes. By doing this, smaller works use proportionally less memory and execute faster, enabling the system to support more users. But much larger scores may also be accommodated; in fact, there are no hard-and-fast limits placed on the number of instruments or the number of instrument parameters. The upper limits are determined entirely by the computer system running MHC.

Currently, our method of operation with MHC consists of two steps. First, we link the user's orchestra (e.g., orcorn) with the score-orchestra version of MHC using the shell script:

```
$ mcr
```

The resulting linked MHC then runs with the designated score file (e.g., test.score) to produce a sample file output (e.g., out-score).

```
$ mhc out-score test-score
```

Two projects are under way to augment the usefulness of MHC. One is to make a library of unit generators, function generators, and instrument definitions; initial tests show that with simple saw waves and the FARPLU-String synthesis algorithm (Kosinski and Strong) the other project is to implement a preprocessor which translates stream-oriented musical notation files into the standard score input for MHC.

In a test of performance MHC took about 5 times real time to execute a single FARPLU-String voice on the RS 11/753. The same program took about 16 times real time on the LMC.

MHC has proven to be very portable and now runs on the following machines/operating systems:

- RS 11/753/4.2 BSD UNIX (at the campus Computer Services Office)
- LMC Magnifica/GERI (at School of Music's Computer Music Project)

Also, projects in the near future are:

- IBM PC AT/PC-DOS
- DEC RAINBOW/MS-DOS

5.2 Graphics (general purpose)

`G' Draw is a library package of functions written in C by George Chaitas for our Silicon Video 100 terminal, which emulates the Tektronix 4010 graphics protocol. Most functions work also for other VT100
emulators and also for 4070 emulators, which are frequently used as part of terminal emulation packages for small microcomputers. The package is structured in two layers. The bottom layer is the device library, which can be modified to handle other graphics devices, and the top layer is that which is accessible to application programs.

The graphics functions are divided into seven types: package control (initialize and terminate graphics); viewing (set window and mapping of user's coordinates into window); attributes (set line and cursor styles); drawing (move and draw vectors); text (display text); input (operator cursor control); and errors (return error conditions).

At this writing we have begun to use and plan to use this package much more extensively for various applications.

5.3 Computer-Assisted Composition

Two quite different projects in this area are occurring at UUCP. One is the MPS project directed by Sverre Tielp, who is the author of a paper in these proceedings on this subject [Tielp, 1985]. The other is the SANGEST project directed by Herbert Brun, a principal speaker at the 1985 ICMC.

Thus far, MPS is our only serious Fortran project. MPS is currently operational on the campus Cyber 175 computer and is in its final stages of adoption for the LMC by Curt Bergmann, a UUCD student. A program which approaches the CAC probe (from a musical viewpoint, MPS consists of an infrastructure of routines, including those which handle stochastic processes and sieves; it may be used to realize to a user's specification a number of compositions. Tielp has described the use of MPS in two previous proceedings [Tielp, 1975, 1980].

Herbert Brun's SANGEST is a C-Language program implemented on the VAX 11/780, and we expect that it will be ported to the LMC later this year. Unlike MPS, SANGEST is both a CAC program and a synthesis program; it computes as efficiently on the VAX as it does on the Cyber 175. Although it uses the most music synthesis programs, which are based on analyzing radar waveforms, in terms of "inks" and builds entire compositions using operations such as "ink", "singe", "merge", and "vary".

5.4 Sound File Management Software

Eurt Nebel has written a series of C functions which will be used to manipulate sample files on the Sound Conversion/Storage System. Most of the code to implement these functions resides on the SCSS's 58800 processor, so only a small portion is used on the LMC host to control these functions. The functions will allow the user to create sample files, have them scanned, copy them, concatenate them, rename them, delete them (partially or wholly), play them (to the SAC), or record them (from the SAC). The sample files reside on the SCSS's 360000 floppy disk. More details are given in Eurt Nebel's Proceedings article [Nebel, 1985].

5.5 Sound Editing, Analysis, and Data Reduction

This project tries to combine various programs and combines sound analysis, synthesis, and graphics. Our first goal will be to implement most of the features of the TONEAR package, at present written in Fortran and operation on the CYBER 175. We are now in the process of making a C language version of TONEAR. The two goals of this project are to simulate live sounds using data reduction techniques and to use analysis/ synthesis to gain a better understanding of musical timbre.

Some of the TONEAR functions are: Interactive graphing of signal waveforms; amplitude vs. time, frequency vs. time using "correlation filter", amplitude and phase vs. time (using "heterodyne filter"), "brightness" vs. time, average and "segment" spectra; time-variant spectrum analysis (constant or variable size), additive synthesis, pulse-in-linear approximations to smooth parameter functions, brightness and amplitude scaled time-variant non-linear synthesis; orchestra development for data reduction synthesis of acoustically-derived sounds with data links to a general purpose synthesis program (Music NFS on the Cyber, in the future MAC on Data). [Se敷山, 1983].

With suitable orchestra definitions general sound editing can be accomplished using MAC. However, this is a "hobby" rather than a "professional" operation. We plan in the future to implement an interactive editing program which will allow fast on-the-spot editing of sample files.

5.6 Manufacturing Software

At least two of the Project grantees (Kunitc and Tielp) frequently find themselves in the position of having to transcribe computer music output for conventional or extended-conventional notation. Thus, we are very interested in implementing a music notation program for computer. Herbert Brun's Graphics, hard copy can be obtained by screen dump from the computer.

5.7 Psychophysical Testing and Data Collection

A particular project of interest to Se敷山 is to develop a package for threshold testing. One goal is to determine the relation between the amount of intelligent data reduction of acoustic timbres and a listener's ability to discriminate between the original and data corrupted versions. (Preliminary tests indicate that listeners can perform these tasks similarly and the ratio among discrimination across different levels of data reduction is fairly invariant with respect to training.) [Se敷山, 1989]. The package would consist of four parts: 1) sound stimulus sequencer, 2) subject response data collector, 3) statistical calculator, and 4) graphic printer of results.

5.8 Links Between Packages

The establishment of links between programs increases the individual value and the value of the entire system. For example, the analysis/data reduction package should be able to inject data into MAC orces for compositions using literal imitations of live sounds. Analysis and psychophysical testing will use a common data format for sound files. There would be a translator which allows MPS to drive MAC. And musical sounds could be defined using a common, easy-to-read alphanumeric format that can be directly and either translated into MAC standard format or into graphic musical notation output.