MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Memo. No. 129.

April 1967.

EUTERPE: A Computer Language for the Expression of Musical Ideas.

Stephen Smoliar.

The electronic medium has vastly increased the amount of material available to the contemporary composer. The various pieces of electronic equipment available today allow one to produce any conceivable sound; yet because of the complex nature of their output, these devices are generally difficult to control and the composer of electronic music may take several hours to prepare but a few minutes of his creation.

EUTERPE was designed during the summer of 1966 by Marvin Minsky as a "real-time music program" to be used at a teletype which was a direct link with a digital computer. The program is an interpreter and compiler, basically a translation device to convert symbolic input into the internal machine language of the computer. The symbolic input consists of up to six "voice-programs" which are strings of words. The words can be of two types: 1) Note words, which represent notes on the musical score, and 2) Macro instructions, simple words whose functions are equivalent to strings of words. The author was responsible for coding this program to run on the MAC PDP-6.

In the course of preparing this program, it became apparent to the author that this was more than a computer-controlled instrument which the user could "play" through his console. Currently, the computer produces the desired sounds using a Digital Equipment Corporation Precision Display, Type 340, as a digital-to-analog converter; but the actual program was designed to be compatible with almost any appropriate output device. Hence, it could be harnessed to drive an electronic synthesizer or even an array of several pieces of electronic equipment. Furthermore, the interpreter could be embedded in a higher level programming language such as LISP, through which the user could prepare functions or macros of his own design.

EUTERPE consists of two levels of programs:
1) The voice programs which present the musical ideas, and 2) the execution program which interprets the voice programs and generates the actual music. While the program is running, all six voices supply data

concerning the playing of a note to the execution program; this data breaks down into four categories: 1) duration, 2) pitch, 3) waveform, 4) articulation. On the PDP-6 the third category refers a block of memory in which a digital record of one period of a particular waveform is stored. The executive program runs in a loop at a sampling rate determined by the pitch parameter for a certain number of cycles as determined by the six duration numbers. When it runs out, it looks at the duration numbers of each voice. One or more of these will be zero, for these voices, the interpreter gets the next note in the program and fills in the new duration number. The executive program then finds the smallest of the six duration numbers, subtracts it from each voice duration number, and re-cycles for a period of time determined by this number.

As the voice programs are being interpreted, the output is compiled and stored away in memory. (The PDP-6 version uses a "low" DDT at address 34000, and the compiled program is stored in large memory beginning at address 400000). The compilation is the most primitive possible technique; each loop is written into free storage in sequence. When a loop runs out, the transfer to the interpreter is replaced by a transfer to the next loop. In the PDP-6 version, starting the program at TUNE sets up the pitch tables of sampling rates, interprets the voice programs, playing as it interprets, and and plays the compiled version. The interpreter may be stopped from the teletype; striking the space bar returns control to DDT while striking any other key plays the material compiled thus far and then returns control to DDT. Once the pitch table is computed, the interpreter may be started at SETUP. The compiled version may be played alone by starting at PLAY.

The fundamental instruction of the voice program is the note word which supplies the interpreter with the parameters mentioned above; the following is an example:

```
K4S 4T(SLUR)
```

K4S is the pitch indicator, 4T the duration indicator, and SLUR designates the articulation. The waveform is determined by a special instruction described below.

Pitch is determined by three concatenated symbols: an octave letter, a scale number, and an S,F, or blank. Octave letters may be H,I,J,K,L,M,N. Scale numbers 1,2,3,4,5,6,7, correspond to C,D,E,F,G,A,B, within the designated octave. Finally, S and F designate sharps and flats. A pitch indicator, R, designates a rest. While the current PDP-6 version is not in tune, this can be accomplished by depositing the appropriate constant in the address portion of location TUNE. In tuning EUTERPE, K6 should sound as 440 cps.

For duration, 1T, 2T, 4T, 8T, 16T, 32T denote whole notes, half notes etc. A "D" in place of "T" designates a dotted note while a "3" following the "T" indicates triplet duration value (i.e. 2/3 normal duration).

Articulation is based on the principles of Peter Samson's music compiler. Each note word is interpreted as two note words; the note itself and a note of rest for articulation. The amount of articulation is determined by the following indicators:

Indicator	Duration of note	Duration of articulation	Use
SLUR	1	0	Slurs, notes longer than dotted whole notes.
LEGATO	$^7/_8$	$^1/_8$	Standard legato.
STACO	$^1/_2$	$^1/_2$	Staccato for organ works.
SLEG	$^3/_4$	$^1/_4$	Articulated legato.
SSTACO	$^3/_8$	$^5/_8$	Short staccato.

Articulation is preset to LEGATO for all six voices and is not altered unless specified by a voice program.

If any parameter is omitted from a note word, the previous value to which it was assigned is assumed.

Hence, coding for "God Save the Queen" would look like the following:

```
K1 4T
K1
K2
J7 4D
K1 8T
K2 4T
```

Associated with each voice is a parameter list of four items: tempo, pitch, waveform, articulation. The tempo parameter is a multiplicative duration factor which is initialized by the execution program and may be altered by special macro instructions cited below. The pitch parameter is a transposition factor indicating the number of half steps up (positive) or down (negative) a note should be transposed. It is initialized to 0 and may be modified by speical macro instructions. The waveform parameter specifies which of three waveforms the voice is "playing": SINE, SQUARE, or SAW; it is initialized to SINE and modified by macros. Finally, the articulation parameter saves the articulation information specified by note words or modified by macros.

In addition to the note words, a voice program may use several macro instructions which take the place of large blocks of note words. These appear below:

JUMP$ VOICE,ADDRESS

Causes designated VOICE to transfer from the current note word in its program to location ADDRESS while remembering the location from which it transferred.

RETURN VOICE,

Returns VOICE to the location saved from the last JUMP$.

PUSHJ$ ADDRESS

Transfer to ADDRESS, remember where to return, and save current parameter list.

POPJ$

Returns from PUSHJ$, restoring old parameter list.

CANCEL VOICE,

Causes executive program to ignore indicated VOICE program.

START VOICE,

Restores indicated VOICE

PITCH VOICE,X

Set pitch parameter for voice to X. Normal mode is 0. PITCH 14 raises everything an octave; PITCH -14 lowers everything an octave; etc. (octal numbers).

RELPIT VOICE,X

Add X to pitch parameter for designated VOICE.

TEMPO VOICE,X

Set tempo parameter for VOICE to X; the parameter is multiplicative and a TEMPO of 0 is disastrous. If no TEMPO is specified at the beginning of the program, it is assumed to be 4.

RELTEM VOICE,X

Multiply tempo parameter for designated VOICE by X if X is positive; divide if X is negative.

WAVE VOICE,X

Set waveform parameter for VOICE to X.

ARTIC VOICE,X

Set articulation parameter for VOICE to X.

TRA VOICE, ADDRESS

Transfers specified VOICE to ADDRESS.

FINE

Returns control to teletype.

In any instruction which expects voice specification, the absence of such notation interprets the voice to be the one which is reading the word. The reader unfamiliar with computer programming may find the above description somewhat baffling. To clarify the situation, we present an annoted copy of a program which was run on the PDP-6 through EUTERPE:
(the circled numbers refer to the comments after the program):

```
VOICE1:     WAVE SAW                (1)
            JUMP$ TRUMPT            (2)
            K6 4D
            8T
            K6
            L1
            K7
            K6
            K5
            K3 16T
            K4S
            K5
            K6
            K7
            K6
            K5 4D
            JUMP$ TRUMPT
            K6 8T
            K7 16T
            L1
            K7
            K6
```

```
                K5S
                K4S
                K5S 8T
                K6 4T
                K5S 8T
                K6 4D
                8T
                4T
                K6
                FINE (3)
VOICE2:         JUMP$ HORN
                L1 4D
                8T
                L1
                L3
                L2
                L1
                K7 4D
                8T
                4D
                JUMP$ HORN
                L1 8T
                L2 16T
                L3
                L2
                L1
                K7 4D
                K6 8T
                K7
                K7
                K6 4D
                8T
                4T
                K6
                FINE (3)
VOICE3:         CANCEL (4)
TRUMPT:         K6 16T
                K7
                L1 4D
                8T
                L1
                L3
                L2
                L1
                K7
                K5 16T
                K4S
                K5
                K6
                K7
                L1
                K7 8T
                L2
                L1
                K7
                RETURN (5)
HORN:           L1 16T
                L2
                L3 4D
                8T
```

```
        L3
        L5
        L4S
        L3
        L2 4D
        8T
        4T
        L3 8T
        L2
        RETURN
VOICE4: WAVE SQUARE
        J6 8T
        4D
        8T
        J6
        J3
        J7
        K1
        J5 4D
        8T
        4D
        8T
        K6 4D
        8T
        J6
        J3
        J5
        J6
        J3 4D
        8T
        4T
        J3
        J1 4D
        8T
        4T
        J1
        J5 4D
        8T
        4T
        J5
        J6
        J2
        J3 8T
        J4
        J3
        J3
        I6 4D
        8T
        4T
        I6
        FINE (3)
VOICE5: WAVE SQUARE
        TRA VOICE1+1 (6)
VOICE6: CANCEL (4)
```

1. VOICE1: identifies that word as having the symbolic address VOICE1, i.e., the first word of voice program #1.

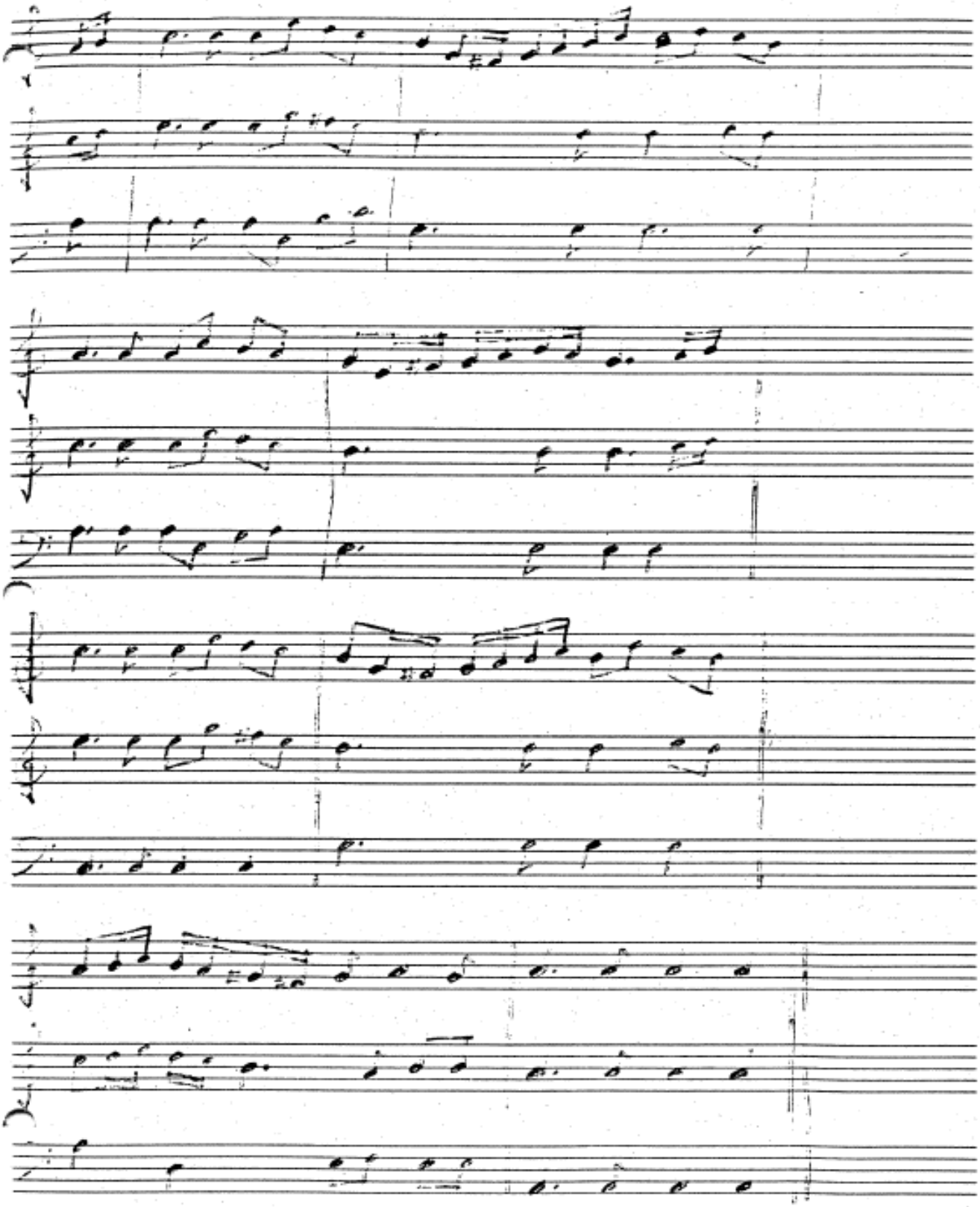
2. The opening motif is repeated so it is stored as a subroutine. It is located in memory in the free space between the end of voice program #3 and the beginning of voice progam #4 with symbolic address TRUMPT.

3. Ends the piece.

4. This voice is inactive.

5. No voice specified; hence any voice program may enter this subroutine and leave correctly.

6. Voice program #5 will play the same music as voice program #1 but with a different wave form.

All articulation is legato.
Voice program #2 plays a sine wave since its initialization is not altered.
The music notation for this example:

PASSAMEZZO (unor.)