

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

Artificial Intelligence
Memo No. 202

July 1970

Peter Samson's Music Processor, BIG

Michael Beeler

Work reported herein was supported by the Warren McCulloch Laboratory, an M.I.T. research program sponsored by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract number N00014-70-A-0362-0002.

Reproduction of this document, in whole or in part, is permitted for any purpose of the United States Government.

contents

commands which create a name	2
commands which create music	3
playing commands	4
plotting commands	6
general utility commands	7
debugging commands, relics of the past, features you might hope to live to see	9
error comments	10
appendix -- MUSCOM	10

BIG

BIG is a music processor, as opposed to some earlier music programs which were either music compilers or music players. BIG accepts the compiled "binary" output of MUSCOM, and playing is one of the processes BIG can perform upon this data. BIG is currently available as PRS;BIG DUMP and may be started with \$G and will probably become available as SYS;TS BIG so it can be loaded with a control-H. BIG currently occupies 26 blocks of core memory.

BIG has two internal data formats for representing music: structured and unstructured. To get structured music, you must (1) have a time signature (e.g. TIME 3 4) in the "english" input to MUSCOM (as opposed to UNITS; see MUSCOM update in appendix), and (2) have a control-T anywhere in the compile command to MUSCOM (e.g. ^TMUSIC BIN_MUSIC ENG). Otherwise, you get unstructured music. Unstructured music, which is for playing, is a list of events (commencement of a given pitch in a given voice or its stopping), with times between events. Structured music, which is for drawing music on the plotter, corresponds fairly closely to a score.

BIG has an initial vocabulary of several items: some are number constituents, some are commands, and some are field delimiters. Words in BIG are either

(1) any single character from the set

^ " # , - / \ () < > [] : ; ? @
0 through 9

or

(2) a string of characters of the same type, delimited by space or any character not of that type.

Characters of type zero are

! \$ % & .
A through Z

Characters of type one are

*
+
=
^ (up-arrow, shift N)
_ (back-arrow, shift O)

Numbers are always decimal (except in debugging commands), and are made up of the following characters or equivalent spelled-out words:

the digits 0 through 9 and the words ZERO through NINE
the word K and the word THOUSAND
the word M and the word MILLION
the sign - and the word MINUS

(these are unary, not binary minus; 2-3 = -2)

spaces to separate words, and any non-number constituent to delimit the entire number

Field delimiters in BIG are comma, colon, semicolon, and space.

Names in BIG are words formed of type zero characters. A name is assigned to one of the following:

- (1) an unstructured piece
- (2) a structured piece
- (3) a harmonic structure table
- (4) nothing (i.e., a free name, a null name)

When a name is created it is a nothing.

BIG attempts to decipher what you mean even in the face of typing errors; for example, MENTION OWL will produce the same effect as the command MENTION ALL.

Commands to BIG are explained below. Extra spaces may be inserted anywhere between numbers, commands and field delimiters. A command is typed entirely on one line and terminated by a carriage return. Typing a RUB OUT character in a command line causes BIG to immediately disregard all characters on that line and start a new line. Further typed input is not echoed until the command is completed. In the case of ENTER, PLEASE ENTER, READNEW, READ, WRITE, ENTITLE and INTREAD, more information is needed; on the line following the command line, BIG types a \ (backslash, shift L) to prompt you to supply this information, which you will terminate by typing a second \. Typing a RUB OUT character in a field BIG has requested by a \ will echo and erase the last character of that field. Fields enclosed in angle brackets in the following descriptions <> are optional. Except for "name", "nx" is used to denote a number of some nature x.

commands which create a name

/
You type a / (slash) in any context requiring a new name. BIG does not echo the slash, but instead echoes a new, distinct five-character name.

Example one:

you type /
which it echoes as FZYLC (or some other five characters). This string of five characters is now a name, assigned to nothing.

Example two:

you type BUILD / 1:1,400 (followed by a carriage return)
which it echoes as BUILD FZYLC 1:1,400

ENTER

\name<,name2<,name3...>>\

You type ENTER, followed by a carriage return. BIG types a \ (backslash, shift L), after which you type your recommendation for a name, ended by another backslash. If your recommendation is

acceptable, BIG will echo it as heard; if it is too similar to another name, BIG will print a complaint. Only the first ten characters are accepted. Independent recommendations for different names may be included in one command if separated by commas.

PLEASE ENTER

\name\

Takes same format as ENTER but causes BIG to loosen its strictures a bit, perhaps allowing you to get in your favorite name.

commands which create music

READNEW nv name

\<dev:><usr;>fn1 fn2\

or

READNEW name nv

\<dev:><usr;>fn1 fn2\

Reads from the disk (or microtape, see TAPES below) the first "nv" voices in the piece specified by the file name "fn1 fn2" and assigns the name "name" to the music. BIG types the first \; you type the file name and second \. A file in another user's directory is specified by the optional "usr;" and a different device which uses the same format as the disk by the optional "dev:". READNEW comments, "unstructured" or "structured" as it reads in a piece. If fewer than "nv" voices exist, you will win if $(\text{"nv"}+2)/3$ (integer divide) equals $(\text{actual number of voices}+2)/3$. This is because music is read as blocks of one word of time followed by as many words of 3 pitch fields each as necessary to make up the required voices. "name" must not be a harmonic structure table.

TAPES n1<n2<n3<...>>>

All tape and disk directories are forgotten; then tape drive numbers n1, n2, etc. (order immaterial) become microtape drives of which BIG is aware. Subsequent READNEW commands will search all such directories for the file to read. Disk is referred to as drive 0. Default case merely forgets all directories.

READOLD nv name

Reads "nv" voices from PDP-10 paper tape reader.

DETEMPO ndetemp

Applies a buggier factor "ndetemp" as READOLD reads in; a satisfactory initial default value (883) exists.

READ name

\<dev:><usr;>fn1 fn2\

WRITE name

\<dev:><usr;>fn1 fn2\

Loads (READ) or dumps (WRITE) any of the three things a name can be onto the disk or "dev" in BIG's internal format. See READNEW for

complete details of file name specification.

MELGEN

Currently inactive command to create a melody by a random number generator. See Peter Samson if you want it loaded into BIG instead of the dummy program now loaded in its place.

KEYBOARD

Currently inactive command to read in a piece as you play it. Hardware interface lacking.

TRANSLATE <FROM >struname <TO >unstruname

Translates structured music of name "struname" to unstructured music of name "unstruname". The name "unstruname" must be assigned either to unstructured music or to nothing.

playing commands

PLOLD <nsec >piecename

Starts six voice, square wave, right six bits of PDP-6 memory indicator (MI, location 32) player. If present, "nsec" seconds of silence precede playing.

PLSCOPE <nsec >piecename

Starts analog player. Four nine-bit fields of the PDP-6 MI (location 32) feed four digital-to-analog converters (DACs), whose outputs are zero to +1.5 volts (0-15 ma current source driving 100 ohms). Associated with each of up to four voices is a harmonic structure table, which is a group of four stored, sampled one cycle of fundamental plus harmonics at amplitudes (9 bits) pre-chosen by the BUILD commands below. Each of the four cycles in a harmonic structure table is added in to its respective output channel to form the total, four-voice signal for each of the four output channels. Harmonic structure must be defined in some way (including, for example, READ) before PLSCOPEing; no initial default harmonic structure table exists.

BUILD name< noutchn>:nh1,ha1<;nh2,na2<;nh3,na3<;...>>>

Defines a harmonic structure table. "noutchn" is the number of an output channel, 1 through 4. If none or zero is specified, the following is applied equally to all four channels. Any previous harmonic structure table with name "name" and output channel "noutchn" is lost. The nh1,na1 etc. are a list of harmonics desired and their amplitudes. The harmonic with its "nh" -1 is the fundamental; the cycle is stored as 256 samples, so it makes no sense to specify a harmonic "nh" greater than 128, but BIG will add to the harmonic structure table even absurdly high harmonics. The amplitudes should be chosen to avoid overflow; having them sum to 700 or 800 should be safe. If an overflow occurs during BUILDing, BIG will type an error comment and leave that harmonic structure

table in a state of disarray. Harmonic 0 is white noise. Examples:
 BUILD ORGANLIKE 1:1,200;2,100;7,100
 BUILD FLUTE 2:1,400;2,200;8,100

BUILD name< noutchn><:>

Clears the output channel "noutchn" part of harmonic structure table "name". With no or zero "noutchn" specified, all four parts are cleared.

BUILDON name< noutchn>:nh1,na1<;nh2,na2<;nh3,na3<;...>>>

Adds amplitudes of harmonics into previously built harmonic structure table "name". Amplitudes may be less than zero to decrease those harmonics. Without a channel number "noutchn", or zero "noutchn", amplitude increments/decrements are applied equally to all four channels.

VOLBUILD name< noutchn>:nnnd

The amplitude of the output channel "noutchn" part of the harmonic structure table "name" is multiplied throughout by the rational fraction "nn" divided by "nd". Without a channel number "noutchn", or zero "noutchn", all four parts are so multiplied.

VOICE nvoice name

or

VOICE name nvoice

Associates with the voice "nvoice", 1 to 8, the harmonic structure table "name". If voice "nvoice" does not select its own harmonic structure table (see STOP, below), then it will produce sound on whatever output channel(s) and at whatever harmonics and amplitudes have been built into harmonic structure table "name".

STOP nstop name

or

STOP name nstop

Associates the harmonic structure table (as in BUILD command) or the nothing "name" with the stop numbered "nstop". STOP commands can be included in MUSCOM input to put data in unstructured output associating a part of it with a particular stop, numbered 1 through 24. This data is ignored if the music is PLOLDED; if PLSCOPEed, BIG looks in BIG's stop table, where you are expected to have done a STOP command in BIG to associate with each stop used a harmonic structure table. In MUSCOM input, STOP 0 is the harmonic structure table which will be associated with the VOICE command in BIG.

WHOA

Stops playing at any point and types out where in the piece it is, in the number of events from the beginning of the piece. Works with either player.

RESUME< nevents>

Resumes playing at "nevents", which may be anywhere in piece. Works with either player once the piece has been played or started to play. Default is where piece was last WHOAed.

NOPLAY

Releases the PDP-6, allowing someone else to use it. Subsequent PLOLD or PLSCOPE commands will re-establish the link to the PDP-6; meanwhile, WHOA and RESUME will have no effect.

TEMPO ntemp

Causes playing tempo to change to "ntemp". Command operative even while playing is in progress because of autonomous players. Works with either player. Initially 65536.

TUNEUP

Tunes up either player from fresh. Will interrupt playing.

TUNING ntune

Sets A to "ntune" cycles per second (e.g. 440, its initial value). Range is not checked. Will interrupt playing.

plotting commands

SCRIPT name< FROM nm1>< TO nm2>

Plots structured music "name" on Calcomp plotter. Pen should be positioned extreme "carriage left" = upper left-hand corner of plotted output. If "FROM nm1" is present, plotting commences with measure "nm1" instead of the beginning; if "TO nm2" is present, plotting concludes at the end of measure nm2 instead of at the end of the piece. If the music has a title (see ENTITLE command below), the title will be plotted at the top.

SCRIPTSIZE nsize

Sets a linear scaling of how large plotted output is. Initially it is 48 (large); a reasonable range is 16 to 64; 24 is nice. Range is not checked.

SCRIPTMNINT nmrint

Sets script measure number interval. Measure number is plotted for first, last, and multiples-of-"nmrint" measures. Initially one, range is not checked.

SLURDENS ndens

Sets how heavily slurs and ties are plotted. Range is 1 to 4, initially 2.

SCRIPTSPACE nspace

Sets amount of horizontal "breathing room" in staff, applied after all notes, etc. have been given room. Contraction/expansion ranges

from -20 to 20; it is initially zero. A multiplier of 5/4 is automatically applied to the spacing whenever words occur in one or more voices.

SCRIPTCLEF nvoice clefname

or

SCRIPTCLEF clefname nvoice

Ordinarily, BIG automatically chooses its clefs from treble, bass, treble octave up, and bass octave down in an attempt to keep notes roughly on the staff and to not change clef too often. SCRIPTCLEF forces a preferred clef which BIG will stick to whenever it can.

"nvoice" is the voice, 1 through 9 (non-leading digits ignored), and "clefname" is one of the following:

SOPRANO

ALTO

TENOR

BASS

TREBLE

? (used to cancel a previously specified preference)

ARDS nards

If "nards" is 1, music is plotted on the Advanced Remote Display Station if one exists; if "nards" is 0, as it is initially, music is plotted on the Calcomp plotter.

general utility commands

VALUE parameter< parameter2< parameter3...>>

Current decimal or symbolic value of variable set by command(s)

"parameter" is typed out. "parameter" may be any of the following:

DETEMPO

SCRIPTCLEF

SCRIPTMNINT

SCRIPTSIZE

SCRIPTSPACE

SLURDENS

TEMPO

TUNING

ENTITLE name

\your verbose title\

Associates a string of characters of your choosing with the non-nothing "name". This title is plotted by the SCRIPT command, listed by the various MENTION commands, and its association with "name" and the music or harmonic data is maintained during WRITE and READ commands.

MENTION ALL

Lists all names which are not nothings. What the name is is shown by a symbol, as follows:

* structured music
 # harmonic structure table
 (space) unstructured music

If a piece of music has a title (see ENTITLE command), the title is also listed.

MENTION name1< name2< name3...>>

Lists only those names "name1", "name2", etc. as in MENTION ALL command. Useful when you wonder what kind of thing a name is assigned to.

MENTION EXCEPT name1< name2< name3...>>

Lists all names which are not nothings except "name1", "name2", etc. as in MENTION command.

= 0

Special case of a debugging command; lists all names which are nothings.

FORGET name1< name2< name3...>>

Clears up some space in BIG by defining as nonexistent one or more pieces of music or harmonic structure tables; the names become nothings again.

PLEASE FORGET ALL

The PLEASE is required, as erasing all definitions is a drastic step.

FORGET EXCEPT name1< name2< name3...>>

Erases definitions of all names except "name1", "name2", etc.

AGAIN

With no arguments, reperforms the last command line. Each argument if given is compared for type in the command line to be reperformed and replaces the first word of that type in the old command line. Successive arguments search from the point where the argument before them went in. No complaint is emitted for an unused argument. No argument past an unused argument is used. As a special case, an argument of type "0" (undefined extant word) will match types 27 and 30 (structured and unstructured music). If substituting numbers, remember each digit is a separate word.

^

If the up-arrow (shift N) is the only command on the line, BIG returns control to DDT; otherwise it is an error.

ALT MODE

A line containing an ALT MODE character and ended with a RUB OUT character causes BIG to type an up-arrow; if you then type a carriage return, BIG returns control to DDT; otherwise, BIG ignores

the line with the ALT MODE character.

[<niterate>commands, names, carriage returns, etc.]
The character string enclosed in square brackets [], exclusive of "niterate", is re-processed as if it were typed "niterate" times. The default case is one. Example:

```
[3
SCRIPT CANTATA
]
```

debugging commands, relics of the past,
features you might hope to live to see

JRST njrst

Transfers control to octal location "njrst" in BIG.

= noctype

Types out all symbol table entries of the octal type "noctype"; type zero is names which are nothings.

*

As first item in command line, causes word translator to echo each command as it is performed.

**

As first item in command line, causes word translator to echo each command as it is processed, but not actually perform the commands.

VIEWWAVE name

Would display a time-domain plot of the harmonic structure table "name" on a scope whose vertical and horizontal deflections were driven from the DACs on the lowest two 9-bit fields of the PDP-10 MI.

INTREAD name

\<dev:><usr:>fn1 fn2\

Reads structured music; identical to "READNEW | name" since structured music appears to have only one part because its data structure has the number of parts as the first item.

++

Will eventually be a very crude editor of unstructured music. Meanwhile, it may cause BIG to die.

VALUE BUILD

May be implemented as a Fast Fourier Transform to give what is actually in a harmonic structure table, as opposed to what BIG thinks it put there.

A feature to quit out of plotting may be implemented.

A feature allowing PLSCOPE to play a variable number of voices, thus giving better response if there are fewer voices, may be implemented.

error comments

Error comments in BIG are mostly self-explanatory. They include:

? word ("word" was not recognized as a word; whole line is translated before execution is attempted)

/ (invalid argument)

// (no core available)

?? (internal error)

TOO SHORT

RESEMBLES name

BAD DATA IN FILE

CHECKSUM ERROR

FILE UNEXPECTEDLY ENDS

BAD SYSTEM NAME

FILE NOT FOUND

NO TAPES

OVERFLOW

CHECKSUM ERROR, WILL RE-READ IF YOU TYPE A SPACE

CAN'T GET PAPER TAPE READER

STOP NOT SPECIFIED

or

or

VOICE HARMONICS name NOT BUILT

SORRY, NO PLAYER ROUTINES

SORRY, NO PLOT ROUTINES

PLEASE START THE PDP-6 AT 100

CAN'T GET THE PDP-6

BIG, when started, links to the PDP-6, loads both players, and tunes them up. The CAN'T GET THE PDP-6 error comment, which may appear at this initial tune-up time, need not disturb you unless you wish to play music.

appendix -- MUSCOM

Following is an update of AI Memo No. 107, "Music Playing on the PDP-6," and in particular the music compiler MUSCOM.

TIME

The phrase "TIME 3 4" should be used in preference to "UNITS 24" etc. A piece with "TIME" instead of "UNITS" may be output in structured form instead of the normal unstructured form by including a control-I anywhere in the MUSCOM command string.

TEXT

Text may now be included in the "english" of a piece of music. At the end of each vocal line, instead of "END" have the command "TEXT" followed by a space followed by the text. Slashes are ignored. However, you must put a slash within about 80 characters after the "TEXT" to fool the input routine. End your text for a given voice with a backslash.

For a polysyllabic word, put hyphens between syllables. If a syllable is held over a plurality of notes, followed by another syllable in the same word, N+1 hyphens will hold the first syllable over N additional notes. If the last syllable of a word is held over a total of N notes, follow it in the text by N-1 underlines. Equal sign may be used instead of underline and in fact is recommended.

The -ldollars feature of TECO enables you to enter the text in both upper and lower case. Ask a system hacker for details. Warning: like upper case letters, backslash (to end the text) must be "quoted" with a slash, as must slash itself. To get the text out on the plotter with BIG (this being currently the only use of the text) remember that your MUSCOM compilation command string must contain a control-T.

NOUNITS

The command "NOUNITS" now exists. Its only effect is to suppress typeout of "MTL" and "MTS" errors. Do not use it to replace the "UNITS" command, which should always be given.

BEAT

The command "BEAT" is followed by two numeric arguments. The second argument specifies, in 32nd notes, how often a beat is to occur. Any non-legato note which ends on a multiple of this beat time is subject to adjustment. The adjustment, if necessary, reduces the articulation of said note so that at least a certain amount of resting exists in the end of the note. This amount is set by the first argument to the "BEAT" command, in eighths of 64th notes.

TEMPAR

The command "TEMPAR" allows tempo to change in one voice without affecting the tempo of the other voices. Its single numeric argument, 5-682, is directly proportional to how long following notes will be sounded; TEMPAR 400 is twice as slow as TEMPAR 200.

MM ntime ncount
 MMPAR ntime ncount

The commands MM and MMPAR vary tempo similarly to TEMPO and TEMPOR. MM specifies tempo of all voices, MMPAR that of the single voice in which it occurs. Their format is closer to musical notation. The first argument, "ntime", is the time field of a note, 1, 2, 4, 8, 1., 2., 4., or 8.; only these eight values are permitted. The second argument, "ncount", is a metronome count telling how many of these notes occur per minute. MM 4 110 means 110 quarter notes per minute.

W

Embellishment "W" is the reverse of embellishment "M"; instead of going down to the basic, it goes up from the basic.

When MUSCOM detects an error, it types out the entire measure in which the error occurs (measures being separated by slashes), preceded, if it is the first error found in a given voice, by a colorful message. If one word in particular is erroneous, it will be surrounded by < > brackets in the printout.

Error codes:

number The measure so numbered ended at different times
 in different parts.

AGM Argument Greedily Masticates
 A number was expected and delicious garbage found.

AIR Accidental In Rest
 It is ignored.

AOR Accidental Out of Range
 Though basic note not.

BBL Bad Bar Label
 A number appeared that does not agree with the
 program's current measure number. Maybe you
 forgot to end your title with a slash.

BLB Bad Left argument to Beat
 Greater than 255.

BLC Bad Left argument to Copy
 Less than 1 or greater than last measure.

BRB Bad Right argument to Beat
 Greater than 127.

BRC Bad Right argument to Copy
Less than left argument.

BTB Bad Time Denominator
Not in range 1 - 64.

BTN Bad Time Numerator
Not in range 1 - 127.

DTU DoT Underflow
Too many dots and X's in a note caused the value
of dot to fall below 1/64 note. It is ignored.

EIT Embellishment on Insufficient Time

EOR Embellishment Out of Range
Though basic note not.

ERT ERRoneous Time
The time field of a note was not a power of two.

ETR Embellishment on TRiplet
Mumble mumble maybe Peter will clean this up some day.

ILC Illegal Location for "COPY"
Namely, not at beginning of a measure.

ILR Illegal Location for "REST"
Not at beginning of measure.

ILT Illegal Location for "TIME"
By now you should know...

ITG Insufficient Time for Grace note
The note after one containing G was too short.

MTL Measure Too Long
Compared to "UNITS" or "TIME" specification.

MTS Measure Too Short

MYK MYsterious Key
"KEY" not followed by +) (- or = .

NIU Non-Integral Units
Time signature * 32 not an integer.

NOR Natural OverRides
When) or = appears in the same note with (+ or - .

- NPS No Psuch Seudo
An unidentified word with no numbers or commas appeared.
- TFF Too Few Fields
- TIC Time In Comma note
- TMC Too Many Commas
More than one comma appeared in a note.
- TME Too Many Embellishments
More than one of P, W, U, N, M, D appeared in a note. Last one takes precedence.
- TMF Too Many Fields
A number, a comma, and the letter R are fields. A note should have a maximum of two fields.
- TMG Too Many G's
More than one G appeared in a note.
- TMR Too Many R's
More than one R appeared in a note.
- TMS Too Many Staccato, legato, etc.
More than one of S, L, E, H, Q appeared in note. Last one takes precedence.
- TS Too Slow
Tempo number greater than 682.
- TTF Tempo Too Fast
Tempo number less than 5, or second argument of MM or MMPAR greater than 1023.
- UAT UnAvailable Tone
Treble clef's EB2 through A1 are available. Sorry.
- UCH Unused CHaracter
A meaningless letter or other character was found in a note.
- UNC UNprepared Comma
A comma appeared in a note, but no note came before it in this voice.
- UNV Unavailable Note Value
First argument to MM or MMPAR not 1, 2, 4, 8, 1., 2., 4., or 8.

WLK Wrong Location for "KEY"

Not at beginning of measure. Will apply
successfully to unstructured output but
will be ignored in structured output.

DNT WLK