SCRIPT REFERENCE MANUAL

Publication SCRIPT-2, July 15, 1981

REFERENCE MANUAL CONTENTS

The SCRIPT Reference Manual is a compact, but complete description of the SCRIPT system. Experienced programmers may turn to this Manual immediately to learn the details of the system. All monitor commands are summarized as well as all SCRIPT statements, along with formats, value ranges, and default settings.

CONVENTIONS USED IN THIS MANUAL

In this Reference Manual, we have adopted certain conventions to demonstrate the format of the various monitor commands and lines in a SCRIPT composition. These conventions are not part of SCRIPT itself; you will not be using them, except when reading this manual.

All monitor commands and SCRIPT words are presented in upper case. Expressions describing the values to be chosen by the user are presented in lower case and enclosed by brackets or parentheses. *Brackets* around the expression indicate that a *fixed* number of user supplied values must be entered in that location, while *parentheses* indicate that a *variable* number of user values can be entered. *Italics* indicate an *optional* value that may or may not be entered.

NOTE WELL: You do not use brackets, parentheses, or italics *in this way* when entering compositions.

MONITOR

The following conventions are used to describe values to be supplied by the user in monitor commands:

[filename]	name of file to be stored on or recalled from diskette
[ln]	single line number
[ln–ln]	two single line numbers separated by a hyphen, representing a block of lines inclusive of line numbers indicated
(linenumbers)	a variable number of single line numbers or blocks of lines, separated by commas (for example, 22, 33–45, 66)
[string]	a specific sequence of characters occurring in your text
[sequence]	the number (from 1 to 6) of a Synclavier® II sequence, which may be stored or recalled during Synclavier® II performance by use of one of the six RECORDER STORE/RECALL buttons on the Synclavier® II panel

SCRIPT LANGUAGE

The following conventions are used to describe values to be supplied by the user in timbre definitions, music notation, the computer music format, and tempo control and synchronization.

EXPRESSION	MEANING
[ln]	line number
[symbol]	alphabetical symbol
(pitches)	the list of pitches on one line
(rhythmic values)	the list of rhythmic values on one line
[value] or [values]	numeric value or values, the range of which is specified under the particular function being described
[time value]	time in seconds or in frames
[beat rate]	the time specification for a beat: either the number of beats per minute, or the length of the beat in seconds or in frames

The monitor is controlled, through the terminal (either screen or hard-copy), with the user commands summarized below. Only the monitor responds to these commands; they are ignored if typed during Synclavier® II operation, or when the compilers are running.

The monitor program is placed in the computer memory and run when the system is first loaded; the monitor program is restored whenever either the S key or any of the keys that activate the reverse compiler are pressed during Synclavier® II operation.

The monitor is based on the concept of the current file, or scratch copy, of the composition placed in computer memory. When the computer is turned off or when the SCRIPT conversion program is run, the current file is lost.

In all definitions below, the word composition means the composition in the current file.

Compositions in SCRIPT consist of lines of typed text that, in contrast to monitor commands, all begin with line numbers. When you finish typing a line and press the RETURN key, the monitor will sort it by line number, not by the order in which you type it. In the case of duplicated line numbers, the last one will replace the earlier version.

PERFORMANCE COMMANDS

PLAY The PLAY command initiates complete compilation and playing of a SCRIPT composition.

RUN The RUN command runs the utility programs (described in the Utility Programs Instructions) or compiled MAX/XPL system programs.

COMPILE The COMPILE command converts a composition into a Synclavier® II sequence and returns control to the monitor. The compiled sequence replaces the current file; the current file name is changed by the monitor by adding a period (.) to the end of the current file name. The compiled sequence may be stored on a Synclavier® sequence diskette by use of the STORE command, or it can be saved on a SCRIPT user diskette by use of the SAVE command. A compiled sequence may be played by the PLAY command.

BASIC FILE MANAGEMENT

NEW The NEW command initiates entry of a new composition and gives it a file name.

NEW [filename]

OLD The OLD command places a copy of a previously saved file in computer memory.

OLD [filename]

RENAME The RENAME command changes the name of the current file.

RENAME [filename]

SAVE The SAVE command saves a new file on a user diskette.

SAVE [filename]

If the file name is omitted, the file will be saved under the current file name. If a file already exists with the same filename, an error message will result.

REPLACE The REPLACE command replaces a file on the diskette with the current file.

REPLACE [filename]

If the file name is omitted, the file with the same name as the current file name will be replaced. If no file exists with the specified name, an error messege will result.

UNSAVE The UNSAVE command erases a file from the diskette.

UNSAVE [filename]

When using this command, you only need to specify a file name when the file you wish to erase has a different name from the current file name.

STORE The STORE command stores a compiled composition on a Synclavier® II sequence diskette for ready access during Synclavier® II operation by use of the RECORDER STORE/RECALL buttons.

STORE [sequence]

RECALL The RECALL command places a copy of a Synclavier® II sequence in the current file. The number of the sequence will become the current file name.

RECALL [sequence]

CATALOG The CATALOG command lists the files saved on the diskette.

CATALOG LENGTH The CATALOG LENGTH command lists the saved files and the number of computer words in each file.

CATALOG ALL The CATALOG ALL command lists the saved files as well as the "hidden" system files.

NAME The NAME command lists the name of the current file.

LENGTH The LENGTH command prints the length of the current file in words.

LAST The LAST command tells you the number of the last line in the current file.

The next two commands are used only in maxidiskette systems.

ACCESSING FILES FROM THE OPERATING SYSTEM DISKETTE

[filename]/F0 The /F0 symbol is inserted after the file name to access the files stored on the left diskette drive.

OLD FORMAT/F0

This command would make the FORMAT utility program, which is stored on the system diskette in the left-hand drive (in maxi-diskette systems), become the current file.

ACCESSING FILES FROM REMOTE DISK DRIVES

[filename]/R0
[filename]/R1 The /R0 or /R1 symbols are inserted after the file name to store a file on remote minidiskette or to recall a file already stored there. /R0 accesses the drive attached to MAIN DISK DRIVE and /R1 accesses the drive attached to AUXILIARY DISK DRIVE.

May be used with OLD, SAVE, REPLACE, STORE and RECALL commands for exchanging files between maxidiskettes and minidiskettes.

EDITING COMMANDS

Editing commands are used to make changes in the current file. In maxidiskette systems, the Screen Editor is also available for editing. See the Screen Editor Manual for instructions.

SED The SED command invokes the Screen Editor (maxidiskette systems only).

LIST The LIST command lists the entire composition or selected lines.

LIST *(linenumbers)*

A single line number, a block of line-numbers, or a series of either or both can be entered after the word LIST. All specified line numbers will be listed on the terminal. If LIST is used with no line numbers, the entire composition will be listed. To stop a listing before it is completed, press the BREAK key.

If you are using a CRT terminal, the composition will be listed in 19-line seqments followed by pauses. You may list successive 19-line segments by pressing RETURN. If you are using a hard-copy terminal, the composition will be listed without any pauses. (If your system does not operate as described here, the terminal type on your system diskette is set incorrectly: see "Diskette Configuration" in the Options Setup Manual.)

PRINT The PRINT command lists the entire composition or selected lines on a hard-copy printer attached to the PRINTER/MODEM socket.

PRINT *(linenumbers)*

A single line number, a block of line numbers, or a series of either or both can be entered after the word PRINT. If you use the word PRINT alone, the entire composition will be listed.

SEQUENCE The SEQUENCE command renumbers the lines of a composition starting with 1 and incrementing by 1.

RESEQUENCE The RESEQUENCE command renumbers the lines of a composition starting with 100 and incrementing by 10.

DESEQUENCE The DESEQUENCE command removes the line numbers of composition.

EXTRACT The EXTRACT command extracts a line or selected lines from a composition and deletes all others.

EXTRACT (linenumbers)

A single line number, a block of line numbers, or a series of either or both can be listed after the word EXTRACT. If more than one number is listed, the numbers must be specified in increasing order.

DELETE The DELETE command deletes a line or selected lines from a composition.

DELETE (linenumbers)

A single line number, a block of line numbers, or a series of either or both can be entered after the word DELETE.

MOVE The MOVE command moves a line or block of lines, inserting them after a specified line number.

MOVE (linenumbers),[ln]

A single line number, a block of line numbers, or a series of either or both can be entered after the word MOVE. These line numbers are followed by a comma and a single line number which designates the new location. The MOVE command also resequences the file.

APPEND The APPEND command joins a specified file to the end of the composition in the current file and resequences the file.

APPEND (filenames)

One or more file names separated by commas can be entered after the word APPEND. They will be appended in the order they are listed.

JOIN The JOIN command also joins a specified file to the end of the composition in the current file but without changing line numbers.

JOIN (filenames)

One or more file names separated by commas can be entered after the word JOIN. They will be joined in the order they are listed.

LOCATE The LOCATE command locates occurrences of a given sequence of characters in an entire composition or in selected lines.

If the string of characters is followed by a comma and the word ALL, every line in the composition containing that string will be listed in 19-line segments on the terminal.

LOCATE [string],*ALL*

If selected lines are specified, only the specified lines will be searched for the string.

LOCATE [string], *(linenumbers)*

If neither the word ALL nor any line numbers are entered, the first mention of the string will be located and listed.

LOCATE [string]

CHANGE The CHANGE command changes all or selected occurences of a specified sequence of characters into a different sequence of characters.

The original sequence of characters (string1) is followed by a comma, then the desired new sequence of characters (string2), and another comma. If the word ALL is entered, every occurrence in the composition of string1 will be changed to string2.

CHANGE [string1], [string2],*ALL*

If selected lines are specified, only in those lines will string1 be changed to string2.

CHANGE [string1], [string2], *(linenumbers)*

If neither the word ALL nor any line numbers are entered, then only the *first* occurrence of string1 will be changed.

CHANGE [string1], [string2]

BUILD The BUILD command directs the system to type line numbers automatically either in a new composition or in an addition to an old one. The numbers start with 100 and increment by 10. To exit from the BUILD mode, press RETURN after a line number appears on the terminal.

USER AND SYSTEM FILES

Up to 30 user files may be stored on SCRIPT user minidiskettes, and up to 29 user files may be stored on SCRIPT user maxidiskettes. The maximum amount of text which may be stored on a SCRIPT user minidiskette is about 35,000 characters, while on a SCRIPT user maxidiskette about 195,000 characters of text may be stored.

In addition to the user text files, SCRIPT user diskettes contain "hidden" system files. Their names begin with periods (.) and are not listed by the CATALOG monitor command. These systems files contain the Synclavier® II timbre banks (file .BNKDATA), the two Synclavier® II sequences on SCRIPT User Maxidiskettes, (files .SQ0DATA and .SQ1DATA), and, on Minidiskette systems, a SCRIPT system program (file .P1-3). Users should not attempt to access these files with OLD, SAVE, and REPLACE commands.

CREATING NEW SCRIPT USER DISKETTES

To create a new SCRIPT user diskette, first format a blank diskette using the FORMAT utility program: follow the instructions in the Utility Programs Manual. Then duplicate a SCRIPT user diskette onto the newly formatted diskette with the DUPLICAT utility program: follow the instructions in the Utility Programs Manual. Since SCRIPT user diskettes are supplied by New England Digital with several demonstration compositions already stored on them for use with the examples in the SCRIPT User Guide, you will probably wish to UNSAVE these examples on one copy of your SCRIPT user diskette to use as a master disk for creating new user diskettes.

MOVING TIMBRE BANKS BETWEEN MINI AND MAXIDISKETTE SYSTEMS

Although the STORE and RECALL monitor commands may be used to move sequences between mini and maxidiskette systems, the SCRIPT monitor has no equivalent command for moving timbre banks. Thus, users with maxidiskette systems and remote minidiskette drives may wish to access directly the .BNKDATA file to transfer timbre banks between mini and maxidiskettes.

1. To move all eight timbre banks from a SCRIPT user maxidiskette or a Synclavier® II timbre maxidiskette in the right-hand drive to a SCRIPT user minidiskette or Synclavier® II timbre minidiskette in Remote Drive 0, type:

   ```
   OLD .BNKDATA
   REPLACE/R0
   ```

2. To move all eight timbre banks from a SCRIPT user minidiskette or Synclavier® II timbre minidiskette in Remote Drive 0 to a SCRIPT user maxidiskette or Synclavier® II timbre maxidiskette in the right-hand drive, type:

   ```
   OLD .BNKDATA/R0
   REPLACE
   ```

After you type the PLAY monitor command, the SCRIPT compiler will place the Synclavier® II operating system in computer memory. At this time, the system will operate identically to a system loaded from a Synclavier® II operating system diskette, with the exceptions noted below. (Assuming, of course, that the Synclavier® II and SCRIPT system diskettes contain the same release of the Synclavier® II operating system.)

During Synclavier® II operation, the eight Synclavier II timbre banks which are stored on all SCRIPT user diskettes may be accessed through use of the TIMBRE STORE/RECALL buttons on the Synclavier® II panel. In addition, on SCRIPT user maxidiskettes, two Synclavier® II sequences of 10,000 note canacity may be accessed through use of the RECORDER STORE/RECALL buttons 1 and 2 on the Synclavier® II panel.

In SCRIPT systems, all timbre and sequence store and recall operations take place during Synclavier® II operation from the right-hand disk drive; the DRIVE SELECT button is automatically "on" and its light is lit. If you wish to wish to store or recall timbres or sequences from Synclavier® II timbre or sequence diskettes, insert the appropriate diskette in the right drive. Replace the SCRIPT user diskette in the right-hand drive before terminating Synclavier® II operation.

(The SCRIPT system diskette in the left-hand drive should never be removed during SCRIPT system operation except when using the DUPLICAT program on the Utility Programs Diskette.)

During Synclavier® II operation, certain keys on the terminal keyboard may be used to terminate Synclavier® II operation and return control of the system to the SCRIPT monitor.

S KEY

S	The S key terminates Synclavier® II operation and restores the monitor with the previous current file unchanged.

USING THE REVERSE COMPILER

The SCRIPT reverse compiler can convert a Synclavier® II sequence into any of seven different computer music formats. These formats are explained in detail elsewhere.

After reverse compilation, the monitor will be restored and the converted sequence will replace the current file; the current file name will be changed by the addition of an asterisk to the end of the name.

PF1 or **ESC** and **1**	These keys convert the sequence into a computer music notelist format with absolute note starting times expressed in seconds.
PF2 or **ESC** and **2**	These keys convert the sequence into a computer music notelist format with relative note starting times expressed in seconds.
PF3 or **ESC** and **3**	reserved for future functions
PF4 or **ESC** and **4**	reserved for future functions
ESC and **5**	This key converts the sequence into a a computer music notelist format with absolute note starting times expressed in frames. Frame rate is 24 frames per second.
ESC and **6**	This key converts the sequence into a computer music notelist format with relative note starting times expressed in frames. Frame rate is 24 frames per second.
ESC and **7**	This key converts the sequence into a computer music notelist format with absolute note starting times expressed in frames. Frame rate is 30 frames per second.
ESC and **8**	This key converts the sequence into a computer music notelist format with relative note starting times expressed in frames. Frame rate is 30 frames per second.
ESC and **9**	This key converts the sequence into a computer music notelist format; the user may make three choices: 1. selection of frame rate from 1.00 to 300.00 frames per second, 2. specification of starting times, absolute or relative, or 3. specification of frames format, either hr:min:seconds:frames or :frames.

Tempo in SCRIPT can be expressed in beats per minute, seconds per beat, or frames per beat. Time in SCRIPT can be expressed in beats, seconds, or frames. Formats for expressing tempo and time are defined below.

BEAT RATE

In SCRIPT music notation, beats are converted into seconds or frame numbers according to a beat rate. You may select any rhythmic value for the beat of the composition, including fractional values. You may then equate this rhythmic value to one of three values: the number of beats per minute, the number of seconds per beat, or the number of frames per beat.

When beat rate is specified in beats per minute, the expression takes this form in a simple TEMPO statement:

[In] TEMPO [rhythmic value] = [beats per minute]

The minimum number of beats per minute is 10; the maximum is 3000.

The equal sign must be included.

When beat rate is specified by the period in seconds, the same statement takes this form:

[In] TEMPO [rhythmic value] = [value] SECOND

The minimum value for the period is .001 seconds; the maximum is 30 seconds. Periods may be expressed up to an accuracy of .005 seconds. NOTE: The equal sign and the SCRIPT word SECOND must be included in this statement.

To convert from beats per minute (bpm) to seconds (sec), use this formula:

sec = 60/bpm

When beat rate is specified by the period in frames, the same statement takes this form:

[In] TEMPO [rhythmic value] = [value] FRAMES

The minimum and maximum number of frames depends on the frame rate. Frame numbers can be expressed up to an accuracy of .01 of a frame per beat.

NOTE: The equal sign and the SCRIPT word FRAMES must be included in this statement.

In the following chapters, whenever we use the expression [beat rate], it usually means that any of the three above methods of specifying beat rate may be used. However, when independent loops are included in the composition, the beat rate must be specified by the period of the beat, in frames or seconds.

The first beat of a notelist is numbered 1, in correspondence with standard musical practice.

TIME VALUE

In the computer music format, starting time and duration can be entered in seconds or frames. If in seconds, the first second is numbered 0.0. If in frames, the first frame is numbered :0.0. Times in frames must always be preceded by a colon.

The format for frame times is as follows:

:hour:minute:second:frame

Hour, minute, and second specifications are optional.

Whenever the expression [time value] is used, it means that time may be specified in seconds or in frames.

FRAME RATE

If frames are referenced in either [beat rate] or [time value] specifications, their composition time will be based on the frame rate or number of frames per second.

Any frame rate, from 1 to 299.9 frames per second, can be specified in an FPS statement, described in the next chapter.

If no frame rate is specified, the SCRIPT system will automatically apply a frame rate of 24 frames per second.

SCRIPT TIMBRE DEFINITION

In SCRIPT, timbres are defined in a definite order. First, each partial timbre is defined. Then a complete timbre is defined by listing the partial timbres along with various statements for special effects.

PARTIAL TIMBRE DEFINITIONS

A SCRIPT partial timbre is a description of a single musical event consisting of a six-segment volume envelope, a six-segment FM envelope, a list of up to 24 harmonic coefficients (overtones), plus other information controlling vibrato, portamento, tuning, FM ratio, and decay.

The partial timbre definition begins with the words DEFINE PARTIAL followed by a partial timbre symbolic name; it ends with the word END followed by the same symbolic name. The partial timbre symbolic name will be used later in the timbre definition to indicate a partial timbre to be included in a complete timbre.

Symbolic names may be any length and may include letters, digits, and the special character "." (period). SCRIPT statement names cannot be used for symbolic names. Between the DEFINE PARTIAL and the END statements are up to eight statements that define the sound of the partial timbre. Each statement begins with a SCRIPT word followed by a number, numbers or symbolic expressions. Details and settings for each of the eight possible statements follow.

The statements can appear in any order in a partial timbre definition. In actual practice, one statement may extend across many lines, or one line may contain more than one statement.

VOLENV

The VOLENV statement establishes the volume envelope for the partial timbre.

[ln] VOLENV [value] [value] [value] [value] [value] [value]

The six values, listed in the order they appear on the Synclavier® II panel, establish the six volume envelope settings. Delay, attack, initial decay, and final decay are specified first by values from 0 to 9999 milliseconds. Peak and sustain levels are then specified by values from 0 to 100.0. The six numbers must be entered in the right order and be separated by spaces.

The default settings for VOLENV are:

VOLENV 0 10 10 10 100.0 100.0

FMENV

The FMENV statement establishes the FM envelope for the partial timbre.

[ln] FMENV [value] [value] [value] [value] [value] [value]

The values establish the six FM envelope settings. Again, the values are listed in the order they appear on the Synclavier® II panel. Delay, attack, initial decay, and final decay are each given a number from 0 to 9999 milliseconds. Peak and sustain levels are each given a number from 0 to 1000. The six numbers must be entered in the right order and be separated by spaces.

The default for FMENV is no FM, or:

FMENV 0 0 0 0 0 0

HARMONICS

The HARMONICS statement defines the wave shape for the partial timbre.

[ln] HARMONICS (values)

Up to 24 values can be specified, one for each harmonic coefficient, using numbers on a scale of 0 to 100.0. Unspecified coefficients of harmonics of higher number than the last harmonic specified will be assigned the value 0.

The default for HARMONICS is a sine wave. (The first harmonic coefficient will be assigned the value 100.0, the 23 other harmonic coefficients will be set equal to zero.)

TIMBRE DEFINITIONS

A SCRIPT timbre is a complete, complex sound defined by the simultaneous occurrence of several partial timbres. The definition starts with the words DEFINE TIMBRE followed by the timbre symbolic name. The rules for timbre symbolic names are the same as those for partial timbre symbolic names. The definition ends with the word END followed by the same symbolic name. In between, one to four partial timbres may be listed and any of the six optional statements for special effects. The *Synclavier® II Instruction Manual* explains the musical effect of each function. The default is the same for all special effects statements: If the statement is not included, the special effect will not be activated.

CHORUS The CHORUS statement activates the chorus function.

[In] CHORUS [value]

The value is a number (from 0.00 to 10.00) which establishes the desired ratio.

REPEAT The REPEAT statement activates the repeat function.

[In] REPEAT

The REPEAT statement must be followed by a RATE statement (described below).

ARPEGGIATE The ARPEGGIATE statement activates the arpeggiate function.

[In] ARPEGGIATE

The ARPEGGIATE statement must be followed by a RATE statement (described below).

RATE The RATE statement sets the rate for REPEAT and ARPEGGIATE.

[In] RATE [value]

The value is a number (from 0.00 to 50.00) in Hertz which establishes the rate for repeat and arpeggiate functions.

NUMVOICES The NUMVOICES statement determines the keyboard polyphony.

[In] NUMVOICES [value]

The value is a number (from 0.00 to 100.0) in Hertz which establishes the rate for repeat and arpeggiate functions.
simultaneous notes.

RTE The RTE statement determines the timbre modifications equivalent to the "real-time effects" patching of Synclavier® II operation. In Synclavier® II operation the timbre is controlled by the pedal; in SCRIPT, timbre modification percentages may be specified on a note-by-note basis, and by the TIMBRE and TIMBRE MULTIPLIER statements.

[In] RTE (symbols) (symbols)

The first symbols denote the partial timbre to be affected by the timbre modification percentages. Any or all of the following symbols can be listed, separated by spaces.

P1 partial timbre 1
P2 partial timbre 2
P3 partial timbre 3
P4 partial timbre 4

The second symbols establish the specific real-time affects parameters. Symbols can be selected from the following list:

VEATTACK	volume envelope attack
VEDECAYS	volume envelope decays
VEPEAK	volume envelope peak and sustain levels
HEATTACK	harmonic envelope attack
HEDECAYS	harmonic envelope decays
HEPEAK	harmonic envelope peak level
HESUSTAIN	harmonic envelope sustain level
PRATE	portamento rate

BASIC NOTELIST FORMAT

The notelist corresponds to a track of the Synclavier® II memory recorder and consists of the notes in one timbre. The notelist begins with a statement specifying the timbre to be used. This statement can take either of two forms:

1. [ln] NOTELIST USING [value] -[value]

   This form is used to specify a timbre stored in a Synclavier® II timbre bank on the user diskette. The first value is a number (from 1 to 8) which establishes the TIMBRE BANK; the second value is a number (from 1 to 8) which establishes the TIMBRE ENTRY. A hyphen must appear between the two numbers.

2. [ln] NOTELIST [symbol]

   This form is used to specify a timbre defined within the SCRIPT composition. The symbol is the timbre symbolic name.

The notelist ends with a line consisting of the word END. In between are the notes on the track.

NOTE SPECIFICATION

For each note, or rest, an entry must appear on both a pitch and a rhythm line. There must be an equal number of entries on the two lines, as in this short example.

```
100 NOTELIST USING 1-1
110   P  C  A  D
120   R  4  4  4
130 END
```

Note specification may optionally include an articulation, timbre modification, and volume line. If any of the latter three lines are included, a percentage must be specified for every note, or rest, on the pitch and rhythm lines above. The percentage given for a rest on these lines is arbitrary.

```
100 NOTELIST USING 1-1
110   P   C   A   R   D
120   R   4   4   4   4
130   A   50  75  75  100
140   V   50  70  70  100
150   T   50  0   0   0
160 END
```

An optional number of notes may be entered on each set of lines. A long sequence of pitches can be broken by repeating the pitch line, in this manner:

```
P  A  G
P  F
R  4  4  4
```

Rhythm, articulation, volume, and timbre lines can be broken similarly. The total number of values in each set of lines must match.

Each line will now be explained in detail.

The Pitch Line

P

1. The pitch line starts with the letter P followed by a list of pitches.

   [In] P (pitches)

   Each pitch is specified by a letter, A through G, followed by optional accidental and octave specification. Pitches are separated by spaces. Examples: A A3 A# A#3

F
#
N

2. Accidentals are specified by these symbols immediately following a pitch: F for flat, # for sharp, and N for natural. Accidental symbols do not affect subsequent notes. Double and multiple accidentals are allowed.

3. Each note will be played in the same octave as the previous note, unless a new octave number is entered after the pitch. The default octave for the first note in the composition is 3. (For a complete list of all octave numbers, see the figure on page 11 of the User Guide.)

R
.

4. Rests are specified by using an R or a period (".") on the pitch line. For each rest, a rhythmic value indicating the length of the rest must appear on the rhythm line.

[ ]

5. Simultaneous notes can be specified by using brackets around pitches. Any pitches so enclosed will *start* at the same time. (For example, [F A] causes F and A to start together.) As always, each pitch must have its own rhythmic value; but the pitches needn't have the same rhythmic values. If the rhythmic values are different, the first note after the right-hand bracket will follow the last note or rest within the bracket. All pitches within the same set of brackets must be on the same set of pitch lines.

{ }

6. Slurs can be specified by using braces around pitches. The pitches will change at the instant specified in the rhythm line below, but without retriggering the envelope or vibrato generators. When the timbre specified in the notelist has active portamento, a glide between notes will be produced. When an articulation line is included, a percentage must be entered for each note enclosed. The actual percentages are ignored, however, for all but the last note within the braces.

   Braces can also be placed around repetitions of the same pitch. The result can sound as a single note, but since different volume and timbre modification percentages can be specified for each note enclosed within the braces, a modulation in volume or timbre can be produced.

   All pitches within the same set of braces must be on the same set of pitch lines. Furthermore, brackets and braces cannot be used around the same pitches.

The Rhythm Line

R

1. The rhythm line starts with the letter R followed by a list of rhythmic values. There must be exactly one rhythmic value for every pitch on the pitch line.

   [ln] R (rhythmic values)

   One rhythmic value is separated from the next by a space.

/

2. Rhythmic values can be any fraction of a whole note (such as 5/8 or 11/17) as well as a fractional value greater than a whole note (such as 4/1 or 10/8).

   When a single number is specified for a rhythmic value, the number represents a fractional division of a whole note. Any number up to 64 is permissable.

   Common rhythmic values can also be written in alphabetical symbols, as follows:

Ryth. Value	Equiv. Value	Alpha. Symbol	Standard Notation
1	1/1	W	whole note
2	1/2	H	half note
4	1/4	Q	quarter note
8	1/8	E	eighth note
16	1/16	S	sixteenth note
32	1/32	T	thirty-second note
64	1/64	X	sixty-fourth note

3. Dotted notes are specified by entering a period after a rhythmic value. The period multiplies the rhythmic value by one and one half.

Ryth. Value	Equiv. Value	Alph. Symb.	Standard Notation
1.	3/2	W.	dotted whole note
2.	3/2	H.	dotted half note
4.	3/8	Q.	dotted quarter note
8.	3/16	E.	dotted eighth note
16.	3/32	S.	dotted sixteenth note
32.	3/64	T.	dotted thirty-second note
64.	3/128	X.	dotted sixty-fourth note

   Notes can be double dotted and triple dotted.

T

4. Triplet notes are specified by entering a T after the rhythmic value. The T multiplies the rhythmic value by 2/3.

Ryth. Value	Equiv. Value	Alpha. Symb.	Standard Notation
2T	1/3	HT	triplet half note
4T	1/6	QT	triplet quarter note
8T	1/12	ET	triplet eighth note
16T	1/24	ST	triplet sixteenth note
32T	1/38	TT	triplet thirty-second note
64T	1/96	XT	triplet sixty-fourth note

+

5. Rhythmic values can be added. Entering a plus sign between two or more rhythmic values creates a single tied note of combined rhythmic value. (For example, 1 + 2 creates the rhythmic value of a whole note tied to a half note.) There should be only one pitch in the line above for the combined rhythmic value and only one volume specification or timbre modification.

−

6. Rhythmic values can be subtracted. This feature can be used to create grace notes. First list the grace note rhythmic value, then the adjacent note value minus the grace note value. (For example, R 64 1−64 creates a sixty-fourth grace note preceding a whole note.)

The Articulation, Timbre and Volume Lines

All three of these optional lines function in exactly the same way. When any of them are used, a percentage must be specified for every note on the pitch and rhythm lines above.

A

1. The articulation line, used to specify the actual duration of each note, starts with an A followed by a list of articulation percentages.

   [ln] A (values)

   Each of the values is an articulation percentage (from 0.0 to 100.0) by which the rhythmic value above is multiplied to determine the actural duration of the note. (Legato is specified by an articulation percentage of 100; staccato, by a percentage of around 10.)

T

2. The timbre modification line, used to specify the percentage of real-time effects used in each note, starts with the letter T followed by a list of percentages.

   [ln] T (values)

   Each of the values is a timbre modification percentage (from 0.0 to 100.0), one for each note on the pitch and rhythm lines above.

V

3. The volume line, used to specify the volume of each note, starts with the letter V followed by a list of percentages.

   [ln] V (values)

   Each of the values is a volume percentage (from 0.0 to 100.0), one for each note on the pitch and rhythm lines above.

Parentheses

( )

Parentheses can be used to specify repeats. They can be placed around any repeating value (pitch, rhythmic value, or percentage for articulation, timbre, or volume) and followed by the number of times the value is to be repeated.

Parentheses can be used to repeat sequences as well as single notes. Long sequences of pitches can be repeated by using parentheses around pitches listed in multiple pitch lines. Long sequences of rhythmic values can be repeated in the same way.

Multiple parentheses are also allowed.

However parentheses are used, the total number of pitches and rhythmic values must match.

Some examples:

EXAMPLE	EQUIVALENT
(C3)3	C3 C3 C3
(C D)2	C D C D
(16)8	16 16 16 16 16 16 16 16
(E (S)6)2	E S S S S S S E S S S S S S

MOTIFS

Motifs can be created for repeated use and transformation within the notelists.

Motif Definition

The motif definition begins with the words DEFINE MOTIF, followed by a motif symbolic name. The motif name must begin with a letter, but it may be any length. A SCRIPT statement name may not be used for a symbolic name. The definition ends with the word END optionally followed by the same motif name. In between are the pitch, rhythm, articulation, volume and timbre modification lines that specify the notes in the motif.

All pitch lines must be grouped together, followed by all rhythm lines, and so on. A KEYSIG statement can be inserted within a motif. It will affect only the notes within the motif definition. If no key signature is specified in the motif definition, then the key signature in effect at the line number of the DEFINE statement will be used.

Basic motif format:

```
[ln] DEFINE MOTIF [symbolic name]
[ln] KEYSIG [symbol]
[ln] P (pitches)
[ln] R (rhythmic values)
[ln] A (values)
[ln] V (values)
[ln] T (values)
[ln] END [symbolic name]
```

Motif Performance

Once the motif has been defined, it may be used in a notelist by inserting the PERFORM statement.

```
[ln] PERFORM [symbolic name]
```

The word PERFORM is followed by the motif symbolic name. When the composition is played, the motif will be played in whatever key it was defined in, regardless of the current key signature.

Immediately following the PERFORM statement, articulation, volume, or timbre modification lines can be inserted. These lines can be used to specify percentages for the notes in the motif or to override such percentages if already specified in the motif definition.

Motif Transformation

A motif may be transformed by any or all of seven different operations. The PERFORM statement is inserted in the notelist as usual. After the motif name, alphabetic symbols indicating the desired operations are listed, separated by spaces, and enclosed in parentheses.

The format is:

```
[ln] PERFORM [symbolic name] ( (symbols) )
```

The symbols are:

T [pitch]	tonal transposition
ST [pitch]	strict transposition
I	tonal inversion
SI	strict inversion
R	retrograde
PR	pitch retrograde
RR	rhythm retrograde

More than one operation may be specified. The operations will be applied to the motif in the order in which they are listed. For example, an inversion operation and a retrograde operation can be listed within the parentheses, as follows:

```
100 PERFORM MOTIF1 (I R)
```

In this case, the motif will be inverted and retrograded and then performed.

(Motif transposition is different from the TRANSPOSE run-time statement which implements the same strict transpose as the TRANSPOSE button on the Synclavier® II panel. A transpose achieved through the use of the TRANSPOSE run-time statement affects all notes of a notelist and could be canceled during Synclavier® II operation by transposing back to middle C.)

The seven operations are:

T For tonal transposition, use the letter T followed by a starting pitch enclosed in parentheses.

200 PERFORM MOTIF1 (T(C3))

In this example, the first note of the transposed motif will be a middle C.

When the piece is played, the tonal transposition will make the first pitch of the motif begin exactly on the pitch in the parentheses. This starting pitch will be modified by the key signature currently in effect, but not by the key signature of the motif definition.

If the original starting pitch of the motif is not modified by an accidental and if the specified new starting pitch is a naturally occurring pitch in the motif's key signature, then the tonal transposition will be performed in the straightforward tonal manner, that is, with all the sharps and flats of the motif key signature.

If either the original starting pitch or the new starting pitch are not the natural pitches in the motif's key signature, the transpose will be carried out in the following manner: a tonal transposition will be applied using the naturally occuring pitch, then all pitches in the motif will be lowered or raised additional semitones as required to make the starting pitch match the pitch specified in the PERFORM statement.

ST For a strict transposition, use the letters ST followed by a pitch enclosed in parentheses.

When the piece is played, the motif will be transposed up or down so that it will start exactly on the pitch as specified and as modified by the current notelist key signature.

The key signature of the motif is disregarded, but the exact melodic intervals of the motif are maintained. Thus, the motif is transposed to a new key signature.

I For tonal inversion, use the letter I. The The motif will be inverted around its starting pitch.

If the starting pitch is a normally occurring pitch in the motif's key signature, it will not be changed by the inversion. If the starting pitch is modified by an accidental in the motif's key signature, it will have the opposite accidental.

All following notes will be inverted around this pitch, retaining the sharps and flats of the motif key signature. If any notes in the motif are accidentals, they will have the opposite accidental applied during inversion.

SI For strict inversion, use the letters SI. The motif will be inverted around the starting pitch of the motif without regard to the key signature of the motif. The exact intervals between the notes of the motif will be retained, but in the opposite directions.

R For retrogrades, use the letter R. The motif will be performed beginning with the last note and ending with the first, without any pitch or duration modifications.

PR For pitch only retrogrades, use the letters PR. The pitches of the motif will be played backwards, but the rhythmic values will be played as defined in the motif.

RR For rhythm only retrogrades, use the letters RR. The rhythmic values will be played backwards, but the pitches will be played as defined in the motif.

Further Notes on Transformations

1. If a strict transposition is followed by a tonal transposition in the same PERFORM statement, the tonal transposition will be performed using the *transposed* motif key signature, as opposed to the original motif key signature.

   Likewise, if a strict transposition is followed by a tonal inversion, the tonal inversion is performed using the *transposed* motif key signature.

   Special key signatures, using double sharps or double flats, are automatically created as required by the SCRIPT system.

2. Retrogrades can be performed even when there are bracketed or braced pitches in the motif. Note that if the rhythmic values for bracketed pitches are unequal, the results of a retrograde may be difficult to understand. Also note that pitch only retrogrades performed on a motif with bracketed pitches may lead to unexpected results.

INSERT STATEMENT

The INSERT statement can be used to insert files stored on diskette into a composition. It can be used to link up sections of a long composition, and to insert previously created timbre definitions, motif definitions and entire notelists. The format is:

[ln] INSERT '[filename]'

The file name is the name of the desired file. The apostrophes must be used.

"CONDUCTING" STATEMENTS

Several statements can be used to "conduct" the composition. Each one affects some aspect (pitch, rhythmic value, articulation, volume, or timbre modification) of the notes that follow, until the end of the composition or until the next occurrence of the statement. Thus, one conducting statement can affect more than one notelist.

These statements establish the conversion factors used by the SCRIPT compiler when it translates a composition in SCRIPT music notation into a Synclavier® II memory recorder sequence. If any of the "conducting" statements are omitted from a composition, the SCRIPT compiler will use the default factors as specified below.

KEYSIG

The KEYSIG statement is used to translate all succeeding notes into a given key; it works similarly to the key signature in standard notation.

[ln] KEYSIG [symbol]

The symbol is a pitch letter that establishes the desired key. Only the following legitimate Western key signatures are allowed. Minor keys are established by using the relative major key pitch.

SCRIPT KEY	MAJOR KEY	MINOR KEY	SCALE						
C	C	A	C	D	E	F	G	A	B
G	G	E	G	A	B	C	D	E	F#
D	D	B	D	E	F#	G	A	B	C#
A	A	F#	A	B	C#	D	E	F#	G#
E	E	C#	E	F#	G#	A	B	C#	D#
B	B	G#	B	C#	D#	E	F#	G#	A#
F#	F sharp	D#	F#	G#	A#	B	C#	D#	E#
C#	C sharp	A#	C#	D#	E#	F#	G#	A#	B#
F	F	D	F	G	A	BF	C	D	E
BF	B flat	G	BF	C	D	E	F	G	A
EF	E flat	C	EF	F	G	AF	BF	C	D
AF	A flat	F	AF	BF	C	DF	EF	F	G
DF	D flat	BF	DF	EF	F	GF	AF	BF	C
GF	G flat	EF	GF	AF	BF	CF	DF	EF	F
CF	C flat	AF	CF	DF	EF	FF	GF	AF	BF

Modulations can be accomplished by entering new KEYSIG statements. They will affect all succeeding notes including those in following notelists.

The KEYSIG default parameter is C.

TEMPO The TEMPO statement sets the rhythmic value of a beat and the number of beats per minute, or the tempo of a composition. As stated in Timing in SCRIPT, it can be expressed in beats per minute, seconds, or frames.

[ln] TEMPO [beat rate]

The TEMPO beat rate default parameter is 4 = 120 or 4 = 0.5 SECOND.

FPS The FPS statement sets the rate for frames per second. If used, it must appear at the beginning of the composition before any reference to frames is made.

[ln] FPS [value]

The value is a number from 1 to 299.9 indicating frames per second, with an accuracy of 0.01 frames per second.

The FPS default parameter is 24 frames per second.

ARTICULATION The ARTICULATION statement changes the SCRIPT articulation default of 100.0 percent (or legato).

[ln] ARTICULATION [value]

The value is a number (from 0.0 to 100.0) which establishes an articulation percentage. This percentage will be used when an articulation line has not been included.

ARTICULATION MULTIPLIER The ARTICULATION MULTIPLIER statement may be used to scale all articulations, both those set by an ARTICULATION statement and those specified in an articulation line.

[ln] ARTICULATION MULTIPLIER [value]

The value is a percentage, up to 1000.0, which will multiply all articulations. NOTE: This statement cannot make a note play longer than was originally established in the rhythm line.

The ARTICULATION MULTIPLIER default is 100.0.

TIMBRE The TIMBRE statement changes the SCRIPT timbre modification default percentage.

[ln] TIMBRE [value]

The value is a number (from 0.0 to 100.0) which establishes a timbre modification percentage. This percentage will be used when a timbre modification line has not been included.

TIMBRE MULTIPLIER The TIMBRE MULTIPLIER statement may be used to scale all real-time effects percentages, both those set by a TIMBRE statement and those specified in a timbre modification line.

[ln] TIMBRE MULTIPLIER [value]

The value is a percentage, up to 1000.0, which will multiply all timbre modification percentages.

The TIMBRE MULTIPLIER default parameter is 100.0.

VOLUME The VOLUME statement changes the SCRIPT volume default percentage of 100.0 percent.

[ln] VOLUME [value]

The value is a number, in the range 0.0 to 100.0, which establishes a volume percentage. This percentage will be used when a volume line has not been included.

VOLUME MULTIPLIER The VOLUME MULTIPLIER statement may be used to scale all volume percentages, both those set by a VOLUME statement and those specified on a volume line.

[ln] VOLUME MULTIPLIER [value]

The value is a percentage, up to 1000.0, which will multiply all articulations.

The VOLUME MULTIPLIER default is 100.0.

[In] TRANSPOSE [symbol]

The symbol is a SCRIPT pitch that sets the transpose distance relative to middle C. All notes in the notelist are transposed up or down that distance. Only one TRANSPOSE statement can be used in a notelist. If used, it must appear at the beginning of the notelist.

GLIDEFROM

The GLIDEFROM statement controls portamento for the first note in the sequence.

[In] GLIDEFROM [symbol]

The symbol is a pitch that sets the starting pitch from which you glide to the first note. If used, the GLIDEFROM statement must appear at the beginning of the notelist. Its effect will only be heard if there is active portamento in the selected timbre.

```
100 GLIDEFROM C3
110 P  C2  G2  C3
120 R  4   4   4
```

C2, the first note in the composition, will be preceded by a slide from C3.

The GLIDEFROM default pitch is C1.

STARTLOOP
ENDLOOP

The words STARTLOOP and ENDLOOP are used to specify the beginning and end point of a loop on an individual track, or an independent loop.

[In] STARTLOOP [value]
[In] ENDLOOP [value]

In a computer music notelist, the value must be a time value. In a SCRIPT music notelist, the value may be either a time value or the name HERE, in which case the starting time of the next note will be assigned. In either case, the STARTLOOP and ENDLOOP statements should be ordered in the notelist at the place where the loop point is to occur.

STARTLOOP HERE must immediately precede the note lines contained within the loop. To create a *Da capo* loop, insert the STARTLOOP HERE statement before the first note in the notelist. ENDLOOP HERE must appear on the line immediately after the rhythm line (or articulation, timbre modification or volume line if used) from which the sequence is to loop back.

[In] STARTLOOP HERE
[In] P (pitches)
[In] R (rhythmic values)
[In] A (values)
[In] T (values)
[In] V (values)
[In] ENDLOOP HERE

To create a *Dal segno* loop, insert STARTLOOP HERE immediately before the *Dal segno* point. Notes can be entered after ENDLOOP HERE as well. They will be played if the TRACK and LOOP buttons are pressed during performance.

[In] P (pitches)
[In] R (rhythmic values)
[In] STARTLOOP HERE
[In] P (pitches)
[In] R (rhythmic values)
[In] A (values)
[In] T (values)
[In] V (values)
[In] ENDLOOP HERE
[In] P (pitches)
[In] R (pitches)
[In] A (values)
[In] T (values)
[In] V (values)

DEFAULT SETTINGS SUMMARIZED

All SCRIPT music notation default settings, which will be used for any unspecified variables, are listed below:

OCTAVE NUMBER (after pitch)	3
OCTAVERATIO	1.000
SCALE	0 1 2 3 4 5 6 7 8 9 10 11
CLICKPERIOD	0.5
SPEED	1.000
LOOP	no loop
TRANSPOSE	no transpose
GLIDEFROM	glide from C1
KEYSIG	C
TEMPO	4=120
FPS	24 frames per second
ARTICULATION	100.0
ARTICULATION MULTIPLIER	100.0
DYNAMICS	100.0
DYNAMICS MULTIPLIER	100.0
VOLUME	100.0
VOLUME MULTIPLIER	100.0

The SCRIPT computer music notelist is a sequential list of all the notes to be played on one track in a composition.

As in a SCRIPT music notation notelist, a computer music notelist begins with a NOTELIST or NOTELIST USING statement. An END statement at the end of the notelist is optional. SCRIPT music notation and computer music notelists may be combined in the same composition.

Each note is on a separate line and is represented by a series of numbers and a pitch, as follows:

[ln] [time value] [pitch] [time value] [value] [value] [symbol]

The following order is used:

line number The line begins with a line number.

starting time The first time value on the line is the starting time of the note. It can be expressed in seconds or in frames. All starting times in a notelist must be in non-decreasing time order.

Starting times may be absolute or relative to the previous note's starting time: Absolute starting time represents the length in time from the beginning of the notelist to the beginning of each note. The maximum starting time is 29,999.99 seconds.

Relative starting time represents the length in time from the beginning of the previous note to the the beginning of the note. It is indicated by adding a plus sign before the time value.

pitch The pitch of each note is indicated by a standard pitch letter, followed by sharp or flat, if any, and octave number.

duration The second time value indicates the duration of each note in seconds or in frames. The maximum duration of a note is 30 seconds.

The remaining entries on the line are optional:

volume The first value is a volume percentage number from 0.0 to 100.0.

(RTE) The second value is a timbre modification percentage number from 0.0 to 100.0. This number must be enclosed in parentheses.

symbol The symbol S, entered after the last number on the line, will slur that note with the preceding note. To do so, the starting time plus duration of the previous note must overlap the starting time of the note.

Some examples of computer music notelist formats:

200 2.500 C3 0.25

(A note starting at 2.5 seconds, with C3 pitch, 0.25 second duration, and with default volume and timbre modification.)

200 +1.0 C4 0.50

(A note starting one second after the preceeding note, with C4 pitch, 0.5 second duration, and with default volume and timbre modification.)

200 2.500 C3 0.25 50 S

(A note starting at 2.5 second, with C3 pitch, 0.25 second duration, 50% volume, slurred to the preceeding pitch, and with default timbre modification.)

200 2.500 D3 0.25 (40)

(A note starting at 2.5 seconds, with D3 pitch, 0.25 second duration, default volume, and with 40% timbre modification.)

The following run-time and conducting statements can be used to control the computer music notelist. (The statements are explained at length in the section on SCRIPT musical notation.)

RUN-TIME STATEMENTS

All Tracks	Individual Tracks
SPEED	GLIDEFROM
CLICKPERIOD	TRANSPOSE
OCTAVERATIO	STARTLOOP
SCALE	ENDLOOP
LOOP	

CONDUCTING STATEMENTS

KEYSIG
TIMBRE
VOLUME
ARTICULATION MULTIPLIER
TIMBRE MULTIPLIER
VOLUME MULTIPLIER

The TEMPO statement is not used. Nor is the ARTICULATION statement. Although a duration must be specified for every note, the ARTICULATION MULTIPLIER may be used to alter the note durations.

The format for independent loops is as follows:

A STARTLOOP statement with the loop starting time must be inserted on the line immediately before the first note to be included in the loop. An ENDLOOP statement with the loop point time must be inserted on the line immediately after the last note in the loop.

This example illustrates the looping procedure:

```
100 NOTELIST USING 1-1
110 0.00 C3 .25
120 0.50 G3 .25
130 STARTLOOP 1.00
140 1.00 C4 .10
150 1.25 C4 .10
160 1.50 D4 .10
170 1.75 D4 .10
180 ENDLOOP 2.00
190 END
```

TEMPO CONTROL

In addition to the "conducting" TEMPO statement explained earlier in the chapter on "SCRIPT MUSICAL NOTATION", SCRIPT provides an overall tempo control for the easy entry of accelerandos and ritardandos.

While different "conducting" TEMPO statements may be used on each notelist, and new TEMPO statments may be inserted into the notelist at any point, the overall tempo statements are all entered at the start of the composition. These overall tempo statements establish tempos on all notelists in the composition. The two types of tempo control may not be combined in the same composition.

The first statement in the overall tempo control section can take the same format as a "conducting" TEMPO statement, as follows:

[ln] TEMPO [beat rate]

This statement will establish the initial tempo on all notelists. Then, a series of AT statements are entered to specify the beat numbers at which tempo change is desired. Abrupt or gradual tempo changes may be indicated.

To specify an abrupt tempo change at a beat number, any of the following statements may be used:

[ln] AT BEAT [value] ACCEL TO [beat rate]
[ln] AT BEAT [value] RITARD TO [beat rate]
[ln] AT BEAT [value] TEMPO [beat rate]

The value is the beat number. The first beat in the composition is numbered 1.

To begin a gradual tempo change at a beat number, the following statements may be used:

[ln] AT BEAT [value] ACCEL TO [beat rate] IN [value]
[ln] AT BEAT [value] LIN ACCEL TO [beat rate] IN [value]

Both these statements will produce a linear accelerando over the number of beats specified by the value after the word IN.

[ln] AT BEAT [value] RITARD TO [beat rate] IN [value]
[ln] AT BEAT [value] LIN RITARD TO [beat rate] IN [value]

Both these statements will produce a linear ritardando over the number of beats specified by the value after the word IN.

[ln] AT BEAT [value] LOG ACCEL TO [beat rate] IN [value]

This statement will produce a logarithmic accelerando over the number of beats specified by the value after the word IN.

[ln] AT BEAT [value] LOG RITARD TO [beat rate] IN [value]

This statement will produce a logarithmic ritardando over the number of beats specified by the value after the word IN.

SYNCHRONIZATION

Three SCRIPT statements can be used to convert composition or beat time into actual, elapsed time in order to synchronize a SCRIPT composition with specific time or frame cues. These statements can be used with compositions in either SCRIPT music notation or the computer music format.

Three synchronization statements are available: SYNC, DELAY, and SMOOTH. In each statement the function is first listed, followed by the composition time in seconds, frames or beats, and then the desired new actual elapsed time in seconds or frames.

The set of synchronization statements divide the composition up into time segments. A time segment is the time between the composition time value in one statement and the composition time value in the previous statement. For the first synchronization statement, the time segment is the time between the beginning of the composition and the composition time value specified in the statement. If there are more notes after the final segment, they will be played in the same tempo as in the final segment.

Up to 100 synchronization statements may be grouped together, creating 100 time segments, at the beginning of the composition before all notelists, but after the tempo AT statements. The various functions can be intermixed as desired.

All statements must be entered so that time values and beat numbers are listed in order of occurrence in time.

SYNC

The SYNC statement causes the compiler to compute a tempo adjustment for the time segment which will synchronize the composition time with the specified elapsed time at the end of the time segment.

The format for a SYNC statement is:

[ln] SYNC [time value] TO [time value]

or

[ln] SYNC BEAT [value] TO [time value]

The first time value is the composition time in seconds or in frames, the second time value is the desired elapsed time in seconds or frames.

Beat numbers can be specified instead of composition time by using the word BEAT followed by a beat number.

With a series of SYNC statements, a new tempo is computed for each segment. But, within each segment, the tempo will remain constant.

DELAY

The DELAY statement causes the compiler to delay the start of the next segment to synchronize the composition time to the actual time. A delay is added to the end of the time segment; tempo is unchanged.

The format for a DELAY statement is:

[ln] DELAY [time value] TO [time value]

or

[ln] DELAY BEAT [value] TO [time value]

Enough time must be left for the notes in each segment to be completed before the next sync point is formed.

SMOOTH

The SMOOTH statement is similar to the SYNC statement, but the tempo change at the start of the time segment will not be abrupt, but will occur gradually.

The format for a SMOOTH statement is:

[ln] SMOOTH [time value] TO [time value]

or

[ln] SMOOTH BEAT [value] TO [time value]

The first value is the composition time. The tempo of the notes on all tracks in the segment will be gradually altered to form a sync point between the composition time and the second time value, or actual time.

Each time segment that is to be "smoothed" can have at most a 2:1 tempo change. Large tempo changes should be avoided or oscillations in tempo during the time segment will be noticeable.

You have inserted the SCRIPT system diskette and pressed the LOAD button. You run into the following problems:

- THE LIGHTS ON THE SYNCLAVIER® II REMAIN ON AND THE SYNTHESIZERS ARE NOT SILENT.

Possible explanation:

Your system diskette is configured for more memory than you actually have. See "Diskette Configuration" in the Options Setup Manual.

- THE TERMINAL IS BEEPING.

Possible explanation:

There is a defect on your system diskette. Replace it.

- THE TERMINAL DOES NOT SAY "READY".

Possible explanations:

1. The terminal connector cable is not connected; the terminal is not plugged in or turned on.
2. The computer transmission speed does not match that of the terminal. See "Setting Baud on the Computer" and "Terminal Care and Setup" in the Options Setup Manual.

- THE TERMINAL IS FILLED WITH RANDOM CHARACTERS

Possible explanations:

1. The computer transmission speed does not match that of the terminal. See "Setting Baud on the Computer" and "Terminal Care and Setup" in the Options Setup Manual.
2. The system diskette is not configured for the right kind of terminal. See "Diskette Configuration" in the Options Setup Manual.

The following situations might occur during operation of the SCRIPT system:

- THE DISPLAY ON YOUR CRT SCREEN IS FROZEN.

Possible explanations:

1. You are a VT100 or VT640 user and you have pressed the NOSCROLL key or CRTL-S by mistake. Press SET-UP and 0 to clear the terminal memory.
2. Your terminal connector cable has become disconnected.

- THE TERMINAL STARTS BEEPING

Explanation:

If this occurs during operation, there is probably a defect in your user diskette. Replace it.

- YOU HAVE RUN OUT OF SPACE ON THE USER DISKETTE

1. UNSAVE obsolete files.
2. Use the SHUFFLE program. See the Utility Programs Instructions.
3. Use a new SCRIPT user diskette.

- YOUR SEQUENCE IS TOO LONG TO BE CONVERTED TO A TEXT FILE

Explanation:

Notes are stored more efficiently in the Synclavier® II operating system than in text files. A maximum length sequence in a 10,000 note Synclavier® II system will be too long to convert directly into text. To convert it, follow these instructions:

a. First store your sequence on a Synclavier® II sequence diskette.

b. Next, using the ERASE button on the Synclavier® II, erase some of the tracks in the memory recorder.

c. Try conversion again. If not successful, erase more tracks. If successful, SAVE the converted, nonerased notelists.

d. Return to the Synclavier® II, recall the sequence, and erase those tracks that were converted, and convert those tracks that were erased.

e. When all tracks are converted and saved, create a composition linking all the saved files with INSERT statements.

SCRIPT INDEX

Each SCRIPT word or monitor command may be spelled out fully and exactly as below or abbreviated to its first three letters.

SCRIPT MONITOR COMMANDS

SCRIPT STATEMENTS USED FOR TIMBRE DEFINITION

Statement	Abbreviation	Page
VOLENV	VOL	11
FMENV	FME	11
HARMONICS	HAR	11
TUNING	TUN	12
VIBRATO	VIB	12
PORTAMENTO	POR	12
FMRATIO	FMR	12
DECAY	DEC	12
CHORUS	CHO	13
REPEAT	REP	13
ARPEGGIATE	ARP	13
RATE	RAT	13
NUMVOICES	NUM	13
RTE	RTE	13
DEFINE	DEF	11, 13
PARTIAL	PAR	11
TIMBRE	TIM	13
END	END	11, 13

ARGUMENTS USED IN TIMBRE DEFINITION

Statement	Abbreviation	Page
(For Vibrato Statements)		
SINE	SIN	12
TRIANGLE	TRI	12
RAMP	RAM	12
INVERTED	INV	12
SQUARE	SQU	12
SINE1	SIN1	12
TRIANGLE2	TRI2	12
RAMP2	RAM2	12
INVERTED2	INV2	12
SQUARE2	SQU2	12
(For Portamento Statements)		
LOG	LOG	12
LIN	LIN	12
(For RTE Statements)		
VEATTACK	VEA	13
VEDECAYS	VED	13
VEPEAK	VEP	13
HEATTACK	HET	13
HEDECAYS	HED	13
HEPEAK	HEP	13
HESUSTAIN	HES	13
PRATE	PRA	13

SCRIPT MUSIC NOTATION STATEMENTS

Statement	Abbreviation	Page
NOTELIST	NOT	14
NOTELIST USING	NOT USI	14
END	END	14
P	P	15
R	R	16
A	A	17
T	T	17
V	V	17
DEFINE MOTIF	DEF MOT	18
PERFORM	PER	18
OCTAVERATIO	OCT	22
SCALE	SCA	22
SPEED	SPE	22
CLICKPERIOD	CLI	22
LOOP	LOO	22
TRANSPOSE	TRA	22
GLIDEFROM	GLI	23
STARTLOOP	STA	23
ENDLOOP	ENDLOOP	23
KEYSIG	KEY	20
TEMPO	TEM	21
FPS	FPS	21
ARTICULATION	ART	21
ARTICULATION MULTIPLIER	ART MUL	21
TIMBRE	TIM	21
TIMBRE MULTIPLIER	TIM MUL	21
VOLUME	VOL	21
VOLUME MULTIPLIER	VOL MUL	21
INSERT	INS	20
AT BEAT	AT BEA	27
ACCEL TO	ACC TO	27
RITARD TO	RIT TO	27
SYNC	SYN	28
DELAY	DEL	29
SMOOTH	SMO	29

(NOTE: ENDLOOP cannot be abbreviated END.)

ARGUMENTS USED IN MUSIC NOTATION

Statement	Abbreviation	Page
(Loops)		
HERE	HER	23
(Perform)		
T	T	19
ST	ST	19
I	I	19
SI	SI	19
R	R	19
PR	PR	19
RR	RR	19
(Accel, Ritard)		
LIN	LIN	27
LOG	LOG	27
IN	IN	27
(Sync, Delay, Smooth)		
BEAT	BEA	28, 29
TO	TO	28, 29