ZETA THREE

INSTRUCTION MANUAL

JANUARY, 1990

ADAMS-SMITH

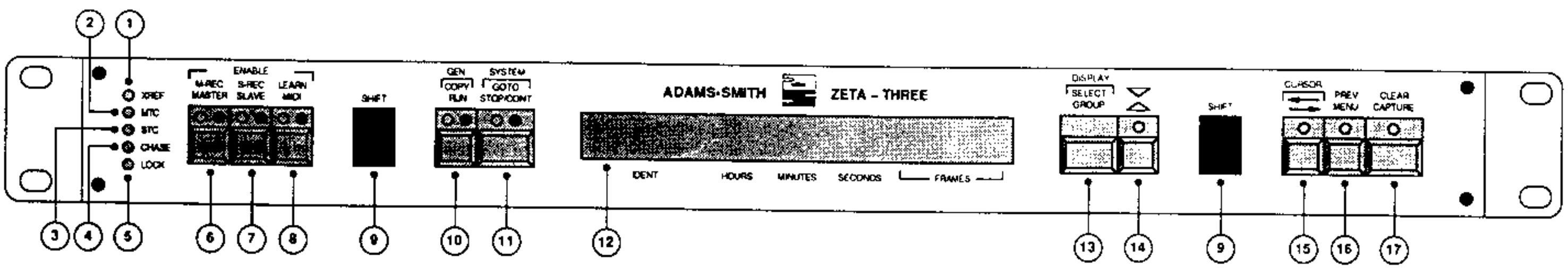
34 Tower Street

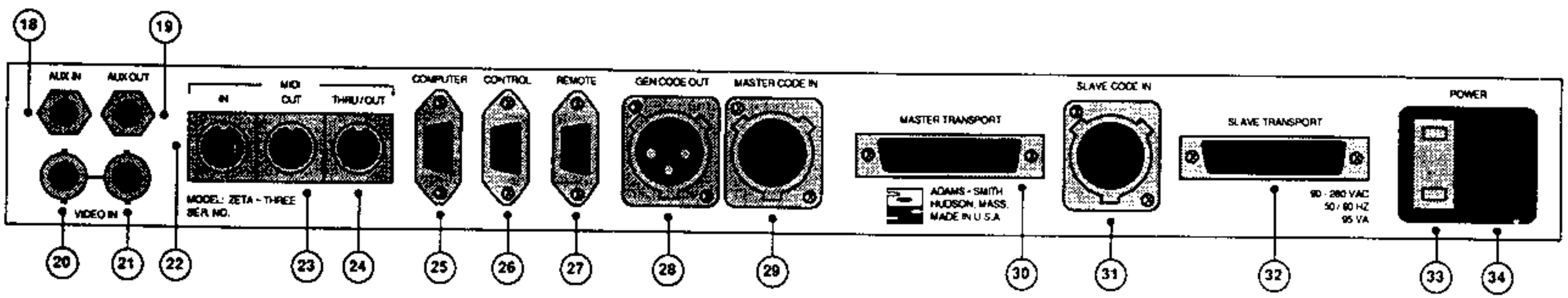
Hudson, MA 01749 USA

Tel.: (508) 562-3801

Telex: 920087

FAX: (508) 568-0404


XREF
MTC
STC
CHASE
LOCK
ENABLE
M-REC MASTER
S-REC SLAVE
LEARN MIDI
SHIFT
GEN
COPY RUN
SYSTEM
GOTO STOP/CONT
ADAMS·SMITH
ZETA - THREE
IDENT
HOURS
MINUTES
SECONDS
FRAMES
DISPLAY
SELECT GROUP
SHIFT
CURSOR
PREV MENU
CLEAR CAPTURE


AUX IN
AUX OUT
VIDEO IN
MIDI
IN
OUT
THRU/OUT
MODEL: ZETA - THREE
SER. NO.
COMPUTER
CONTROL
REMOTE
GEN CODE OUT
MASTER CODE IN
MASTER TRANSPORT
ADAMS - SMITH
HUDSON, MASS.
MADE IN U.S.A.
SLAVE CODE IN
SLAVE TRANSPORT
90 - 260 VAC
50 / 60 HZ
95 VA
POWER

Display Menu List

ADAMS·SMITH ZETA~THREE

Quick Reference Card – Software Version 3.50

Groups	Selections	Menu Items
1) Generator (G)	G_TC 01 00 00 00 G_UB 00 00 00 00	G01 PRESET 01000000 G02 COPY MODE = XFER/JAM G03 TC COPY = TC/UB/ZETATIME G04 UB COPY = UB/TC/OFF G05 CONSTANTS ->
2) Master (M)	M_TC 00 00 00 00 M_UB 00 00 00 00	M01 PRESET -------- M02 MASTER = READER/GENERATOR M03 OUTPUT = TRANSPORT/OFF/AUX3~10 M04 LIMIT -------- M05 TRANSPORT ->* M06 SAVE TRANSPORT->* M07 CONSTANTS ->
3) Slave (S)	S_TC 00 00 00 00 S_UB 00 00 00 00 S_OFS 00 00 00 00.00 S_ERR 00 00 00 00.00 S_SLEW 00.00	S01 PRESET -------- S02 LOCK MODE = ADR/FWL/AUTO S03 SLOW RELOCK = OFF/ON S04 SPLICE TRAP = OFF/ON S05 LIMIT -------- S06 TRANSPORT ->* S07 SAVE TRANSPORT->* S08 CONSTANTS ->
4) MIDI (D)	D_B120.00 0001/01.1 D_TC 00 00 00 00 D_OFS 00 00 00 00.00 D_ERR 00 00 00 00.00 D_SLEW 00.00	D01 EDIT ->* D02 SONG SETUP ->* D03 LEARN MODE ->* D04 LEARN QNTIZE = 8 D05 FPB FRM = 24 D06 MERGE = OFF/REAL-TIME/NO REAL-TIME/ALL D07 MIDI CLK = ON/ON+SSEL/DT LOCK/OFF D08 MIDI THRU = IN/OUT-2 D09 MIDI TC = OFF/ZETATIME D10 LOCK MODE = ADR/FWL/AUTO D11 SLOW RELOCK = OFF/ON D12 SPLICE TRAP = OFF/ON D13 TIMEBASE ->* D14 MAP LOAD/SAVE ->* D15 BEEP = OFF/RMT/RMT COUNT/MIDI/MIDI COUNT/RMT+MIDI/RMT+MIDI CT D16 MIDI CONSTS ->*
5) System (Z)	Z_GO -- -- -- -- Z_IN -- -- -- --.-- Z_OUT -- -- -- --.-- Z_END -- -- -- --	Z01 IN/OUT = RECORD/REHEARSE Z02 AUTO EDIT = OFF/ON/ON+CUE Z03 PREROLL = 00 00 Z04 LOOP = OFF/CYCLE/AUTO REW/AUTO STOP Z05 FRAMES = 30.00/24/25/29.97/29.97 DF Z06 RESOLVE = OFF/VIDEO/AUX IN Z07 ZETATIME = MASTER/SLAVE Z08 COMPUTER PORT ->* Z09 RS-232 BAUD ->* Z10 RS-232 FORMAT ->* Z11 CONTROL PORT ->* Z12 SYS ADDR = 0/8282 Z13 XOUT TIP = AUX 1/TIMBASE/OFF Z14 XOUT RING = AUX 2/OFF Z15 LOCAL CHASE = ON/OFF Z16 RMT REC'D = SYSTEM/SLAVE Z17 RMT Z-ECHO = OFF/ON Z18 ZETA TC LINK = OFF/ON Z19 CLEAR REGISTERS Z20 SYSTEM RESET ->*
6) Event (E)	E_01 -- -- -- --.-- E_02 -- -- -- --.-- E_03 -- -- -- --.-- E_04 -- -- -- --.-- E_05 -- -- -- --.-- E_06 -- -- -- --.-- E_07 -- -- -- --.-- E_08 -- -- -- --.-- E_09 -- -- -- --.-- E_10 -- -- -- --.--	E01 EV_01 = DISARMED/AUX OUT/MIDI NOTE/MIDI PROG/REMOTE FN . . . . . . E10 EV_10 = DISARMED/AUX OUT/MIDI NOTE/MIDI PROG/REMOTE FN E11 EV_ALL DISARM E12 MIDI TRIGGERS ->* E13 EVENT CONSTANTS ->*

Basic Operation

ADAMS•SMITH ZETA~THREE

Quick Reference Card – Software Version 3.50

Entering Data Into a Group Selection

DISPLAY GROUP	CURSOR	INDEX
X_XX	(to number to be changed)	(to change number)

Synchronize Master Slave MIDI

ENABLE MASTER	ENABLE SLAVE	ENABLE MIDI

Transport Start / Stop

ENABLE TRANSPORT(S)	SYSTEM STOP / CONT

Generator Start / Stop

DISPLAY GROUP	SHIFT SELECT	(enter generator starting address)	GEN RUN
G_XX	G_TC		

Locating to GOTO Point

(enter or CAPTURE GOTO point)	SHIFT GOTO

Capture Slave Offset

DISPLAY GROUP	SHIFT SELECT	CAPTURE
S_XX	S_OFS	offset = STC – MTC

Pressing DISPLAY GROUP changes the display to the next group.

Pressing DISPLAY SHIFT SELECT moves to the next selection in the group.

Pressing MENU shows the current group menu.

Pressing PREV (SHIFT MENU) moves the display to the previously accessed menu level.

Regenerate Time Code

Connect old code into master reader
Select XFER or JAM in GEN COPY menu
Press SHIFT COPY
Start old code source
Generator outputs regenerated code

Set GOTO Point

DISPLAY GROUP	SHIFT SELECT	CURSOR	INDEX
Z_XX	Z_GO	(to number to be changed)	(to change number)

Capturing a GOTO Point

(move tape to desired GOTO point)	DISPLAY GROUP	SHIFT SELECT	CAPTURE
	Z_XX	Z_GO	captures current master time code

Entering an END Point

DISPLAY GROUP	SHIFT SELECT	CURSOR	INDEX
Z_XX	Z_END	(to number to be changed)	(to change number)

Set Slave Lock Mode

DISPLAY GROUP	MENU	INDEX	CURSOR	INDEX
S_XX	SXX	S02 lock mode = XX	(move cursor to right)	(choose lock mode)

Setting CYCLE Function

(enter GOTO and END points)	DISPLAY GROUP	MENU	INDEX	CURSOR	INDEX
	Z_XX	ZXX	Z04	(move cursor to right)	(loop = cycle)

Slewing Slave Offset

DISPLAY GROUP	SHIFT SELECT	CURSOR	INDEX	CAPTURE
S_XX	S_SLEW	(select slew rate)	advance/retard (slave xport)	or CLEAR

Set System Frame Rate

DISPLAY GROUP	MENU	INDEX	CURSOR	INDEX
Z_XX	ZXX	Z05 frames = XX	(move cursor to right)	(choose frame rate)

Select Master (or Slave) Transport

	DISPLAY GROUP	MENU	INDEX	MENU	INDEX	CAPTURE
Master:	M_XX	MXX	M05 TRANSPORT ->	.XX TRANSPORT	(to name of master xport)	CONSTS LOADED
Slave:	S_XX	SXX	S06 TRANSPORT ->	.XX TRANSPORT	(to name of slave xport)	CONSTS LOADED

Resolving

DISPLAY GROUP	MENU	INDEX	CURSOR	INDEX		
Z_XX	ZXX	Z06 resolve = XX	(move cursor to right)	(choose EXT SYNC source) VIDEO / AUX / OFF	connect external reference to VIDEO IN or AUX IN	amber LED will be on solid if locked to XREF

Setting an Event

DISPLAY GROUP	SHIFT SELECT	enter or CAPTURE desired time code address or event	DISPLAY GROUP	MENU	INDEX	CURSOR then INDEX	CURSOR then INDEX
E_XX	E_01 (tip) or E_02 (ring)		E_XX	EXX	E01	EV_01 (to arm event 1) EV_02 (to arm event 2)	(choose to arm or disarm event listed on display)

Automate Record Function (Punch In / Punch Out)

enter or CAPTURE record start address (punch in)	enter or CAPTURE record stop address (punch out)	MENU	INDEX	INDEX	ENABLE M-REC and/or S-REC
Z_IN	Z_OUT	ZXX	Z01 in/out = XX	(choose RECORD or REHEARSE function)	(to arm master and/or slave record function)

1 INTRODUCTION - Getting Started

1.1 The ZETA-THREE Manual

The ZETA-THREE is easy to use, but it is capable of performing some very diverse and sophisticated synchronizing and MIDI tasks. This manual includes three chapters which describe how to basically use the ZETA-THREE. Other chapters in the manual provide additional information for reference and for more complex operations.

IMPORTANT - Unless you are **very** familiar with time code and synchronization operations, **please** read Appendices A and B. These Appendices provide background information which is **needed** to use this manual. The appendices will not only help you understand time code and synchronizing in general, but will also inform you about ZETA-THREE functions and routines which will help you in various synchronizing situations.

1.2 Finding the Information You Need

Chapter 2, **Installing Your ZETA-THREE**, provides instructions for installing the ZETA-THREE unit. This chapter also includes information about providing power to the unit, i.e. power cord attachment, power requirements and limitations, and fusing.

Chapter 3, **Getting Acquainted with Your ZETA-THREE**, provides instructions to help you become familiar with the basic operation of the ZETA-THREE. This chapter also includes information about locating Groups and Menus and how to enter or change information in Group and Menu items.

Chapter 4, **Basic ZETA-THREE Operations**, presents step-by-step procedures for generating, recording, and reading time code and how to synchronize two transports. This chapter also describes the cable connections needed for all the procedures described. The information in this chapter, along with the information found in Chapter 3, explains the general use of each of the front panel keys.

Chapter 5, **General Operations**, assumes that the user has a working knowledge of the ZETA-THREE and describes its most commonly used functions. Chapter 5 also provides common set-up schemes for various synchronizing operations.

Chapter 8, **MIDI Section**, describes all MIDI operations in detail. Functions include syncronizing a sequencer to time code via the ZETA-THREE's internal Tempo Map, and entering, learning and editing the Tempo Map.

Chapter 13, **Computer Port [Adams-Smith Protocols]**, will be needed by anyone writing software to communicate serially with the ZETA-THREE. The chapter contains general descriptions of the internal software operations and hardware connections, plus a complete list of communication protocols.

Appendix A, **Time Code**, provides information about what SMPTE/EBU time code is, and how time code is used by synchronizers such as the ZETA-THREE.

Appendix B, **Synchronization**, provides valuable general information about the synchronizing process.

Appendix C, **Cable and Back Panel Information**, includes power information and pin connections for each connector found on the ZETA-THREE's rear panel.

Appendix D, **ZETA-THREE Menus**, describes each of the items in the ZETA-THREE's Menus. Each possible setting of each item, and its effect on the ZETA-THREE, is also described.

Appendix E, **Slave Transport Constants**, lists and describes internal parameters which are used to adapt the ZETA-THREE slave syncronizer section to specific tape transports. These are loaded automatically when a transport is selected, but may be further manipulated by the user if so desired.
The **Master Transport** section uses a subset of these parameters.

Appendix F, **MIDI Section Constants**, describes similar operating parameters for the MIDI section.

Appendix G, **Generator Constants**, contains parameters for control of the Time Code generator.

Appendix H, **Events Constants**, lists parameters to control the finer points of Event triggering.

Chapter 2 - Installing Your ZETA-THREE

About this Chapter...

The ZETA-THREE unit requires no internal assembly. This chapter provides mounting and power-up instructions for the unit.

This chapter, along with chapters 3 and 4, includes numbers in brackets following the name of a front panel key or rear panel connector (i.e. GEN RUN [10]). The numbers refer to the location of the key or connector on the diagram provided at the very beginning of this manual. The diagram and associated numbers can help you locate a specific key or connector easily.

2.1 Mounting

The ZETA-THREE unit is designed to operate in a studio environment and should not be subjected to extremes of heat, humidity, dust, shock or vibration.

The ZETA-THREE is shipped ready for rack-mounting. The mounting angles can be removed if table-top placement of the unit is desired. The positioning of these angles is shown in Figure 2.1. The ZETA-THREE may be rack- or cabinet-mounted using only the Rack Mounting Angles.

If the ZETA-THREE is to be mounted on a desk or table, or on top of another piece of equipment, Rubber Feet should be installed on all four corners. The Feet available with the ZETA-THREE are of the self-adhering type; no holes need to be drilled in the bottom of the ZETA-THREE.

CAUTION

DO NOT, under any circumstances, drill into the ZETA-THREE's casing.

An optional slide kit may be purchased for the ZETA-THREE from your ZETA-THREE dealer. The slides will allow the ZETA-THREE to slide freely into and out of the rack or cabinet. Threaded inserts have been provided on the sides of the ZETA-THREE for the purpose of mounting the slides. Installation of the slide kit requires that one rack space be provided both above and below the ZETA-THREE.

CAUTION

Do not mount the instrument over or under equipment which generates an unusual amount of heat.

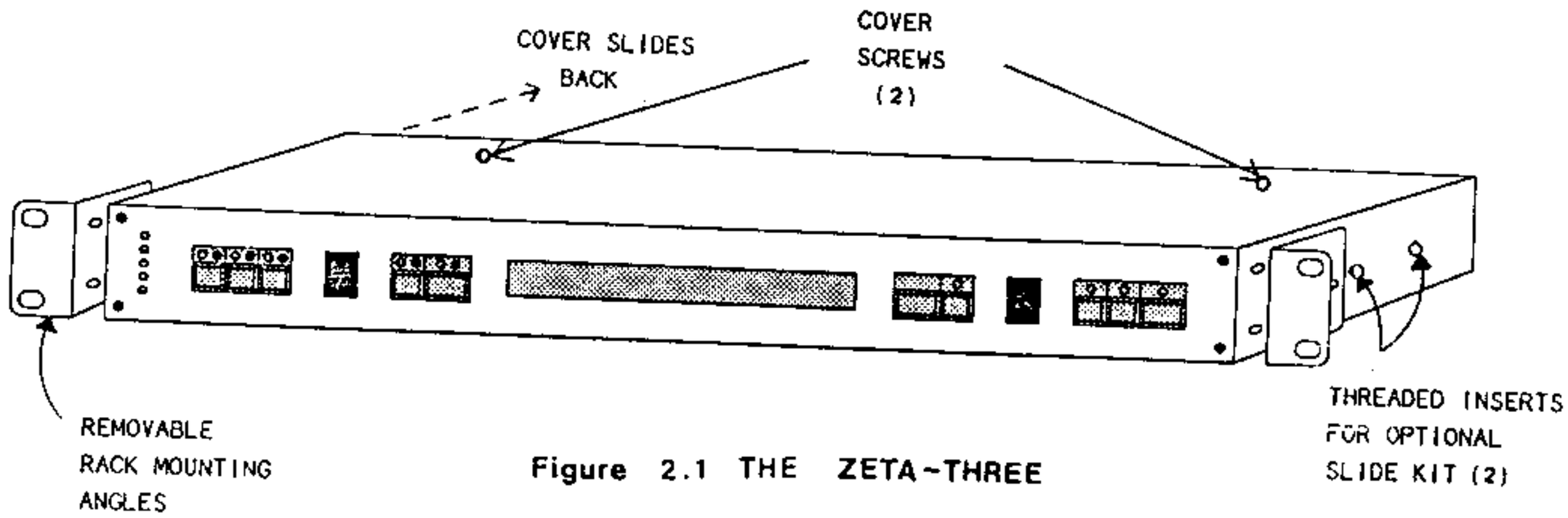


Figure 2.1 THE ZETA-THREE

2.2 Power

2.2.1 Power Requirements

THE ZETA-THREE WILL OPERATE FROM ANY POWER SOURCE WITHIN THE RANGE OF 90 TO 260 VOLTS, 95 VA, 50-60 Hz AC. For example, NO adjustments are necessary when switching from 100/120V to 220/240V.

To achieve optimum reliable operation, the ZETA-THREE should be connected to power mains which provide noise-free current (i.e. not the same circuit as other machinery).

2.2.2 Power Connections

The ZETA-THREE is supplied with a 3-wire power cord. One end of the power cord is to be plugged into the MAINS INPUT [34] on the back panel of the ZETA-THREE, as shown in Figure 2.2. The other end of the power cord is supplied with a standard USA-type 3-pin, which may be plugged into any 3-wire, <u>properly grounded</u> power outlet of sufficient capacity. If the USA-type plug is not appropriate (i.e. for European use), it may be replaced with an appropriate connector.

Power cords should be routed so that they are not likely to be walked on or pinched by items placed upon or against them.

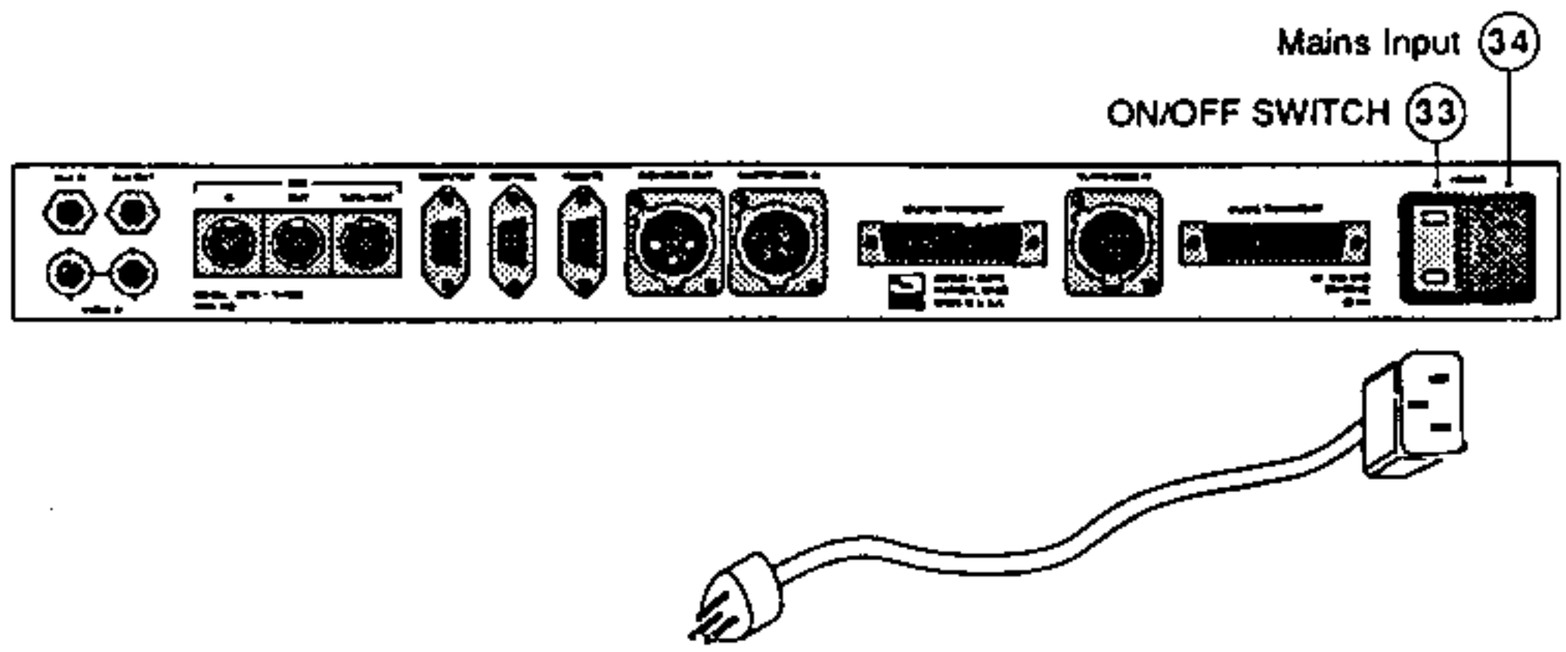


Figure 2.2 Attachment of power cord

After the ZETA-THREE unit is plugged in, power to the unit may be turned ON and OFF by operating the ON/OFF switch [33] on the back panel.

The ZETA-THREE is supplied with a 2A, 250V fuse, 5mm wide X 20mm long. The fuse is found inside the unit. To replace the fuse, unscrew the two screws on the top of the unit, as pictured in Figure 2.1, and slide the cover back.

WARNING
Always be sure that the unit is unplugged whenever the cover is to be opened.

Looking down into the ZETA-THREE from the front, the fuse can be seen in the left, rear quadrant of the unit. **BE CERTAIN THE UNIT IS UNPLUGGED BEFORE ATTEMPTING TO REPLACE THE FUSE.** Replace the fuse with one of equivalent value and slide the cover back to the unit. Replace both top cover screws.

2.3 Rear Panel Connectors

Each connector on the rear panel has specific uses. Please refer to Appendix C for specific information concerning the ZETA-THREE's rear panel connectors.

Chapter 3 - Getting Acquainted with Your ZETA-THREE

This chapter provides step-by-step procedures to help you become familiar with the front panel of the ZETA-THREE, and explains how to read the display and how to access the ZETA-THREE's data displays and Menu items. Follow the steps described in each section to learn what the ZETA-THREE does when the keys are pressed.

When following this chapter, feel free to practice using the front panel keys. Don't worry about making mistakes; the ZETA-THREE will not be damaged if the wrong key is pressed.

3.1 The Display

When the ZETA-THREE's power is turned on, the front panel display appears as shown in Figure 3.1. This type of display is called a "Group". A Group display contains information relevant to that Group. The ZETA-THREE contains six Groups.

In each Group, the character on the left, in the IDENT section of the display, IDENTifies the Group. The IDENT characters are mnemonics of the actual names of the Groups: G for Generator, M for Master transport, S for Slave transport, D for MIDI, Z for ZETA-THREE system, E for Events.

The IDENT characters immediately to the right of the Group mnemonic IDENTify the display SELECTed. For instance, the letters G_TC identify the display as "Generator Time Code".

The characters to the right of the Group's IDENTity display information relevant to the current Group and SELECTion. In the case of the Generator Time Code display, the contents on the right represent the time code number that the generator is currently generating.

The location of the cursor is indicated by the blinking character (in Figure 3.1, the left hand digit in the "HOURS" section of the display). The cursor may be moved across the display. Its location defines which character on the display may be changed.

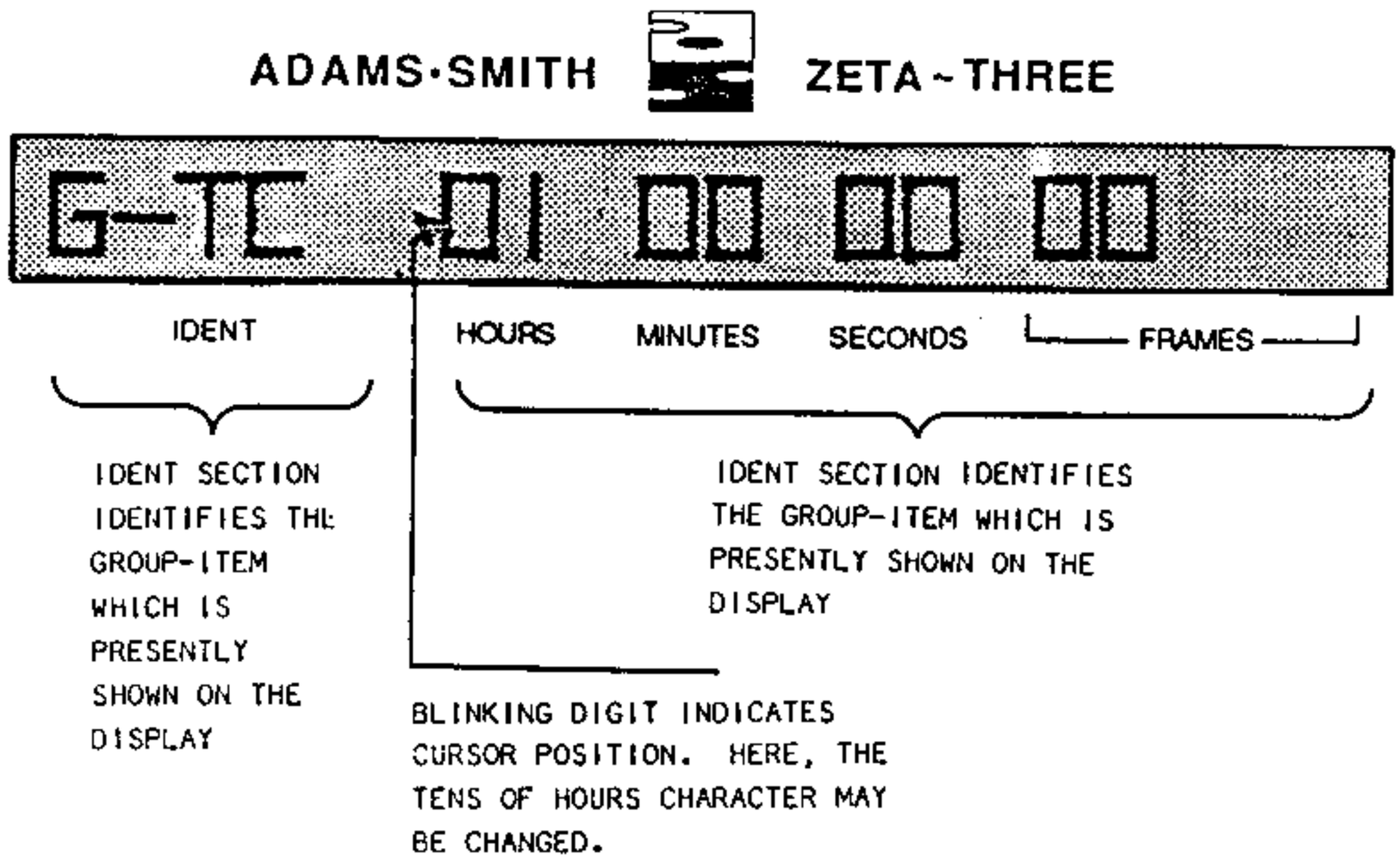


Figure 3.1 Power-up Display

3.2 The Front Panel Keys

The keys on the front panel are used to accomplish a variety of functions. This manual describes all of the ZETA-THREE functions that may be accessed through the front panel display and keys.

Figure 3.2 shows how the front panel key labels should be interpreted.

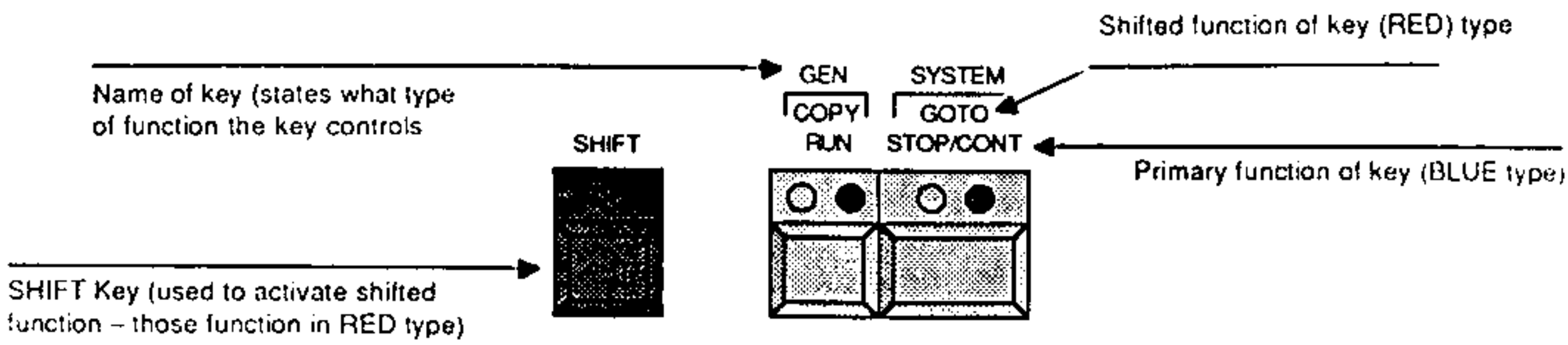


Figure 3.2 Front panel labels

Most of the keys support two functions. The primary function of a key is labeled above the key in blue and the SHIFTed (secondary) function is labeled in red. When the SHIFT key [9] is pressed, its LED lights. While that red LED is on, the key containing the desired SHIFTed function can be pressed to implement the SHIFTed function.

In this manual, the un-SHIFTed functions of a key will be written as, for example, GEN RUN. The shifted function of a key will be written as, for example, GEN COPY [SHIFT GEN RUN], indicating that both the SHIFT key (key number "9" on the diagram at the beginning of the manual) and the GEN RUN key need to be pressed to activate the function.

3.3 Using the SHIFT Key

The SHIFT key may be used one of two ways:

1) Press the SHIFT key once to light its red LED. Then press any other key, and the SHIFT key's LED will turn off. If the SHIFT key is pressed inadvertantly, pressing the SHIFT key a second time will "unshift" the key, and the red LED on the SHIFT key will turn off.

2) Press and hold down the SHIFT key (its red LED remains On). Then, while holding the SHIFT key down, press the key which contains the desired function. This procedure is useful for repeated SHIFTed keystrokes because, as long as the SHIFT key is held down, the SHIFT function remains activated.

NOTE

The SHIFT key on the left side of the ZETA-THREE's front panel affects only the keys on the left side of the panel, while the SHIFT key on the right side of the panel affects only the keys on the right side of the ZETA-THREE panel.

3.4 Entering and Modifying Data (Using the Cursor)

Data in the ZETA-THREE's Selection and Menu displays is entered or modified by using the CURSOR -> and CURSOR <- keys to place a visual indicator (called a Cursor) at the character (or word) to be entered or modified. Once the cursor is in the correct position, the /\ (Index up) or \/

(Index down) key is used to enter or modify the character (or word).

For example, suppose it is desired to have the Time Code Generator begin counting from a particular time code number. This can be accomplished by entering the starting (initial) time code number into the generator time code.

To enter a particular number into the G_TC selection, the CURSOR [15] and INDEX [14] keys are used. The cursor is located at the blinking number in the display. It may be moved across the display to the number that you wish to change by pressing the CURSOR -> key (to move the cursor to the right) or the CURSOR <- key [SHIFT CURSOR ->] (to move the cursor to the left).

Once the cursor is positioned at the number which is to be changed, press the /\ key [14] to index the number up (i.e. 1,2,3,4,...) or press the \/ key [SHIFT /\] to index the number down (i.e. 0,9,8,7,...). All of the numbers in the item can be changed in this manner.

Some items may sometimes require a negative entry. Negative numbers are entered by moving the cursor to the first (left-most) number in the item's contents and indexing the number up or down to zero. Pressing the INDEX \/ key will cause a "-" sign to then appear to the left of the number. Continuing to press the INDEX \/ key will cause the number to count backwards to -1, -2, etc.. The other numbers in the item may then be indexed up or down in any order. As long as the "-" is at the left, the number is recognized as a negative number.

When told to Run, the generator will begin counting from the number in the G_TC selection. The cursor disappears (blinking stops) when the generator is running.

NOTE

> The number entered into an item must be a valid number for the type of function the item represents. In the case of the G_TC Group selection, which contains a time code address, the values entered must always represent a valid time code number. Values outside the time code range, i.e. 69 minutes, cannot be entered or displayed.

The same method which is used to change the value in the G_TC Group selection may be used to change the number in

any display. The method for gaining access to the other Groups and selections is described in Section 3.5.

Some Group selections (namely those in the System and Events Groups) contain a set of dashes instead of numbers. The dashes mean that there is no entry in the item or that the item has been cleared. Numbers can be entered, using the CURSOR and INDEX keys, as they were entered into the G_TC display, but, if it is desireable to clear the item, the CLEAR key [SHIFT CAPTURE] must be pressed. It is not acceptable to enter a value of 00:00:00:00 into the item to clear it, as the ZETA-THREE will interpret an entry of 00:00:00:00 as a valid time code address instead of "no entry".

3.5 Gaining Access to Other Groups and Selections

In Figure 3.1 a Group Selection is shown. The name, or the title, of the Group is given in the IDENT section of the display. The first character, "G", tells us that this display is associated with the Generator.

The DISPLAY SELECT key allows you to SELECT the next item in the Generator Group. When pressing the DISPLAY SELECT key, the ZETA-THREE will only access Selections within the present Group. In Figure 3.1, the display shows the TIME CODE Selection of the GENERATOR Group.

When the Generator Time Code (G_TC) display shown on the display, press the DISPLAY SELECT key once [SHIFT DISPLAY GROUP] to change the display to the Generator User Bits (G_UB) Selection. Pressing the key again will toggle back to G_TC.

Each of the six Groups -- Generator (G), Master (M), Slave (S), MIDI (D), System (Z), and Event (E) -- contains two or more Selections which are related to the Group name. For example, the Generator Group contains two Selections which were shown to be Generator Time Code (G_TC) and Generator User Bits (G_UB). The Slave Group, on the other hand, contains five Selections: S_TC (Slave Time Code), S_UB (Slave User Bits), S_OFS (Slave OFSet), S_ERR (Slave ERRor) and S_SLEW (Slave SLEW).

To gain access to the other Groups, press the DISPLAY GROUP key. The display will sequence to the next Group each time the DISPLAY GROUP key is pressed. After all six Groups in the list have been displayed, pressing the DISPLAY GROUP key again will toggle the display to the first Group (Generator). The list of Groups, and the Selections found in each Group, is shown in Figure 3.3.

The Selections in any Group may be accessed by pressing the DISPLAY SELECT key [SHIFT DISPLAY GROUP], as was described in the Generator Group example. By pressing the DISPLAY SELECT key repeatedly, the ZETA-THREE will toggle through the list of Selections and, after all the items in the Group have been displayed, return to the top of the list.

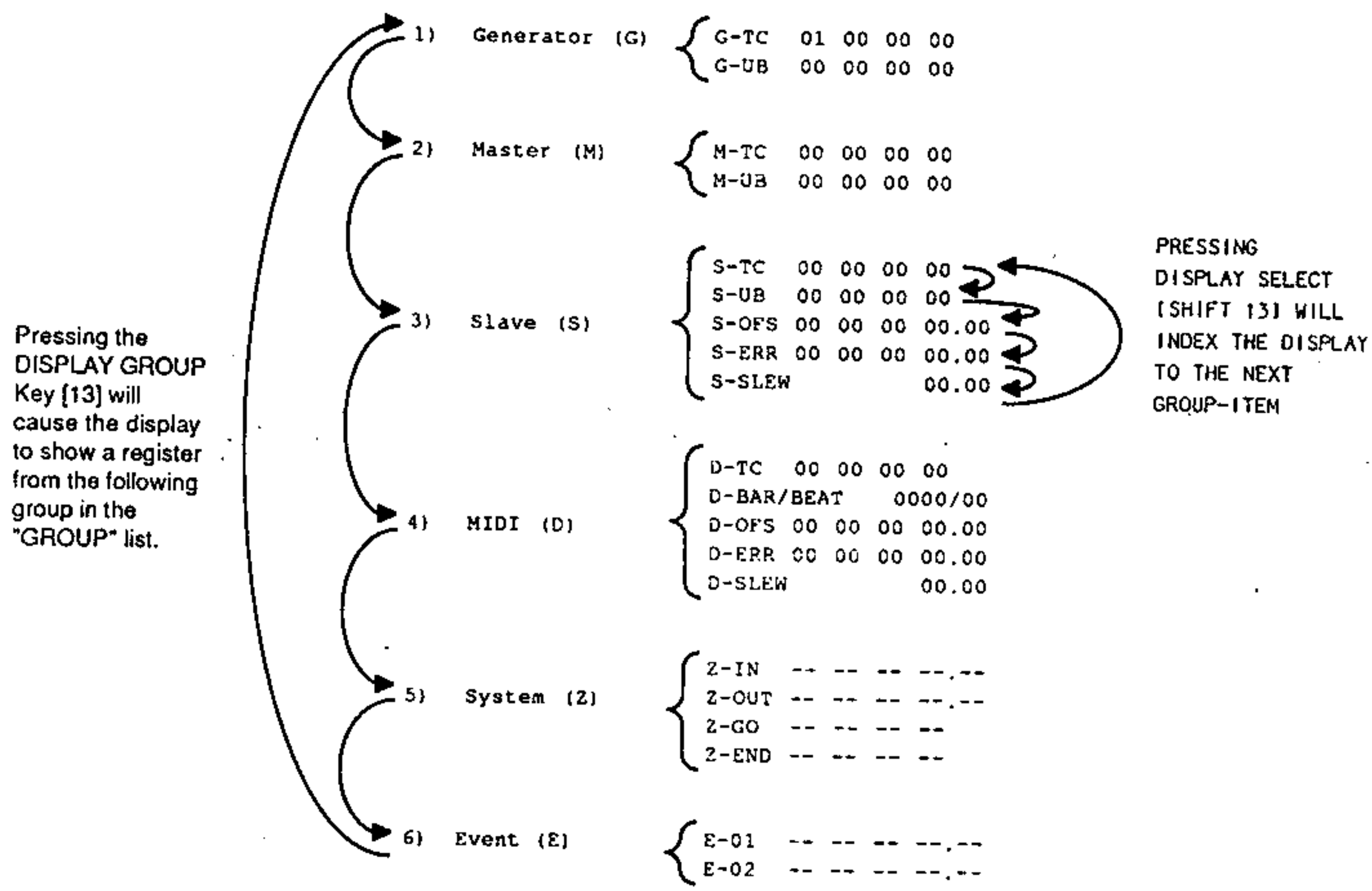


Figure 3.3 LIST OF GROUPS AND SELECTIONS

3.6 Using Menus

3.6.1 What Is a Menu?

There are six Menus in the ZETA-THREE, one for each Group. Each Menu contains a list of items related to that Group and, like the Group selections, the title of the Menu item describes the function that the item affects. For example, the Generator Menu, shown in Figure 3.4 contains four items which determine how the ZETA-THREE's time code generator will operate.

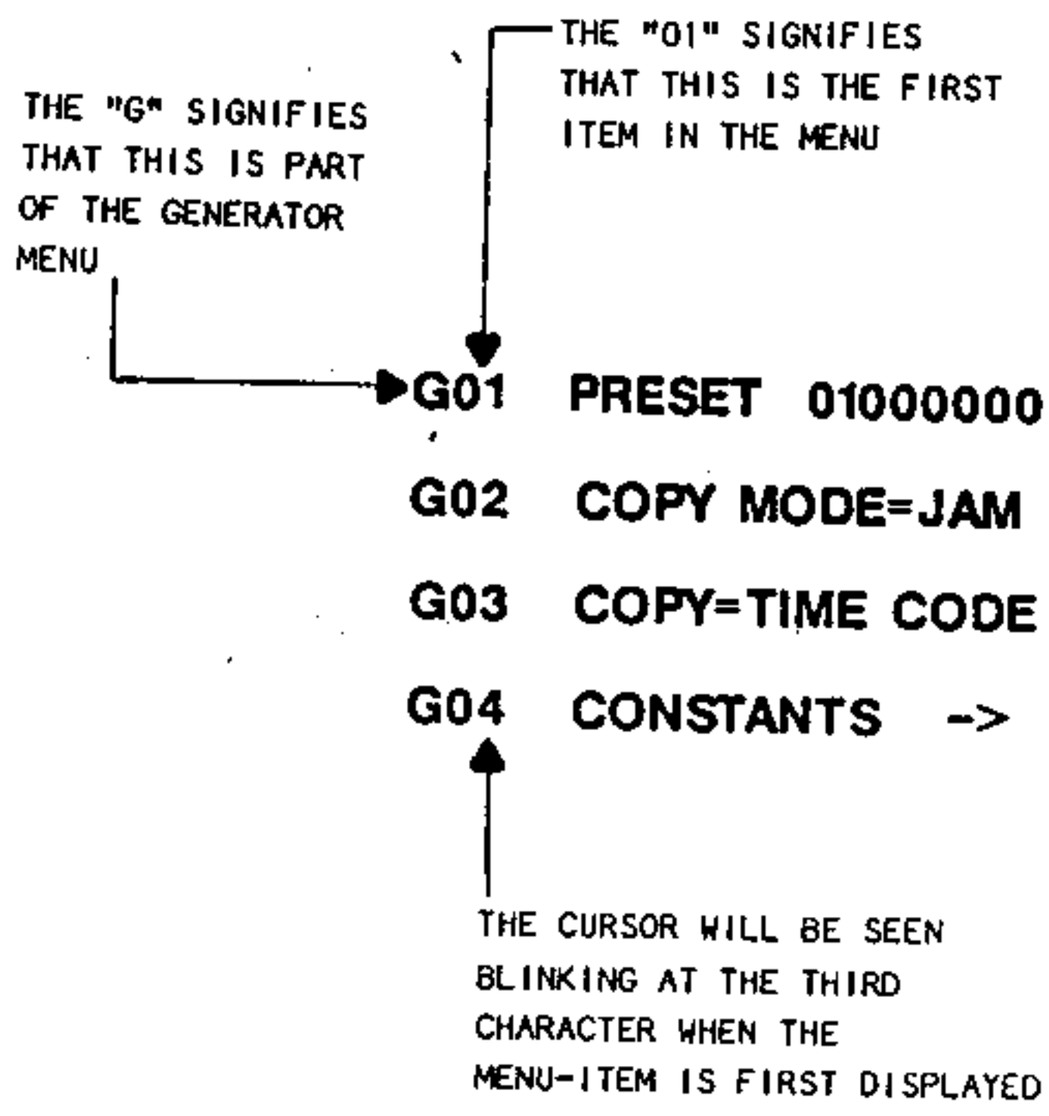


Figure 3.4 THE GENERATOR MENU

Virtually all of the ZETA-THREE's operations are directed through the use of the Menus. A complete list of the Menu items, and information about what each item affects, is given in Appendix D.

3.6.2 How to Enter and Leave Menus

To access a Menu, press the MENU key. The display will then show an item of the Menu relevent to the Group displayed before the MENU key was pressed. For example, if

the Generator Group is displayed (either Selection), press the MENU key and the Generator Menu is accessed, as shown in Figure 3.5.

After pressing the MENU key, notice that the cursor is located on the last digit of the Menu number (the second and third characters of the IDENT section of the display). The cursor must be located on the Menu item number to index to other items in the Menu. Each item in every Menu is described thoroughly in Appendix D.

To exit the Menu, and return to the Group which was displayed before the MENU key was pressed, press the DISPLAY GROUP key.

With the cursor located on the Menu number, index up or down through the various Menu items for the current Group by pressing the /\ key or the \/ key. Once a Menu has been accessed, pressing the /\ key and \/ key will index the display only through the items of that particular Menu. To gain access to the items in a different Menu, it is necessary to press the DISPLAY GROUP key to exit the current Menu, and then to press it again until the display has been toggled to the desired Group. Enter that Group's Menu by pressing the MENU key again.

3.6.3 The Menu Display

For practice, set the display to the Generator Group (either Selection). Press the MENU key, and an item from the Generator Menu appears on the display.

Use the /\ or \/ key, if necessary, to index to the Menu item titled "G01 PRESET 01 00 00 00", shown in Figure 3.6. The prefix "G" indicates that this Menu item relates to the ZETA-THREE's time code generator. The next two characters (01) indicate that this is Generator Menu number 01. The next word (PRESET) is the title of the Menu. The rest of the display shows the ten digits which, since the Generator Preset is a time code address, is read as: 1 hour, zero minutes, zero seconds, and zero frames.

Appendix D shows that Menu Number G01 is responsible for setting the preset value on the generator. The Menu's display requires a specific number to be entered into it. For convenience, the Generator Preset has been given a default of 01:00:00:00, but any other valid time code address may be entered using the CURSOR key and indexing the number up or down.

3.6.4 Data Entry into a Menu Item

There are three types of Menu items: those that require entering a number, those which require picking one of several possible choices, and those which contain additional Menu levels.

The Menu shown in Figure 3.6, requires the entry of a number. To change the number in the item, Cursor to the first digit of the Menu item, Index the digit up or down to the desired value, and repeat as necessary. Note that, as with Group Selections, only valid values will be accepted by the ZETA-THREE.

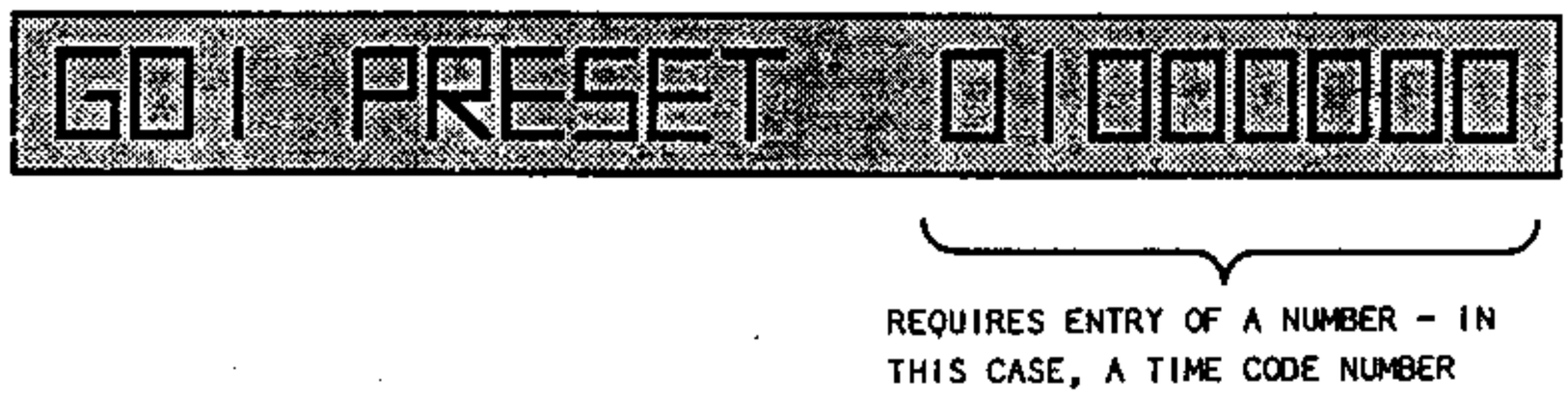


Figure 3.6

In some cases, an item which requires a number entry, such as Menu item M03, contains a set of short dashes instead of some default number. As was the case with certain System and Event Selections, the dashes mean that there is no entry in the Menu item. Numbers can be entered into these Menu items by the usual Cursoring and Indexing routines. If it is desireable to clear this type of Menu item, the CLEAR key must be pressed, as the ZETA-THREE will interpret an entry of "all zeroes" as a valid number.

G02 COPY=JAM

REQUIRES A SELECTION OF ONE OF SEVERAL CHOICES BY MEANS OF THE INDEX KEY

Figure 3.7

Menu Number G02, in Figure 3.7, is an example of a Menu which requires choosing one of several alternative displays. When the CURSOR key is pressed once, the entire word at the right side of the display starts to blink, indicating that the entire word may be changed by Indexing up or down. The choice displayed when this Menu display is accessed is the setting that the ZETA-THREE will follow. See Appendix D for more information about how each Menu item affects the ZETA-THREE.

3.6.5 Identifying and Using Higher Menu Levels

Some Menu items contain additional Menu levels. When an item of this type is accessed, an arrow appears on the right side of the display (indicating an additional Menu level), as shown in Figure 3.8, and the LED on the MENU key starts to blink. Pressing the MENU key again will access the higher Menu level. Items in a higher-level Menu are numbered and can be Indexed up or down, just as lower level Menu items are.

There are two ways to exit from a higher Menu level: 1) press DISPLAY GROUP to exit to the original Group display; 2) press PREV (SHIFT MENU) to return to the PREVious Menu level.

The PREV key may also be used when it is desireable to return quickly to a higher Menu level from a Group Selection display. When a particular Group Selection is displayed,

pressing PREV will return the display to the Menu which was last accessed, regardless of the Menu level.

Some higher-level Menus may be a list of items from among which a choice is to be made. If, when a higher-level Menu is accessed, the LED on the CAPTURE key lights solidly, it is indicating that: 1) the higher-level Menu is a list of choices; and 2) the item currently being displayed is the most recently CAPTUREd choice (or is the default).

If the higher-level Menu list is Indexed up or down from the previously-chosen item, the CAPTURE key's LED will begin to blink. If another item in the Menu list is preferred, pressing CAPTURE will cause the CAPTURE key's LED to light solidly again, indicating that the currently displayed item is the most recent choice. If the higher-level Menu list is Indexed up or down from the previously-chosen item, but the higher-level Menu list is exited without CAPTURE having been pressed, the choice will not change.

For example:

1) access the M_TC Selection.

2) press MENU, and Index up or down to Menu Number M04. The arrow indicates that there is a higher Menu level to be accessed.

3) press MENU again. The CAPTURE LED will light solidly, indicating that the transport displayed is either the most recently CAPTUREd transport or that is the default.

4) Index up or down to a different transport. The CAPTURE key LED will begin to blink, indicating that the item has not been CAPTUREd.

5) press the CAPTURE key to CAPTURE the item on the display. The CAPTURE key's LED will light solidly again.

6) Index up or down to a different transport. The CAPTURE key LED will again begin to blink.

7) press the DISPLAY GROUP key to exit the Menus.

8) press PREV [SHIFT MENU] to return to the PREViously accessed Menu level. The display will return to the Menu level which was last accessed, but the display will show the item which was CAPTUREd, not the item which was last seen on the display.

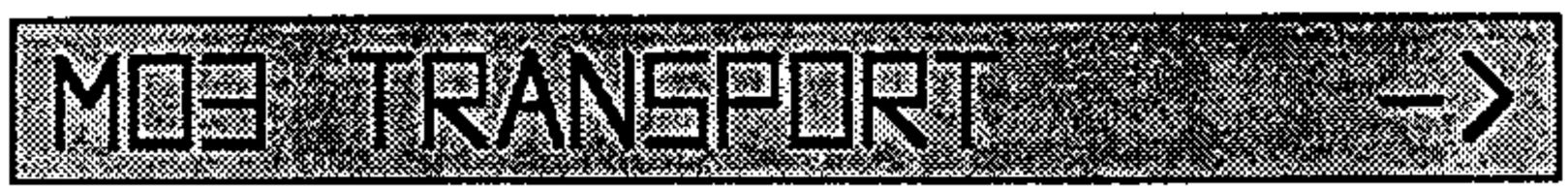


Figure 3.8

The entire list of Menu items is found in Figure 3.9 and the Quick Reference Card.

Chapter 4 - Basic ZETA-THREE Operations

How to use this chapter...

This chapter provides step-by-step instructions for operating the time code generator, recording time code on tape, reading time code from tape, and synchronizing a slave transport to a master time code source. Follow this Chapter carefully to become familiar with basic ZETA-THREE operation.

NOTE

If following these procedures step-by-step for the first time, DO NOT plug in the transport connector cables yet.

4.1 Time Code Generator Operation

One of the most important features of the ZETA-THREE is its ability to generate SMPTE/EBU time code. The time code numbers being output can be displayed whenever desired by accessing the Generator Time Code (G_TC) display.

4.1.1 Starting and Stopping the Generator

To start the generator, press the GEN RUN key [10]. Note that the amber LED on the GEN key will light when the key has been pressed, showing that the generator is running, and the cursor will disappear, as shown in Figure 4.1. Time code, counting up from the address displayed prior to GEN RUN being pressed, will be shown on the display.

The GEN RUN key acts as a toggle to start and stop the generator counting. To make the generator stop, press the GEN RUN key again. The amber GEN RUN LED will turn off and the display will show the last time code address generated.

The generator can be toggled on and off regardless of the current display. Therefore, it is possible to observe or adjust other displays while the generator is running. The amber LED on the GEN key will always light, no matter what is being shown on the display, whenever the generator is running.

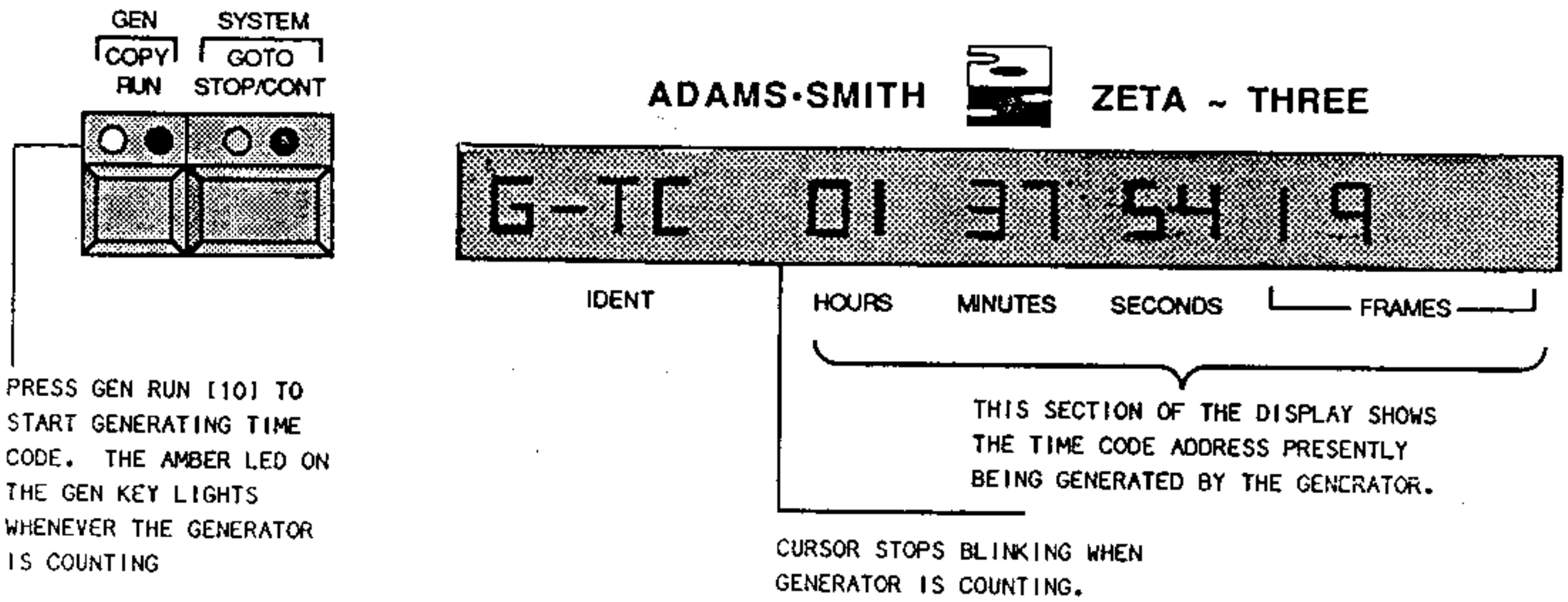


Figure 4.1 GENERATOR OPERATION

When the generator is stopped, it will continuously generate the number which is shown in the generator time code display. Because of this, it is advisable to stop a tape, on which time code is being recorded, before stopping the generator. This will avoid recording a series of identical time code numbers.

4.1.2 Setting the Generator Numbers

The address in the G_TC display may be changed anytime the generator is stopped and Generator Time Code is displayed on the screen.

When G_TC is being displayed, pressing the CLEAR key will reset the time code generator to 00 00 00 00. Alternatively, when G_TC is being displayed, pressing the CAPTURE key will set the G_TC display to the value which is stored in the "G01 PRESET" Menu item. The G01 item can be set to any valid time code number.

4.1.3 Setting the System Standard

Prior to using the time code generator, it is important to set the ZETA-THREE generator to the proper system standard. The system standard determines the type of time code the generator will generate. The five types of time code and frame rate combinations which can be generated by the ZETA-THREE are:

30-frame SMPTE
29.97-frame (video rate)
NTSC/SMPTE drop-frame
29.97-frame NTSC/SMPTE non-drop-frame
25-frame PAL/EBU
24-frame film/SMPTE

See Appendix A for more information about time code.

The system standard is set by accessing the System (Z) Menu, Indexing up or down to System Menu item number "Z04 FRAMES=30.00", CURSORing to the right side of the display, and then Indexing up or down to change the frame rate to the appropriate system standard.

4.2 Recording Time Code on Tape

Recording time code on a tape (striping) is the first step in synchronizing and autolocating. The time code allows the ZETA-THREE to identify unique locations on the tape.

The following sections provide information about the procedures which should be followed when striping a tape. THE TYPES OF CABLES NEEDED FOR ALL CONNECTIONS ARE DESCRIBED IN APPENDIX C.

4.2.1 Setting a Tape Transport for Recording Time Code

When recording time code, the transport must always be set to run at **"internal" or "fixed"** speed so the transport will run at its standard play speed.

Some transports have front panel or rear panel internal/external switches. Others automatically remain in "internal" unless instructed to switch to "external" through their Remote connector. See your tape transport's manual to set your tape transport to "fixed" or "internal".

4.2.2 Sending Time Code to the Recorder

Connect the GEN CODE OUT jack [28] on the ZETA-THREE to the "audio in" jack on the tape transport which corresponds to the track on which time code is to be recorded. Some transports provide a special TIME CODE IN jack and a wide-band time code track specifically for this purpose.

Use shielded, twisted-pair audio cable with appropriate connectors on each end. The ZETA-THREE end of the cable requires a professional quality 3-wire audio plug (type XLR-

3J or equal). The transport end may require a similar plug (type XLR-3P or equal), an RCA phono plug, or other type, depending upon the transport. Consult the transport manual to determine the type needed.

The following figure shows proper connections between the ZETA-THREE's GEN CODE OUT jack and various types of tape transport audio in jacks.

NOTE

> To determine whether a plug should be male or female, remember that the pins of the male connectors should point in the direction of signal flow (in this case, from the ZETA-THREE to the transport).

Although there is no standard rule for recording time code, it is usually recorded on an outside audio track of a tape, most commonly on the highest-numbered track.

4.2.3 Recording Procedure

First enter the desired starting time code address into the G_TC display. A starting time code is usually a number rounded to the nearest hour or half-hour. If desired, the G01 PRESET address may be loaded into G_TC by pressing CAPTURE while G_TC is displayed and the Generator is stopped (the G01 PRESET default is 01:00:00:00, but the Preset may be any valid time code address). A starting time of 00:00:00:00 should not be used because crossing from 00:00:00:00 to 23:59:59:29 (NTSC) and back again during cueing can sometimes cause system problems.

On the audio track chosen to record time code, set the time code recording level as follows:

1" vtr's = -5vu to -10vu
3/4" & 1/2" vcr's = -3vu to -5vu
audio transports = -5vu to -10vu

Values lower than -10 dB may make time code recovery unreliable; values higher than -3 dB may cause adjacent track cross talk. Select the time code track on the transport, put the transport into Record, and start the generator by pressing GEN RUN. When the desired length of time code has been recorded, stop the transport.

NOTE

The ZETA-THREE outputs time code (with unchanging numbers) even when the generator is not counting. Because of this, the recording level can be set without putting the generator into the RUN mode.

Until recording time code becomes a routine operation, the time code should be checked by following the steps described in Section 4.3 BEFORE using it for any synchronizing purposes. Checking the time code could prevent many future headaches by confirming that it has been recorded properly (i.e. a proper level, no discontinuous sections, etc.).

4.3 Reading Time Code From a Tape

After recording time code on a tape, it should be played back and read by the ZETA-THREE to confirm that it was recorded properly. Poor time code on tape is the primary cause of synchronizing problems. The ZETA-THREE is capable of recognizing good, readable time code and the MTC LED [2] or STC LED [3] turns on when good time code is present.

4.3.1 Connecting Time Code to the ZETA-THREE

The ZETA-THREE reads time code through the MASTER CODE IN [29] and SLAVE CODE IN [31] jacks. To read the time code from the master transport, connect the appropriate "audio out" jack of the master transport (or "time code out" jack if the transport has a time code track) to the MASTER CODE IN jack [29] located on the rear panel of the ZETA-THREE. To read time code from a slave transport, connect to the appropriate "audio out" jack of the slave transport the SLAVE CODE IN jack.

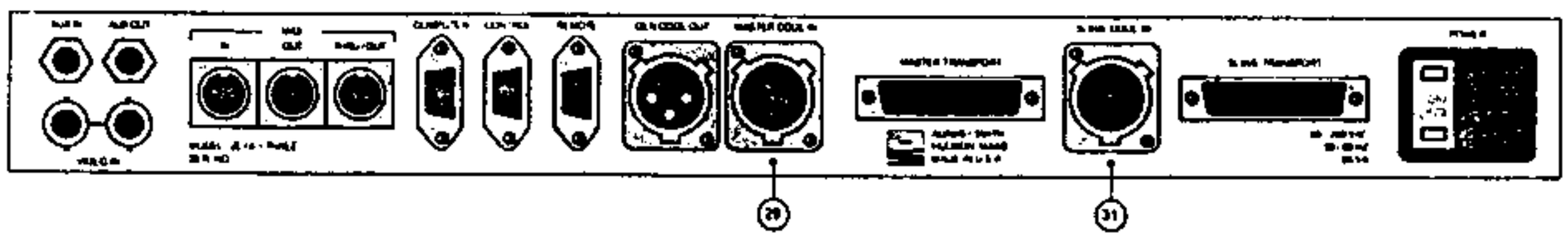


Figure 4.2 Connections for Reading Time Code

Use shielded, twisted-pair audio cable with appropriate connectors on each end. The ZETA-THREE end requires a professional quality 3-wire audio plug (type XLR-3P or equal).

4.3.2 Play-back Procedure

Rewind the tape to (approximately) the beginning of the time code on the tape.

Access the M_TC display.

Put the transport into Play. The numbers in the M_TC item will not change until the ZETA-THREE starts to read time code from the tape.

With the master tape playing, look at the green MTC LED [2], shown in Figure 4.3. IF the MTC LED is NOT lit, no time code is present. If the MTC LED flickers, the time code is discontinuous or is being received at a level which is too low. Check the play-back level on the transport to be sure the time code was recorded at an acceptable level. Adjust the play-back level, if possible. If the level remains too low, the time code <u>must</u> be re-recorded. Check the recording level on the transport before re-recording, because it is most likely the cause of the problem. If the recording level was set properly, then refer to section 6.1.1.

Use the S_TC display and the STC LED [3] for observing slave time code.

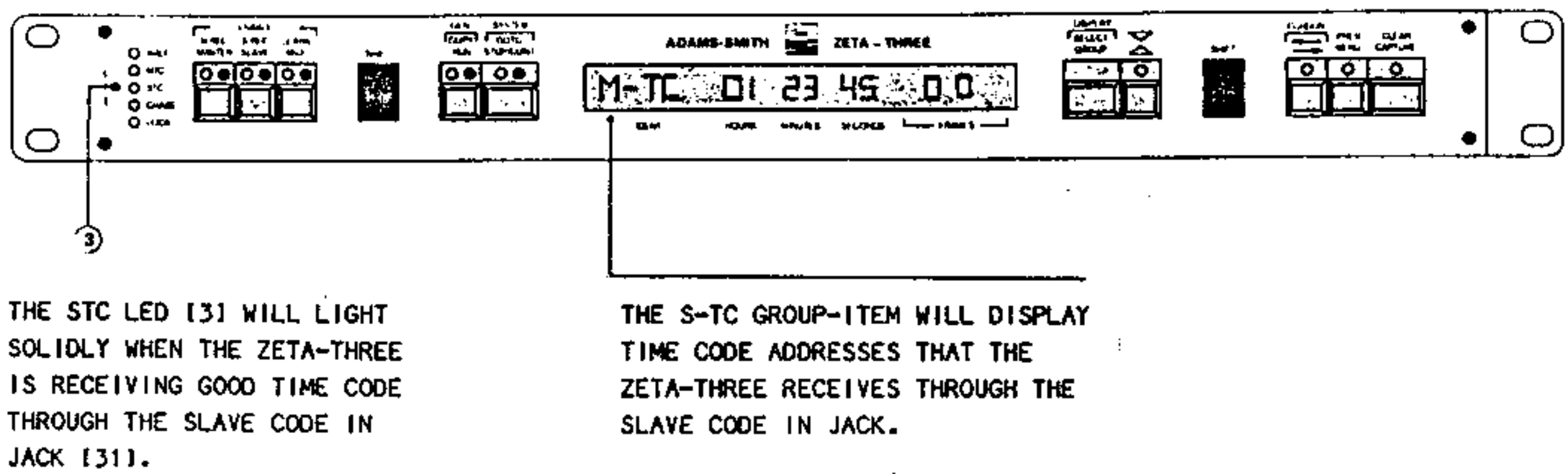


Figure 4.3 Reading Slave Time Code

4.3.3 Reading Time Code During Fast Forward or Rewind

Tape transports which have a wide-banded audio track, such as an "address track", can reproduce time code at a wide range of speeds. Time code from these transports can be read by the ZETA-THREE while the transports are in Fast Forward and Rewind as well as in Play, because the time code readers in the ZETA-THREE can decode time code from approximately 1/20 to 100X play speed.

If wide-band time code is not available (the transport does not reproduce time code in Fast Forward and Rewind), the ZETA-THREE keeps track of tape location by using tach pulses or control track pulses to update the last accurately-read time code address. The tach/control track pulses are fed to the ZETA-THREE over the same Interface cables which are used to control the master and slave transports.

4.4 Synchronizing Two Transports

The ZETA-THREE tape synchronizer's primary job is to control the capstan of a "slave" tape transport in such a way that the transport's tape runs in synchronism with the the tape on the "master" transport whenever the master is in Play. At other speeds -- Fast Forward, Rewind, and Stop -- the ZETA-THREE keeps the slave in the proper relationship to the master to re-establish synchronism, as quickly as possible, when the master is returned to Play. Most professional audio tape transports and video tape transports (ATRs and VTRs) can be used by the ZETA-THREE as master transports; most ATRs and a number of VTRs/VCRs can be used as slaves. This section explains how to use the ZETA-THREE to synchronize tapes. Appendix B provides a general discription of tape synchronizing.

When using the ZETA-THREE to synchronize a slave tape to a master tape, some limitations on types of transports must be observed. If the master is to be a video transport, then the slave may be either an audio or video transport. If the master is an audio transport, then the slave can only be another audio transport. A single ZETA-THREE can NOT, by itself, synchronize a video transport to an audio transport.

4.4.1 External Connections

In order for a tape transport to be controlled and/or synchronized by the ZETA-THREE, it must be connected to the ZETA-THREE by means of an Interface Cable. Interface cables are available for most popular professional quality tape transports and are listed on the ZETA-THREE price sheet. When looking up transports for use with the ZETA-THREE, make

sure to use the most recent price list because cables for new transports are added very frequently.

The Interface Cables handle all of the control signals described in Section 4.1.1 with the exception of time code. Time code connections between a transport and the ZETA-THREE must always be made by means of shielded, twisted pair audio cables, as described in Sections 4.2 and 4.3.

The slave tape transport must always be connected to the ZETA-THREE by means of both a time code cable and a ZETA-THREE Interface Cable.

If the master tape transport is to be controlled by its own controls or by another controller, and never by the ZETA-THREE, <u>and</u> if the transport is capable of reproducing readable time code reliably in fast forward and rewind, as well as at play speed, then the only connection needed between the master transport and the ZETA-THREE is a time code cable.

If the master tape transport is to be controlled by its own controls or by another controller, and never by the ZETA-THREE, <u>but</u> the transport is <u>not</u> capable of reproducing readable time code reliably in fast forward and rewind, as well as at play speed, the connections needed between the master recorder and the ZETA-THREE are a time code cable and a cable connnecting the tach/control track and direction signals to the ZETA-THREE's MASTER TRANSPORT connector.

If the master tape transport is to be controlled by the ZETA-THREE, then, a ZETA-THREE Interface Cable as well as a time code cable <u>must</u> be used between the master transport and the ZETA-THREE.

WARNING

The power switch [33] for the ZETA-THREE should be OFF when the transport cables are plugged into, or unplugged from the ZETA-THREE's MASTER TRANSPORT connector [30] or SLAVE TRANSPORT connector [32].

4.4.2 Synchronizer Set-up for Controlling Transports

The ZETA-THREE contains within itself the information it needs to control the transports which are connected to it, **but it MUST be told which transports are indeed connected to it.** This is done through the ZETA-THREE's transport list.

To choose a slave transport:

1) Access the Slave (S) Menu.

2) Index to Menu item "S05 TRANSPORT ->".

3) Access the next higher Menu level (the list of Slave Transports).

4) Index up or down through the list of Slave Transports to the name of the desired Slave Transport.

5) When the desired Slave Transport is shown on the display, CAPTURE it into the ZETA-THREE's memory by pressing the CAPTURE key.

The Slave Transport in the ZETA-THREE's memory can be changed at any time simply by Indexing to another transport on the list and pressing CAPTURE again.

Choosing a Master Transport is accomplished in the same manner, through Master Menu item number "M04 TRANSPORT ->".

4.4.3 Transport Set-up for Synchronizing

Before the ZETA-THREE can synchronize the slave to the master transport, the transports must be set up as described below. (Note that some of the conditions may have already been implemented by following the procedures mentioned in previous sections.)

1. The Master transport MUST be set to internal (fixed) speed control.

2. The Slave transport MUST be set to **external** speed control, allowing the transport's speed to be controlled by the ZETA-THREE.

3. The Master and Slave transports must be chosen as described above.

NOTE

Once transport selection has taken place, it is not necessary to re-select transports whenever the ZETA-THREE is turned on. Even after the ZETA-THREE has been powered-down it will remember transports last chosen.

4.4.4 Controlling the Transport Through the Front Panel Keys

4.4.4.1 The ENABLE Keys

The ENABLE keys [6, 7, and 8] determine what devices will obey ZETA-THREE commands in a particular ZETA-THREE installation.

All ENABLE keys are toggles. Pressing ENABLE MASTER tells the ZETA-THREE that there is a master in the system. The Master is "ENABLEd" when the amber LED on the ENABLE MASTER key [6] is lit. Pressing ENABLE MASTER again removes the master from the system EVEN IF A MASTER TRANSPORT IS CONNECTED TO THE ZETA-THREE'S MASTER TRANSPORT CONNECTOR. The Slave transport or MIDI are ENABLEd or disabled in the same manner. Any combination of the ENABLE keys may be used. THE ZETA-THREE ASSUMES THAT THE LEFT-MOST ENABLE KEY HAS THE HIGHEST PRIORITY.

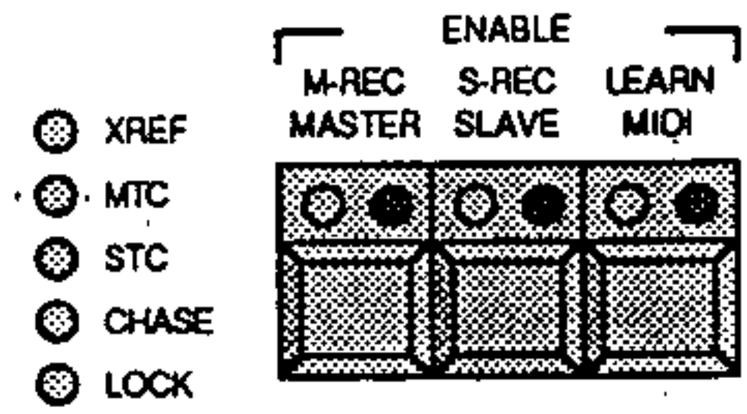


Figure 4.4

The following are examples of the ENABLE keys' prioritization:

1) If only a Master is ENABLEd (and proper cable connections have been established):

   the ZETA-THREE will be capable of sending to the Master transport either STOP/CONT commands, or GOTO commands.

2) If both a Master and a Slave are ENABLEd:

   paragraph 1 will be true;

   the ZETA-THREE will read the time code address in the M_TC, S_TC and S_OFS Group selections; and

the ZETA-THREE will synchronize the slave to the master (for best results, play the master and slave tapes for a few seconds prior to enabling either);

Also, whenever both Master and Slave are in either Fast-Forward or Rewind, and either is disabled, both transports will stop. If both transports are in Play, and either is disabled, both transports will continue in Play.

3) If the Master and MIDI are ENABLEd:

   paragraph 1 will be true; and

   MIDI sync will be synchronized to the Master time code and sent according to information in the ZETA-THREE's Beatmap.

4) If the Master, Slave, and MIDI are ENABLEd, paragraphs 1, 2 and 3 will be true.

5) If only the slave is ENABLEd, then the ZETA-THREE will send either STOP/CONT commands, or GOTO commands, to the Slave transport.

6) If the Slave and MIDI are enabled, the MIDI sync being sent will be synchronized to Slave time code.

7) If only MIDI is enabled, MIDI sync is transmitted at its own rate, as determined by the Beatmap.

NOTE

The ZETA-THREE will accept time code from the Master and/or Slave transport through the MASTER and SLAVE CODE IN jacks [29] and [31], and will display such time code, WHETHER OR NOT THE TRANSPORTS ARE ENABLED. If the transport is not ENABLEd, however, the ZETA-THREE will not make any effort to control or synchronize it.

4.4.4.2 The STOP/CONT Key

The SYSTEM STOP/CONT key [11] will toggle between Stop and Play commands to the left-most ENABLEd transport. For example, if only the Slave and MIDI are enabled, the Slave is ACTING as the master of the system and MIDI will follow it. If only MIDI is enabled, MIDI is ACTING as the master of the system (but, of course, there is nothing in the ZETA-THREE to follow it).

The amber LED on the key will be lit when the left-most ENABLEd transport is in Play (CONTinued). Immediately after the key has been pressed to issue a Play command, the amber LED on the key may blink before the transport starts to actually play the tape. The blinking LED indicates that the ZETA-THREE has acknowledged that the key has been pressed, but the transport has not yet begun to Play.

4.4.4.3 The GOTO Key

When the SYSTEM GOTO key [SHIFT SYSTEM STOP/CONT] is pressed, the left-most ENABLEd entity will "go to" the time code address stored in the Z_GO Group Selection (the address in this display is also known as the "GOTO address"). When the transport reaches the GOTO address it will stop and wait for a new command. This action (GOing TO the GOTO address and stopping at it) is called "cueing and parking". If no address is stored in the Z_GO Selection display when the GOTO key is pressed, the command will be ignored.

NOTE

> The time code address in the Z_GO Selection display will be used as the "cue point" for the left-most ENABLEd transport. If an offset has previously been established between the Master and Slave time codes, and left-most ENABLEment is switched between Master and Slave, the ZETA-THREE automatically adjusts the Z_GO address by the amount of the offset.

4.4.5 Reactions of the ZETA-THREE when Synchronizing

Synchronizing of the slave to the master will take place when both the Master and Slave transports are ENABLEd.

With both transports ENABLEd, and the master transport in Play, the ZETA-THREE will control the speed of the slave transport to keep it running in synchronism with the master.

With both transports ENABLEd, and the master stopped, the ZETA-THREE will move (cue) the slave and stop (park) it to the location which will permit the most rapid synchronization to occur when the master is returned to Play.

With both transports ENABLEd, and the master in Fast Forward or Rewind, the ZETA-THREE will cause the slave to chase the master by issuing fast forward and rewind commands to the slave. The CHASE LED [4] on the left side of the

ZETA-THREE's front panel lights whenever the slave is chasing the master, and turns out when the slave has parked.

4.4.6 Initial Synchronizing Setup

The purpose of tape synchronizing is to keep time-related material recorded on the slave tape running in synchronism with corresponding material recorded on the master tape. Therefore, good, readable time code MUST be recorded on both master and slave tapes. Thereafter, if the appropriate Interface and time code cables have been installed, and if the ZETA-THREE has been told what tape transports are in use (all as described in previous sections), then tape synchronizing can begin.

First, however, the ZETA-THREE must be told the address relationship between master and slave time codes. This relationship, determined by subtracting master time code address from slave time code, is called the offset, and is conventionally referred to as "slave minus master".

In many cases, the offset will be zero because corresponding master and slave material have the same time code addresses. In fact, zero-offset time code is often recorded, specifically to keep things straightforward.

When initially setting up the ZETA-THREE, it is usually convenient to begin with tapes which have the same time codes on them, so the offset can remain set to zero. During initial testing, it is the operation of the ZETA-THREE that must be tested and observed, and not the material recorded on the tapes. In fact, tapes containing only time code are perfectly acceptable for test synchronizing on the ZETA-THREE.

When synchronizing for the first time it is advisable to follow all the steps listed below.

Preparing Tapes and Transports

1. Load tapes on both master and slave transports. **Stripe a MINIMUM of 30 minutes of time code on both the master and slave tapes.** Set Z04 FRAMES=29.97 (25 PAL); set Z05 SYSTEM=UNRESOLVED. Refer to Sections 4.1, "Time Code Generator Operation" and 4.2, "Recording Time Code on Tape" when preparing tapes for synchronization.

2. Make SURE that the Internal/External Capstan Speed Control switch on the slave, if there is one, is in the External position.

3. Access the M_TC Selection, press ENABLE MASTER and then press SYSTEM STOP/CONT. The master transport will go into Play, the MTC LED [2] will light, and the M_TC display will show play-speed time code.

   It has now been confirmed that the Play line of the ZETA-THREE Interface cable is properly connected, and that the Master transport will output play-speed time code, and that the ZETA-THREE will read it.

4. Press SYSTEM STOP/CONT again to stop the Master.

   This will keep the Master from playing past the end of recorded time code while Step 5 is followed.

5. Access the Z_GO Selection. Press CAPTURE, and the time code address in the M_TC Selection display will be CAPTUREd into the Z_GO display.

   A cue point has now been CAPTUREd from incoming master time code.

6. Press SYSTEM STOP/CONT again to return the Master to Play. Let the Master continue to play for a minute or two. While it is Playing, access the M_TC display. After the Master has played for a minute or two, stop the Master. After the Master stops, press SYSTEM GOTO. The Master will cue (in this case, rewind) to the GOTO address. While the Master is cueing, the M_TC display will show master time code counting backwards at rewind speed. If the Master's time code channel is not wide-banded, the character "T" will appear on the right side of the M_TC display, showing that the ZETA-THREE is using tach/control track updating.

   It has now been confirmed: that the Rewind line of the ZETA-THREE cable has been properly connected; if the transport's time code channel is wide-banded, that wind-speed Master time code is being output and read; or, if the time code channel is not wide-banded, that tach/control track updating is taking place.

7. After the Master has cued and parked, disable the Master.

Slave Transport Operations

8. Make sure that the Master is disabled. Access the S_TC Selection, press ENABLE SLAVE and then press SYSTEM STOP/CONT. The slave transport will go into Play, the STC LED [3] will light, and the S_TC display will show play-speed time code.

9. Press SYSTEM STOP/CONT again to stop the Slave.

10. Access the Z_GO Selection. Press CAPTURE, and the time code address in the S_TC Selection display will be CAPTUREd into the Z_GO display.

11. Press SYSTEM STOP/CONT again to return the Slave to Play. Let the Slave continue to play for a minute or two. While it is Playing, access the S_TC display. After the Slave has played for a minute or two, stop the Slave. After the Slave stops, press SYSTEM GOTO. The Slave will cue (in this case, rewind) to the GOTO address. While the Slave is cueing, the S_TC display will show Slave time code counting backwards at rewind speed. If the Slave's time code channel is not wide-banded, the character "T" will appear on the right side of the S_TC display, showing that the ZETA-THREE is using tach/control track updating.

12. After the Slave has cued and parked, disable the Slave.

    The same cable connections and time code functions which earlier were confirmed for the Master transport have now been confirmed for the Slave.

Synchronizing

13. Access Menu number S01 LOCK MODE. Confirm that LOCK MODE equals ADR. If it does not, Index the display to ADR.

14. Access the Slave Offset Selection (S_OFS). If the display does not show all zeroes, press CLEAR [SHIFT CAPTURE].

    The offset has now been set to 00:00:00:00.00. Since this manual section (Section 4.5.6, Initial Synchronizing Setup) began with the identical time code number stream being recorded on both slave and master tapes, this is the correct offset between them.

15. Access the Slave offset ERRor selection (S_ERR). Since it is highly improbable that the two

transports are stopped at exactly the same time code address, there will be an error value shown on the display. This value is the difference between the time code address of the Master and the time code address of the Slave (calculated as slave minus master).

16. ENABLE both Master and Slave. The CHASE LED [4] will turn on and the Slave will immediately begin chasing the Master. Since the Master is stopped, the chasing will, in this instance, consist of cueing to the proper address, relative to the master's address, which will allow subsequent rapid synchronization. The S_ERR display will show the error value counting down towards zero. When the Slave has cued and parked, the CHASE LED will turn off, but a small value of offset error will remain on the display after the Slave has cued and parked.

17. Press SYSTEM STOP/CONT. The CHASE LED will turn on, the Master will Play, and the ZETA-THREE will commence achieving synchronization of the Slave to the Master. The S_ERR display will count down to zero. When it reaches zero the LOCK LED [5] will turn on and the CHASE LED will turn off. The Slave is now synchronized to the Master.

4.5 Offsets

4.5.1 What is an Offset?

Initial synchronizing setup as per the preceding section was accomplished with identical time code sequences on both Master and Slave tapes. When the Master and Slave tapes have DIFFERENT time code numbers, however, there must be an offset to establish a positional relationship between the master and slave tapes.

An offset is calculated as slave time code address minus master time code address and includes the sign of the difference, positive or negative. A negative offset occurs when the slave time code address is smaller than the master time code address. A negative offset can be calculated either as slave-minus-master, or, since the slave address is the smaller of the two, it can be subtracted from the master address and prefixed as a minus sign. An offset value is always shown as 10 digits -- hours, minutes, seconds, frames, and sub-frames (tenths and hundredths of frames).

The exact value of the desired offset must be entered or CAPTUREd in the S_OFS display in order for the master and

slave tapes to align correctly -- to achieve audio-to-video "lip-sync", for instance.

If the time code on the slave tape has been copied from the master tape along with audio material, the desired offset will be zero, as in the setup procedure in the previous section.

In many cases where an offset between the master and slave tapes is required, frame-accuracy will often be sufficient, and the sub-frames digits may be zeros.

If the exact offset is known, then it can be entered into the S_OFS Selection by means of CURSORing and Indexing, as explained above.

NOTE

> If one (or more) of the tapes contains drop-frame code, refer to Section 4.6.6.

If the exact offset is not known, then it must be determined by a trial-and-error routine. A common way is to position the master and slave tapes, either manually or by use of the ZETA-THREE controls, in the approximately correct positional relationship. The approximate offset may then be CAPTUREd. Capturing an offset, and then adjusting the offset value by either "trimming" or "slewing" until the correct offset is established, is explained below.

Any CAPTUREd or entered offset will be retained on non-volatile (NV) memory when the ZETA-THREE is powered down. If it is desirable to reset the offsets to all zeros whenever the ZETA-THREE is turned on, refer to Menu item Z13, described in Appendix D.

4.5.3 Capturing an Offset

Offsets can be CAPTUREd while the tapes are playing or stopped. In either case, time code must have been read from both master and slave tapes since the ZETA-THREE was last powered up.

NOTE

> For greatest accuracy when capturing an offset, it is recommended that both transports be playing when the CAPTURE key is pressed.

4.5.4 Trimming Offsets

Trimming an offset means changing the number in the S_OFS Selection display to a more accurate number (usually by a small amount). Offsets can be trimmed at any time by incrementing or decrementing any of the digits in the S_OFS display by CURSORing and Indexing.

Whenever the offset is trimmed, synchronism will be temporarily lost and then regained, with the time to re-gain synchronism depending primarily upon how much the offset was incremented or decremented.

4.5.5 Slewing an Offset

Another method of changing the offset is to adjust the positional relationship of the master and slave tapes. This is accomplished by increasing or decreasing the speed of the slave transport while it is running in synchronism with the master, thereby changing the running position of the slave tape with respect to the running position of the master tape. The process is called "slewing".

Slewing can be done at either a slow (1/10 frame per second) or fast (1 frame per second) rate. Slewing can <u>only</u> be accomplished while the transports are running in synchronism.

When the slave transport's speed is increased, its running position advances with respect to the master. When the slave transport's speed is decreased, its running position retards with respect to the master. As the running position of the slave transport advances or retards (with respect to the master), the offset changes correspondingly.

The ZETA-THREE monitors the offset **change** caused by slewing and lists the amount of change in the S-SLEW display. Once the desired running position of the slave with respect to the master is reached, the amount of offset change caused by the slew can then be easily added to the value in the S_OFS display.

How to slew:

1. Establish synchronism of the slave to the master, in Play.

2. Access the S_SLEW Selection. The S_SLEW will show the cursor in the tenths-of-frames position in the display, indicating that the ZETA-THREE will slew the tape in increments of 1/10 of a frame (the fast-slew rate of one frame per second). If the cursor is moved to the hundredths of frames position in the display, the ZETA-

THREE will slew the tape in increments of 1/100 of a frame (the slow-slew rate of 1/10 of a frame per second)

3. For now, leave the cursor at the slow-slew position. Next, hold down the /\ key to speed UP the slave. The word "ADVANCE" will appear in the S-SLEW display, both the CHASE and LOCK status LEDs will begin to blink, and the offset change caused by the slew will appear at the cursor position. The slew value will continue to increase as long as the /\ key is held down. When the /\ key is released, the last slew value will remain in the display.

   To slew the slave in the opposite direction, press the \/ key to slow DOWN the slave. The word "RETARD" will appear in the S_SLEW display and, as when advancing the slave, the offset change caused by the slew will appear at the cursor position. The slew value will continue to decrease as long as the \/ key is held down. When the \/ key is released, the last slew value will remain in the display.

4. After slewing, the value in the S_SLEW display can then be captured by pressing the CAPTURE key. Pressing CAPTURE will cause the SLEW value to be added to the S_OFS value, the CHASE status LED to turn off, and the LOCK LED to light solidly.

   After slewing, if the decision is made not to retain the new running position of the slave with respect to the master, the S_SLEW value can be returned to zero by pressing the CLEAR key. The running position of the slave returns to the original offset as the slew value decreases to zero. When CLEAR is pressed, the slew value will return to zero (and the running position of the slave will return to the original offset) at the same speed (1/10 frame per second or one frame per second) as slewing took place. When the slave transport has returned to its original offset, the CHASE status LED will turn off, and the LOCK LED will light solidly.

4.5.6 Entering and Capturing Drop-Frame Time Code Offsets

In general, the techniques used to enter non-drop frame offset values are <u>not</u> valid when one or both time codes are drop-frame. Specifically, subtracting a master time code address from a slave time code address to determine an offset will yield an **incorrect** answer, with two exceptions. The only two special cases where slave-minus-master will be correct are: when the two values are equal (i.e. the offset is zero); and when the value of the master is zero (i.e., slave-minus-master equals the value of the slave time code address. Except for these two special cases, an offset

value should not be entered into the S_OFS register when one or both time codes are drop-frame, as it will lead to an incorrect running position of the slave with respect to the master.

While a drop-frame time code offset should not be calculated and manually entered, it can be CAPTUREd, and the ZETA-THREE will calculate the correct offset automatically. However, when the offset is captured and calculated by the ZETA-THREE, the offset value in the S_OFS display will be in non-drop-frame format; therefore, it will not necessarily be equal to the value which would be obtained by subtracting the slave from the master time code reading.

In order for the ZETA-THREE to calculate an offset involving drop-frame codes, however, it must know which tapes contain drop-frame code. Since the ZETA-THREE determines which codes are drop-frame by monitoring the drop-frame bits in the codes it is receiving, each tape **must** be played for a few seconds to allow the ZETA-THREE to read the code and determine the type of code the tape contains.

Once a drop-frame offset has been CAPTUREd, it can be trimmed and slewed, just as non-drop-frame offsets.

Chapter 5 - General Operations

Using this Chapter...

Each ZETA-THREE operator function is described in this chapter. Many sections of this chapter begin with a list of Group Selections and/or Menu numbers to be accessed, and keys to be pressed, to implement the function being discussed.

NOTE

IT IS ASSUMED THAT THE USER IS COMFORTABLE WITH ALL THE FUNCTIONS AND OPERATIONS DISCUSSED IN CHAPTERS 2, 3 AND 4.

NOTE

IT IS ASSUMED THAT THE USER IS COMFORTABLE WITH ALL THE FUNCTIONS AND OPERATIONS DISCUSSED IN CHAPTERS 2, 3 AND 4.

NOTE

IT IS ASSUMED THAT THE USER IS COMFORTABLE WITH ALL THE FUNCTIONS AND OPERATIONS DISCUSSED IN CHAPTERS 2, 3 AND 4.

5.1 Reading Time Code

5.1.1 Reading Master Time Code

To read Master Time Code, the time code output of the tape transport must be connected to the MTC IN jack [29] on the ZETA-THREE's rear panel. When the master tape is playing, the ZETA-THREE will read time code from the tape and display the numbers in the M_TC Selection.

The master transport does not need to be ENABLEd when the ZETA-THREE is reading time code. M_TC will always display time code numbers which are coming in through the MTC IN jack.

5.1.2 Reading Slave Time Code

To read Slave Time Code, the time code output of the tape transport must be connected to the STC IN jack [31] on the ZETA-THREE's rear panel. When the slave tape is playing, the ZETA-THREE will read time code from the tape and display the numbers in the S_TC Selection.

The slave transport does not need to be ENABLEd when the ZETA-THREE is reading slave time code. S_TC will always display time code numbers which are coming in through the STC IN jack.

5.1.3 Tach/Control Track Updating

Many transports do not recover time code at other than play speed. If time code is being read from a transport while it is in Play, and that transport is put into Rewind, time code will disappear providing that: the transport's time code channel is not wide-banded; tape lifters operate (ATRs); or the tape unwraps (VCRs). In such a situation, the ZETA-THREE will update the last accurately read time code address by counting incoming control track or tach pulses from the transport, if control track/tach and direction lines from the transport have been properly connected to a transport connector on the rear panel of the ZETA-THREE. When tach/control track updating is taking place, the M_TC (or S_TC) Selection will show a "T" at the extreme right of the display.

5.2 Operating the Time Code Generator

Basic time code generation and recording, and the external connections used therein, are described in Section 4.2. This section describes additional generator operation. Any additional cables needed for a given procedure are listed.

5.2.1 Generator Preset

Access Selection: G_TC
Press keys: CAPTURE

If a time code address has been entered into the G01 display, then pressing CAPTURE <u>while G_TC is displayed</u> causes the ZETA-THREE to copy the number from the G01 PRESET Menu to the G_TC display.

5.2.2 Resolving the Generator to an External Source

Access Selection: none
Access Menu number: Z05, Z06. Set Z05 to SYSTEM=RESOLVED; set Z06 to resolve reference choice.
Press key(s): GEN COPY

NOTE

Setting Z05 to SYSTEM=RESOLVED resolves the entire system, not merely the time code generator. This has important implications for the use of video tape recorders as explained later in this section.

Time code must be generated at a specific rate. In the ZETA-THREE, this rate is derived either from an internal quartz crystal (Z05 SYSTEM=UNRESOLVED, the default condition), or, if the entire ZETA-THREE system has been resolved, from an external reference (Z05 SYSTEM=RESOLVED). This reference can be:

1) a video signal from a sync pulse generator (composite video, composite sync, black burst, bars).

   This signal is fed to the ZETA-THREE through either of its rear panel's VIDEO IN jacks. The other VIDEO IN loopthrough jack is provided either to output the video signal to additional video equipment or to terminate the video signal by a 75 Ohm load.

2) the "video out" signal from a video transport. THIS REFERENCE MAY BE UNSUITABLE IF THE VIDEO IS AT ALL UNSTABLE.

   This signal is fed to the ZETA-THREE through either of its rear panel's VIDEO IN jacks. The other VIDEO IN loopthrough jack is provided either to output the video signal to additional video equipment or to terminate the video signal by a 75 Ohm load.

3) composite sync from an Adams-Smith MODEL 2600 RG Reference Generator.

   This signal is fed to the ZETA-THREE through either of its rear panel's VIDEO IN jacks. The other VIDEO IN loopthrough jack is provided either to output the video signal to additional video equipment or to terminate the video signal by a 75 Ohm load.

4) mains (AC line) frequency.

When resolved to MAINS, the ZETA-THREE automatically derives a reference rate of 30 frames per second (60Hz mains) or 25 fps (50Hz mains).

5) a square wave or pulse signal.

The signal is fed to the ZETA-THREE rear panel's AUX IN jack. Resolving the system via the AUX IN jack is very unusual; an example of its use would be setting up to generate off-speed time code.

One example of the need to resolve the ZETA-THREE system when generating time code is when recording new time code on video tape. In this case, a video sync reference (see above) must be fed to the ZETA-THREE (see above), and the same reference must be used as an external reference for the video tape recorder on which the time code the time code is to be recorded. This will insure that both the ZETA-THREE's time code generator and the video recorder are operating at the same rate, and that the frame edges of the newly generated time code accurately coinside with the frame edges of recorded video.

To resolve the ZETA-THREE (to a video signal, for this example):

1) set Menu number Z05 to SYSTEM=RESOLVED. If no external reference has been fed to the ZETA-THREE, the XREF status LED will begin to blink unless Z06 RESOLVE=MAINS.

2) Set Z06 to RESOLVE=VIDEO (default setting) if it has not already been done. If no external reference has yet been fed to the ZETA-THREE, the XREF status LED will blink.

3) If it has not yet been done, feed an external reference (as described above) to the ZETA-THREE. Once the external reference connection is made, the XREF LED will turn on solidly.

4) Set Z04 to the rate compatible with the standard of the incoming video sync reference.

5) Set up for striping time code as explained in previous sections. When GEN RUN is pressed, the newly generated time code sent out through the GEN CODE OUT jack will be at the same rate as the video reference, and will have frame edges aligned to those of the reference.

5.2.3 Generating Time Code

Access Selection: G_TC
Access Menu number: G01, Z04. Set G01 to desired preset; Set Z04 to desired frame rate.
Press keys: GEN RUN

The ZETA-THREE is always generating time code, whether or not the generator is "running". If the generator is not running, the same time code address is generated continuously.

See section 4.2 for instructions on generating time code.

5.2.4 Generating User Bits

Access Selection: G_UB. Set to desired User Bit information (see below).
Access Menu number: none
Press keys: GEN RUN

In addition to always generating time code, the ZETA-THREE will also generate user bits. The user bits generated may be any eight hexadecimal digits, and those eight digits are a part of each time code address that the generator generates.

Enter the desired user bits information into the G_UB display (default: 00:00:00:00) by CURSORing and Indexing. Whatever is set into the G_UB Selection will be generated as a part of the time code.

5.2.5 Copying Time Code and/or User Bits

Access Selection: none
Access Menu number: G02, G03. Set copy mode in G02; set copy choice in G03.
Press keys: GEN COPY

Sometimes it is convenient to copy time code from one tape to another. For example, when preparing tapes for synchronizing with a zero offset (S_OFS = 00:00:00:00), the exact same time code number stream must be on both tapes; if one tape already has time code on it, that time code can be copied to the other tape.

The ZETA-THREE provides two types of copy modes, Jam Sync and Transfer. In both modes, the ZETA-THREE reads the incoming time code number stream (from the left-most ENABLEd device) and generates a new, but identical time code number

stream. The user bits from the same time code source may also be copied, if desired (see below).

There is only one difference between jam syncing and transferring, but it is a very significant one.

When jam syncing, if the COPY source stops, the newly generated time code (and user bits, if desired) **will continue to be output through the GEN CODE OUT jack, continuing the time code number sequence without skipping or pausing.**

Jam syncing can be used to reconstruct defective sections of time code on a tape. The tape is positioned at the start of the earliest sequence of good time code and is copied in the jam sync mode. When the copy source time code fails at the defective section, the new time code will continue the original time code sequence.

When transferring, if the COPY source time code sequence stops, the newly generated time code sequence will also stop. If the source sequence starts again, the newly generated time code sequence will also continue, and will again output the COPY source time code number stream.

To copy time code, connect the "time code out" or "audio out" jack of the device from which time code is to be copied to either the ZETA-THREE's MTC IN or STC IN jack. Then connect the ZETA-THREE's GEN CODE OUT jack (through which the new copy of the time code is to be generated and sent) to the "time code in" (or "audio in") jack of the machine which is to record the copied time code. The addresses coming from the GEN CODE OUT jack can then be recorded onto tape in the same manner that was described in section 4.2.

5.2.5.1 Copying

Access Selection: none
Access Menu number: G02, G03. Set G02 to copy mode choice;
set G03 to copy choice.
Press key(s): GEN COPY

ENABLE only the transport which contains the original time code, and rewind it to the beginning of the time code numbers which are to be copied.

When GEN COPY is pressed:

1) If G03 COPY=TIME CODE, new first-generation time code, exactly COPYing the time code of the copy source, will be sent out through the GEN CODE OUT jack.

The newly-generated time code will contain the user bit information entered in the G_UB Selection display.

2) If G03 COPY=USER BITS, newly generated time code, starting with the time code address entered in the G_TC display will be sent out through the GEN CODE OUT jack. This time code will contain the exact same user bit information as was contained in the user bits of the copy source.

3) If G03 COPY=TC AND UB'S, newly generated time code, containing the exact same time code number stream and user bit information as is contained in the copy source, will be sent out through the GEN CODE OUT jack.

5.2.5.2 Resolving when Copying

When copying time code from a video recorder to another video recorder, recommended practice is to reference the ZETA-THREE to the same video rate signal as the source is referenced to. This may be accomplished by resolving the ZETA-THREE to composite video from the source transport, or, more elegantly, by resolving the ZETA-THREE to an external video rate reference signal which is also fed to both video recorders.

When copying time code from an audio recorder to a video recorder, it is recommended practice: to resolve the ZETA-THREE to a video signal; to feed the same external reference to the video recorder; and to resolve (synchronize as a slave) the audio recorder to the same reference signal.

When copying time code from a VTR to an ATR, or from one ATR to another, the ZETA-THREE may be run unresolved.

5.3 Transport Control from the ZETA-THREE's Front Panel

Through the use of the SYSTEM key, it is possible to send commands to the left-most ENABLEd device. All possible commands are listed in this section.

5.3.1 Stop and Play Commands

Access Selection: none
Access Menu number: none
Press keys: SYSTEM STOP/CONT

The SYSTEM STOP/CONT key is a two-way toggle. Pressing SYSTEM STOP/CONT sends either a Stop or Play command to the left-most ENABLEd transport.

If the tranport is Stopped (or Paused), pressing the SYSTEM STOP/CONT will cause its amber LED to blink, indicating that the ZETA-THREE has issued the command to the transport but it has not yet begun to Play. When the transport reaches Play speed, the amber LED on the SYSTEM STOP/CONT key will light solidly. An ENABLEd slave will follow, synchronize, and maintain synchronism.

If the left-most ENABLEd transport is in Play, pressing the SYSTEM STOP/CONT key will turn off its amber LED and will Stop (or Pause) the transport.

5.3.2 The GOTO Command

Access Selection: Z_GO
Access Menu number: none
Press keys: SYSTEM GOTO [SHIFT SYSTEM STOP/CONT]

The SYSTEM GOTO address may be either entered or CAPTUREd into the Z_GO Selection display.

Entering an address directly into the Z_GO Selection is accomplished by CURSORing and Indexing.

To capture a number into the Z_GO Selection display:

1) Access the Z_GO Selection display.

2) Press the CAPTURE key. The number which is in the left-most ENABLEd transport's Time Code Selection (either M_TC or S_TC) will be copied into the Z_GO Selection whenever the CAPTURE key is pressed.

NOTE

For the purpose of capturing an address in the Z_GO Selection display, the ZETA-THREE will default to capturing from the M_TC display if none of the ENABLE keys have been pressed.

NOTE

The address in the Z_GO Selection display will automatically adjust to left-most ENABLEment. For example, if only the Slave had been ENABLEd, and the GOTO address captured from Slave time code, the address will change when the

Master transport is ENABLEd. The new GOTO address will be the captured-from-slave address adjusted by the offset value in the S_OFS Selection display.

Pressing the SYSTEM GOTO key, once a GOTO address has been either entered or CAPTUREd, will cause its red LED to light and the left-most ENABLEd device to GO TO (cue to) the GOTO address (cue point). An ENABLEd slave will follow a cueing ENABLEd master, and the CHASE status LED will turn on. When both transports have stopped after cueing, the CHASE LED and the red SYSTEM GOTO LED will both turn off.

If it is desired to stop the left-most ENABLEd transport while it is cueing to a Z_GO address, press the SYSTEM GOTO key again (it is a two-way toggle). Its red LED will turn off and the transport will stop.

The left-most ENABLEd transport may be put into Play immediately upon reaching the SYSTEM GOTO address; this is called Play Preset. To invoke Play Preset:

1) press SYSTEM GOTO. Its red LED will light solidly and the left-most ENABLEd transport will begin cueing.

2) after SYSTEM GOTO has been pressed (Step 1 above), press the SYSTEM STOP/CONT key. Its amber LED will blink, and the transport will continue to cue. When the transport has parked at the Z_GO address, the SYSTEM GOTO key's red LED will turn off, the transport will begin to play, and the SYSTEM STOP/CONT key's amber LED will light solidly.

5.3.3 LOOPing

Access Selection: Z_GO, Z_END. Enter or CAPTURE the starting address of a Loop into the Z_GO display; enter or CAPTURE the ending address of a Loop into the Z_END display.
Access Menu number: Z03. Index to desired Loop type.

Setting the LOOP function determines the action of the left-most ENABLEd transport between the Z_GO and Z_END points.

When set to CYCLE, the ZETA-THREE, upon the left-most ENABLEd transport's reaching the Z_END address, will cue the left-most ENABLEd transport to the Z_GO address, and will then put the transport into Play. The transport will play until the Z_END address is reached, and then repeat the cycle. The cycle will continuously repeat until stopped.

NOTE

If any other device is ENABLEd, the left-most ENABLEd device, while cycling, will wait for the other devices to chase and park before returning to Play from a cued position.

When set to AUTO REWIND, the ZETA-THREE, upon the left-most ENABLEd transport's reaching the Z_END address, will cue the left-most ENABLEd transport to the Z_GO address, and park the transport there. Press SYSTEM STOP/CONT to repeat the sequence.

When set to AUTO STOP, the ZETA-THREE, upon the left-most ENABLEd transport's reaching the Z_END address, will stop the transport and park it. Press SYSTEM GOTO, and then SYSTEM STOP/CONT (to invoke Play Preset) to repeat the sequence.

To stop or interrupt a CYCLE:

1) while cueing is taking place and Play Preset is invoked (SYSTEM GOTO red LED lighted and SYSTEM STOP/CONT amber LED blinking), a cycle can be stopped by pressing SYSTEM GOTO again. Both LEDs will turn off and the transport will stop. Cycling will resume if the SYSTEM GOTO key is pressed again.

2) while cueing is taking place and Play Preset is invoked (SYSTEM GOTO red LED lighted and SYSTEM STOP/CONT amber LED blinking), the cycle can be stopped by pressing SYSTEM STOP/CONT. The red LED will turn off, the amber LED will light solidly, and the transport will go into Play.

3) while the transport is in Play (amber LED on solidly), the cycle can be stopped by pressing SYSTEM STOP/CONT. The amber LED will turn off and the transport will stop.

5.4 Synchronizing

The ZETA-THREE is capable of synchronizing one slave transport to one master. Basic synchronizing operation is described in section 4.5. The following sections describe specific requirements for synchronizing various combinations of audio and video transports.

5.4.1 Generator as Master

It is possible to use the ZETA-THREE's time code generator as the master for a synchronized slave. This procedure is useful when first setting up a synchronizing system, as it removes one complete set of variables (i.e., quality of master time code, master cable, etc.). To make the ZETA-THREE's time code generator the master:

1) Access menu number M01 and set its display to MASTER=GENERATOR.

2) Enter or CAPTURE an appropriate offset to prevent the slave from winding its tape off the reel while attempting to cue and park.

3) Press ENABLE MASTER and ENABLE SLAVE. The slave will cue to the G_TC address and park.

4) Access G_TC; the time code addresses appearing in this display will be considered master time code by the ZETA-THREE. Press either GEN RUN or SYSTEM STOP/CONT; the ZETA-THREE will now synchronize the slave transport to the time code generator.

As an alternative to Steps 1 through 4:

A) route time code from the ZETA-THREE's GEN CODE OUT jack to the MASTER CODE IN jack.

B) enter or CAPTURE an appropriate offset.

C) access M_TC or G_TC to observe master time code.

3) Press ENABLE MASTER and ENABLE SLAVE. The slave will cue to the M_TC/G_TC address and park.

D) press GEN RUN. The ZETA-THREE will synchronize the slave to master time code (which, in this case is also the generator time code).

5.4.2 Transport as Master (one ZETA-THREE)

When using one Zeta-Three to synchronize a slave transport to a master transport, the following cables must be used:

1) The slave machine requires a ZETA-THREE interface cable between the slave transport and the ZETA-THREE's SLAVE TRANSPORT Connector.

2) An appropriate cable from the time code channel on the slave transport must be connected to the SLAVE CODE IN jack.

3) An appropriate cable from the time code channel on the master must be connected to the MASTER CODE IN jack.

4) If time code is not available at other than play speed (the master transport time code recovery amplifier is NOT wide-banded, and/or its tape does not remain against its heads during Fast Forward and Rewind), then tach (or control track) pulses, a direction signal, and a ground connection from the master transport must be sent to the ZETA-THREE's MASTER TRANSPORT connector. This wiring can be provided by a ZETA-THREE interface cable.

5) If it is desired to control the master transport with the ZETA-THREE's STOP/CONT and GOTO keys, a ZETA-THREE interface cable is **required** between the master transport and the MASTER TRANSPORT Connector.

5.4.2.1 Audio to Audio

To synchronize an audio slave transport to an audio master transport:

1) Set frame rate for system (Menu number Z04) to conform to proper local standard (any NTSC rate or 25-frame PAL).

2) Confirm that the system is not resolved to an external reference (Menu number Z05 SYSTEM=UNRESOLVED).

3) Ascertain that both master and slave tapes have been properly prepared for synchronizing; with a sufficient amount of known-good time code.

4) Make the proper ZETA-THREE interface cable connections between the transports' connectors and the ZETA-THREE's MASTER TRANSPORT and SLAVE TRANSPORT connectors.

5) Make the appropriate time code connections between the transports and the ZETA-THREE's MASTER CODE IN and SLAVE CODE IN jacks.

6) Press ENABLE MASTER. Confirm that master time code is appearing in the M_TC Selection during Play, Fast Forward and Rewind. Confirm that the

transport obeys SYSTEM STOP/CONT and SYSTEM GOTO commands. Confirm that tach/control track pulse updating is occuring correctly. Disable Master, press ENABLE SLAVE and repeat confirmation sequence.

7) Access the S_OFS Selection and enter or CAPTURE an acceptable offset between master and slave time codes.

8) Press both ENABLE MASTER and ENABLE SLAVE to initiate master transport control and synchronization of the slave.

5.4.2.2 Audio to Video

To synchronize an audio slave transport to a master video transport, follow Steps 1 through 8 in Section 5.4.2.1, <u>except</u>:

in Step 1, set frame rate for system (Menu number Z04) to either NTSC video rate, or to 25-frame PAL.

if an external video rate reference is available, it is recommended that it be fed to the video master (if the transport can accept it) and to the ZETA-THREE. This will ensure that the sync word of the time code on the audio tape will synchronize to the video frame edge, instead of to the sync word of the master time code (which may not be properly aligned to the video frame edge). If using an external reference, set Z05 SYSTEM=RESOLVED (Step 2), and set Z06 RESOLVE=VIDEO.

NOTE

If no external video rate reference is available, connect the Video Out of the master video machine to one of the VIDEO IN jacks of the ZETA-THREE, and connect the other VIDEO IN jack to the video monitor (or terminate it). If the video out is stable enough, this may be adequate as a system reference.

5.4.2.3 Video to Video

To synchronize a video slave transport to a master video transport, follow Steps 1 through 8 in Section 5.4.2.1, <u>except</u>:

in Step 1, set frame rate for system (Menu number Z04) to either NTSC video rate, or to 25-frame PAL.

feed the same external video rate reference to both video transports; this will insure that synchronism is not lost due to differences between the two transports' internal crystals. In addition, it is recommended that the same reference signal be fed to the ZETA-THREE (it may be looped through the ZETA-THREE on the way to one of the transports), to ensure that the sync word of the time code on the slave tape will synchronize to the video frame edge, instead of to the sync word of the master time code (which may not be properly aligned to the video frame edge). If feeding an external reference to the ZETA-THREE, set Z05 SYSTEM=RESOLVED.

NOTE

Some less expensive VTRs do not have sync inputs BUT some of these will lock to a video signal fed into their Video Input jacks. Rarely do the VTR manuals mention this feature.

Synchronizing two video machines without external sync, or using VTRs which do not accept an external sync reference, **MIGHT** work, but the situation is MARGINALLY ADEQUATE. If it is necessary to synchronize two video machines on a frequent basis, proper video machines should be used.

5.4.3 Synchronizing to an External Reference

Synchronizing an audio transport to an external sync reference is known as resolving. To resolve an audio transport, follow Steps 1 through 8 in Section 5.4.2.1, <u>except</u>:

in Step 2, set Z05 SYSTEM=RESOLVED.

if the external reference is mains frequency, set Z06 RESOLVE=MAINS.

if the external reference is a video rate reference signal, it must be fed to one of the ZETA-THREE's VIDEO IN jacks, and terminated at the other. Set Z06 RESOLVE=VIDEO.

if the external reference is to be a waveform from a function generator, set Z06 RESOLVE=AUX IN.

in Step 3, be concerned only with the preparation of the tape for the transport which is to be resolved.

in Step 4, connect the proper ZETA-THREE interface cable to the ZETA-THREE's SLAVE TRANSPORT connector.

in Step 5, connect the transport's time code output to the ZETA-THREE's SLAVE CODE IN jack.

in Step 6, confirm time code in the S_TC Selection only. Confirm SYSTEM STOP/CONT and SYSTEM GOTO commands for ENABLEd slave transport only.

skip Step 7.

in Step 8, ENABLE slave only.

NOTE

The transport will resolve if it receives a Play command from the ZETA-THREE but **not** if it receives a Play command from its own control panel.

5.4.4 Resolved Transport as Master (two ZETA-THREEs)

It may occasionally be necessary to synchronize a slave transport to a master transport which is itself being resolved to an external reference. An example of this would be synchronizing a slave video transport to a master audio transport. Since the slave video transport will be locked to its own servo when playing after address synchronism has been achieved (or to an external reference, if frame-edge alignment is required), it is necessary to force the audio master to run at the same video rate (by resolving it) when playing. In this configuration, one ZETA-THREE would be used to resolve the master transport, and another ZETA-THREE would be used to synchronize a slave transport to the master.

To synchronize a video slave to a resolved audio master:

1) prepare the master audio transport to be resolved by one of the ZETA-THREEs, as described in Section 5.4.3. Note that this transport must be considered the slave of the ZETA-THREE which will be resolving it, notwithstanding the fact that it is the master transport in the overall synchronizing system.

2) feed the same external reference (to which the master is being resolved, and which should by this point have been fed to the resolving ZETA-THREE) to both the other ZETA-THREE (the synchronizing ZETA-THREE) and the video slave transport.

3) in the resolving ZETA-THREE, set G02 COPY MODE=XFER. Set G03 COPY=TIME CODE. Establish proper connections between the GEN CODE OUT jack of the resolving ZETA-THREE to the MASTER CODE IN jack of the synchronizing ZETA-THREE. Press GEN RUN, ENABLE SLAVE, and then SYSTEM STOP/CONT on the resolving ZETA-THREE to put the master transport into Play. Master time code will appear in the MTC Selection of the synchronizing ZETA-THREE.

NOTE

Remember that in this configuration, the master transport is the resolving ZETA-THREE's slave transport, and that there will be no ZETA-THREE interface cable between the master transport and the synchronizing ZETA-THREE. Therefore, to play, stop, or cue the master transport, it is necessary to ENABLE the slave transport of the resolving ZETA-THREE and use its SYSTEM STOP/CONT and SYSTEM GOTO keys.

4) prepare the slave video transport to be synchronized by the other ZETA-THREE, as described in Section 5.4.2.3. When this has been accomplished, pressing ENABLE SLAVE on the synchronizing ZETA-THREE will allow the slave transport to be played, stopped or cued by the synchronizing ZETA-THREE's SYSTEM STOP/CONT and SYSTEM GOTO keys. Pressing both ENABLE MASTER and ENABLE SLAVE on the synchronizing ZETA-THREE will allow synchronizing to begin. Be sure that a proper offset between master and slave time codes has been entered or CAPTUREd into the S_OFS display of the synchronizing ZETA-THREE.

5.4.6 Lock Modes

When synchronism of a slave transport to its master has been achieved, the slave is said to be "locked" (as indicated by the ZETA-THREE's LOCK status LED). The ZETA-THREE has three lock modes, Address, Freewheel and Auto, selectable via Menu number S01 LOCK MODE.

5.4.6.1 Address Mode (default)

Address Mode (S01 LOCK MODE=ADR) is used under normal synchronizing situations, with both master and slave tapes containing known-good time code throughout. In Address Mode, the ZETA-THREE uses both time code addresses and frame edge references when maintaining the slave transport in synchronism with its master. Slave time code is specifically related to master time code, and the relationship is defined by the offset. The slave frame edge reference is derived from slave time code, and the master frame edge is derived either from master time code or from an external video sync reference.

Address mode is used for most transport-to-transport synchronizing, and provides lock accuracy to 1/1000 of a frame.

5.4.6.2 Freewheel Mode

In Freewheel Mode (S01 LOCK MODE=FWL), the ZETA-THREE refers only to master and slave frame edges when synchronizing; time code address numbers on both tapes are ignored. As with the address mode, the slave frame edges are derived from slave time code, and master frame edge is derived either from master time code or from an external video rate reference. When synchronizing in Freewheel Mode, the slave transport will run at the same speed as its master, but, since time code addresses are ignored, there will be no specific positional relationship between master and slave and therefore no meaningful offset.

When Freewheel Mode is invoked, and both master and slave transports are ENABLEd, the ZETA-THREE will locate the nearest time code frame edge on the slave tape and lock it to the next master frame edge it receives.

Freewheel Mode is occasionally used to resolve an audio tape recorder to an external source of time code. When resolving to an external video rate reference with the ZETA-THREE, however, lock mode selection is automatic and Menu number S01 need not be considered. It is also used when it is desirable to play through discontinuous patches of time code, because the synchronizer will ignore breaks in time code addresses.

Freewheel Mode, since it does not use time code addresses, ignores offsets greater than one frame; however, because the ZETA-THREE synchronizes the frame edge of the slave time code to the master frame edge, the sub-frame portion of the offset is not ignored. To adjust the relative position of the tapes within a frame, slewing the tranport is possible.

5.4.6.3 Auto Mode

The Auto Mode (S01 LOCK MODE=AUTO) is a combination of the Address and Freewheel Modes. When Auto Mode is used, the ZETA-THREE will operate in Address Mode while the slave is being synchronized to the master, and will switch automatically to Freewheel Mode a selected number of seconds after synchronism has been achieved. The ZETA-THREE returns to the Address mode whenever synchronous running ceases (i.e., transport is stopped, or put into Fast Forward or Rewind).

The Auto mode is useful while playing through sections of discontinuous or poorly-recorded time code when it is desireable to retain the initial offset. Retaining the initial offset in this manner is useful for re-synchronizing when the tapes are re-cued or stopped and started again.

5.4.6.4 Slow Relock

If a tape has been physically spliced, or if time code has been electronically edited without reference to time code frame edges, a frame edge phase error (a shift in position of frame edge with respect to the preceding frame edge) will probably occur creating a sub-frame offset error and, therefore, a loss of synchronism. The Slow Relock routine (Menu number S02 SLOW RELOCK=ON) can be used to provide sub-audible re-syncronization of the slave to the master following the splice. Slow Relock is especially useful when synchronizing in Freewheel or Auto Mode, to provide sub-audible re-synchronization following a time code discontinuity.

5.4.7 Recovering from Poor Time Code

Time code interruptions, discontinuities or drop-outs will cause loss of synchronism, disrupting synchronizing operations. The ZETA-THREE has two recovery routines to cope with such situations.

5.4.7.1 Lock Hold

If slave time code drops-out (becomes unreadable, but not discontinuous) the ZETA-THREE will automatically maintain slave tape speed very accurately, permitting the drop-out to be "played through" without an error occuring. When a slave transport is playing through such a time code drop-out, it is said to be in a "Lock Hold" condition.

5.4.7.2 Splice Trap

If a splice occurs in an area containing critical material, where no re-synchronization can be tolerated, the Splice Trap routine (S03 SPLICE TRAP=ON) can be used. Splice Trap allows synchronization to continue in the Address Mode without interruption by capturing a new offset following detection of the splice. Lock will be maintained, and no re-synchronization will occur. Since the new offset will replace the old offset in the S_OFS Selection, the old offset will be lost and must be re-entered or re-CAPTUREd if it is to be used again.

5.5 Record and Rehearse

5.5.1 Record

If Menu number Z01 IN/OUT=RECORD, the ZETA-THREE can send a Record signal to the master and/or slave transport.

To set-up the ZETA-THREE to "punch" an ENABLEd master transport in and out of Record, press ENABLE M-REC [SHIFT ENABLE MASTER]. The ENABLE M-REC key's red LED will start blinking indicating that a record signal will be sent to the master when a time code address from the master transport matches the time code address which has been entered into the Z_IN Selection.

When the Record signal is sent (at the Record In point), the red LED will light solidly, indicating that the transport is in Record.

NOTE

> The Record LED will indicate that the transport is in Record if a record tally is fed to the ZETA-THREE through the Interface cable. If the transport does not have a record tally output, then the Record LED will light solidly when the ZETA-THREE sends the Record command.

The ZETA-THREE will also send a "Stop Record" signal when the time code address in the Z_OUT Selection is reached. The red LED will begin blinking again when the Record Out occurs.

To disable Master Record, press ENABLE M-REC again. If Master Record has been ENABLEd, pressing ENABLE MASTER will both disable the transport and disable Master Record. If recording is in process, disabling Master Record will **not**

cause recording to stop, but will merely turn off Master Record triggering at the Z_IN or Z_OUT time code addresses.

An ENABLEd slave transport can be punched in and out of Record when a time code address from the left-most ENABLEd transport coincides with the addresses in the Z_IN and Z_OUT Selections. The Slave Record function is turned on or off (for an ENABLEd slave) by pressing ENABLE S-REC [SHIFT ENABLE SLAVE].

5.5.2 Rehearse

If Menu number Z01 IN/OUT=REHEARSE, the ZETA-THREE can send a Rehearse signal to the master and/or slave transport, IF THE TRANSPORT HAS A REHEARSE FUNCTION.

Z01 IN/OUT=REHEARSE allows transports which are capable of Rehearse functions to simulate Record without any actual recording taking place. The "Rehearse" function <u>does not</u> send a Record command to the transport.

Rehearsing is accomplished with the ENABLE M-REC and ENABLE S-REC keys, just as Record functions are.

5.6 Events

5.6.1 What is an Event?

An event is an action which occurs when a specific time code address on a tape is reached. Event outputs can be used to send a command signal to almost any device. For example, an event output might be used to trigger an effect, give a command to a transport, or turn on a cart machine -- or an electric coffee pot.

5.6.2 Events 1 and 2

The ZETA-THREE provides two momentary event outputs via the AUX OUT jack [19]. The Event 1 output will be sent to the "tip" of the AUX OUT mating plug (default -- Menu number Z10 XOUT TIP=EVENT 1); Event 2 to the "ring" (default -- Menu number Z11 XOUT RNG=EVENT 2). These outputs will trigger when a time code address from the left-most ENABLEd transport coincides with the time code address entered into the display of either the E-01 Selection or the E-02 Selection.

5.6.3 Event Operation

To implement use of the ZETA-THREE's Event 1 output:

1) Confirm that Menu number Z10 XOUT TIP=EVENT 1.

2) Access the E_01 Selection, and enter or CAPTURE a time code address into the E_01 display.

3) Wire the ZETA-THREE's AUX OUT jack to the device which is to receive the event output signal. Use a 1/4-inch stereo phone plug at the ZETA-THREE end of the cable.

4) Access Menu number E01, Index to EV_01 and set to ARMED. The ZETA-THREE's Event 1 output will now trigger whenever incoming time code from the left-most ENABLEd device coincides with the time code address in the E_01 Selection display.

   To implement use of the ZETA-THREE's Event 2 output, follow the same procedure using Menu number Z11, the E_02 Selection display, and EV_02 in Menu number E01.

5.6.4 Event Output Duration

If the device being triggered by a ZETA-THREE event output fails to respond properly to the event output signal, it may be necessary to change the duration of the output signal.

The duration of the ZETA-THREE's momentary event outputs (the length of the pulse itself) is adjustable in increments of 8-1/3 milli -seconds. Adjustment is accomplished by setting the desired increment in EVENT CONTSTANT 35 PULSE DURation. The default value is 06, or 50 micro-seconds.

ZETA THREE MANUAL ADDENDUM

ZETA THREE MIDI SECTION

Software Rev 2.00 or higher
(Test software X.10 or higher)

December 7, 1987 (GBL)

PRELIMINARY DOCUMENTATION

Revision 2

INTRODUCTION

WHAT DOES THE MIDI SECTION DO ?

The objective here is to SYNCRONIZE a MIDI controlled sequencer to the current Master device. The sequencer is then the second Slave device controlled by the Zeta Three. (For "sequencer" also read "drum machine".)

In order to do this, the Zeta must know song TEMPOS for every part of the song.

This information is stored in the Zeta Three TEMPO MAP.

When the Master begins to play, the Zeta can calculate from the MAP exactly where in the song the sequencer should be - send it there - give it a Start command - and keep it precisely locked to the Master for the remainder of the song, or until the Master is stopped.

WHAT IS A TEMPO MAP ?

"500 bars of 4/4 at 120 beats per minute" is a Tempo Map.

Of course, vastly more complicated Maps can be created - the Zeta can hold between 400 and 500 tempo and/or time signature changes in its Map.

A new tempo can be specified at any 1/16 note in the song.

New time signatures can be specified at any bar boundary.

Along with the time signature, the user may specify a "Metronome Click Length". This determines the type of beat used in the beats-per-minute specification. For example, it is usually preferable in a 6/8 bar to specify tempo in "dotted quarter notes per minute".

TEMPO MAP EXAMPLE

Here is a (not terribly typical) musical example . .

```
Tempo      = 120              = 150              = 148.50                 = 108
Bar 1                 9                  17                 20      21                 33

   || 4   8 bars    | 6    8 bars     | 3   3 bars    |    rall| 4   12 bars    ||
   || 4  |------|   | 8   |------|    | 4  |------|   | / / / | 4  |-------|    ||
```

Here's how the Zeta Three Tempo Map describes the same song . .

Beats-per-Minute	Time Sig.	MM Click*	Bar / Beat . 1/16'th
120.00	4 / 4	4	0001 / 01 .1
150.00	6 / 8	4+	0009 / 01 .1
148.50	3 / 4	4	0017 / 01 .1
144.00	..	4	0020 / 02 .1
139.50	..	4	0020 / 02 .2
135.00	..	4	0020 / 02 .3
130.50	..	4	0020 / 02 .4
126.00	..	4	0020 / 03 .1
121.50	..	4	0020 / 03 .2
117.00	..	4	0020 / 03 .3
112.50	..	4	0020 / 03 .4
108.00	4 / 4	4	0021 / 01 .1
108.00	..	4	+033 / 01 .1

* A plus sign (+) following the Click number indicates that the Click Length is a "dotted" value.

MIDI GROUP DISPLAYS

Before doing anything PRACTICAL with the Zeta MIDI section, we shall take a quick "walk" through the MIDI displays themselves - having your Zeta Three at your side would be good. No attempt is made here to explain ALL of the operating functions availiable through these displays - this is done in later sections.

THE "MAIN" DISPLAY - BARS AND BEATS

This is the first MIDI display you will see after dialling the GROUP key past the Generator, Master and Slave display groups . .

D _ B 1 2 0 . 0 0 0 0 0 1 / 0 1 . 1

The "B" in the ident portion stands for "Beats-per-minute", and the 5 numbers following it specify beats-per-minute at the current 1/16 note. (i.e. 120 BPM)

The 7 numbers to the right of the display identify the current bar, beat and 1/16 note. i.e. 1st BAR / 1st BEAT . 1st 16th note.

You will notice that the cursor is blinking at in the ident portion at the letter "B" (no other Zeta display does this !). By pressing the UP/DOWN arrow key now, you may obtain more information about the current beat . .

D _ F 1 2 . 0 0 0	"F"rames-per-Beat for film work (See FRAMES-PER-BEAT TEMPOS).
D _ C 0 4	Metronome "C"lick Length (4 = 1/4 note)
D _ T 0 4 / 0 4	"T"ime Signature ("numerator"/"denominator")

Time Signature "numerators" may be any number from 1 to 15.

Time Signature "denominators" may be any of 2, 4, 8 or 16.

Available Click Length values are . .

2+	Dotted half note
2	Half note
4+	Dotted quarter note
4	Quarter note
8+	Dotted eighth note
8	Eighth note
16	Sixteenth note

Return to the "B"eats-per-minute display before continuing.

ADJUSTING THE BAR/BEAT POSITION

Move the cursor to the "Beats" digit. [D_B120.00 0001/01.1]
^

Pressing the UP/DOWN key now will adjust your position in the song, beat by beat.

Go to beat 3.

Move the cursor back to the "Bars units" digit. [D_B120.00 0001/03.1]
^

Press the UP key.

We are now at Bar 2.

Notice that the "Beats" have been reset back to 1. This is common to all numeric adjustments in the MIDI Bar/Beat display - when any digit is adjusted, all the digits to the right of it are reset to their lowest value.

Now move the cursor to the "Bars hundreds" digit. [D_B120.00 0002/01.1]
^

Successively pressing the UP key takes us through bars 101, 201, 301, 401 and finally to bar "+501". The "+" sign here indicates that we have reached the end of the song recorded in the Tempo Map. We cannot move the bar or beat numbers any higher with the UP key.

(The Zeta Three has powers up with a default Tempo Map 500 bars in length.)

It is also possible to see a "-" (minus) sign just before the Bar number. This shows that the Zeta is positioned before the starting time of the Map. However, this condition is normally produced only when locking or cueing to Master time code, and cannot be produced with the UP/DOWN key.

Back to the display - the fastest way to return back to Bar 1, Beat 1 is to move the cursor to the "Bars thousands" position (now occupied by the "+" sign) and press the DOWN key (Shift UP). [D_B120.00 +501/01.1]
This backs us up a thousand bars at a time ! ^

THE "TIME CODE" DISPLAY

Press SELECT (Shift DISPLAY) at the "MAIN" MIDI Display to arrive at the "TIME CODE" Display.

D _ T C 0 0 0 0 0 0 0 0

This time code is purely artificial - it is the current SONG TIME, assuming that the song starts at time 00:00:00:00. If the sequencer controlled by the Zeta Three were a tape transport, then this is the code that would be recorded on the tape.

It is always in NON-DROP FRAME mode.

NOTE that this code has NOTHING WHATSOEVER TO DO WITH MIDI TIME CODE, which is an entirely different animal.

THE "OFFSET" DISPLAY

Press SELECT (Shift DISPLAY) at the "TIME CODE" Display to arrive at the "OFFSET" Display.

D _ O F S 0 0 0 0 0 0 0 0 . 0 0

This display performs EXACTLY the same function as the Offset display in the Slave Syncronizer group.

Here however, the Offset is calculated as the Song Time Code (see previous display) minus the Master Time Code.

NOTE that an Offset may also be established by setting a number in the "START TIME" menu display (see below). These two displays are interactive.

THE "ERROR" DISPLAY

Press SELECT (Shift DISPLAY) at the "OFFSET" Display to arrive at the "ERROR" Display.

D _ E R R 0 0 0 0 0 0 0 0 . 0 0

This display performs EXACTLY the same function as the Error display in the Slave Syncronizer group.

Lock up time for MIDI devices controlled by the Zeta Three is intentionally not instant. The locking process requires changing the tempo fed to the slaved sequencer, which must itself lock to the tempo received. The Zeta Three ensures that these changes are relatively smooth, and you may observe the resultant locking process by dialling up this display.

THE "SLEW" DISPLAY

Press SELECT (Shift DISPLAY) at the "ERROR" Display to arrive at the "SLEW" Display.

D _ S L E W 0 0 . 0 0

This display performs EXACTLY the same function as the Slew display in the Slave Syncronizer group.

Press SELECT (Shift DISPLAY) to return to the "MAIN" Display.

MIDI MENU DISPLAYS

1st LEVEL MENU		2nd LEVEL MENU	DESCRIPTION
D01 EDIT	->	.1 INS 000 BARS AT X .2 DELETE A -> B .3 COPY A -> B TO X .4 MARK A ----/--.- .5 MARK B ----/--.- .6 MARK X 0001/	This section controls bar insertions, deletions, and copying one section to another. (See INSERTING, DELETING AND COPYING)
D02 SONG SETUP	->	.1 START 00000000.00	Song starting time, expressed as a Master Time Code number. (May be Captured or entered digit by digit.) Interacts with Offset.
		.2 END ACTION=RUN STOP	Choice of actions taken at the end of the Song.
		.3 SONG NUMBER=001	Numeric label of the current song (may be transmitted as a MIDI Song Select message.)
D03 LEARN MODE	->	.1 AVRG CAPTURE TAP .2 EXACT CAPTURE TAP .3 MIDI CLOCK IN .4 AUX IN TIMEBASE	Selects mode for Tempo Learning. (see LEARNING TEMPOS)
D04 LEARN QNTIZE=8			Sets threshold for tempo averaging in the Learn mode. (see LEARNING TEMPOS)
D05 FPB FRM=24 25 30 24/25 24/25/30			Selects the frame rate to be used in Frames-Per-Beat tempo calculations. (see FRAMES-PER-BEAT TEMPOS)

D06 MERGE=OFF REAL TIME NO REAL TM ALL	Tells Zeta Three which types of data received at the MIDI In port should be Merged with Zeta data and re-transmitted from the Output port. (see MERGING)
D07 MIDI OUT=ON ON+SSEL OFF	Enables the Zeta to transmit MIDI messages, with or without Song Select commands. (Has NO EFFECT on D06 MERGEing requests)
D08 MIDI THRU=IN OUT 2	Assigns the MIDI Thru port to directly copy the MIDI In port (as specified by the MIDI standard), or to act as a second Output port, in parallel with the normal Output port.
D09 LOCK MODE=ADR FWL AUTO	Lock modes - see Syncronizer section.
D10 SLOW RELOCK=OFF ON	Slow Relock mode - see Syncronizer section.
D11 SPLICE TRAP=OFF ON	Splice Trap enable - see Syncronizer section.
D12 TIMEBASE -> .01 12 PER BEAT .02 16 PER BEAT .03 20 PER BEAT .04 24 PER BEAT .05 32 PER BEAT .06 40 PER BEAT .07 48 PER BEAT .08 60 PER BEAT .09 80 PER BEAT .10 96 PER BEAT .11 120 PER BEAT .12 160 PER BEAT .13 240 PER BEAT	Establishes the frequency (cycles per beat) appearing at the Timebase output (AUX OUT, Tip, if assigned). Also sets the expected frequency when Learning from a timebase input (via AUX IN).

D13 MAP LOAD/SAVE ->	.1 SAVE TO TAPE .2 LOAD FROM TAPE	Initiates saving the Tempo to tape and Loading it back again. (see SAVING THE TEMPO MAP TO TAPE.)
D14 MIDI CONSTS ->	01 - /LKERR 06 05 LOCK LIMITS 72 06 WIDE/NARROW F3 34 LOCK EXIT 20 37 - /SMS TYPE 00 38 AUTO DELAY 01 45 S-PTR WAIT 00 46 S-PTR SPACING 00 51 - /1=NVRAM 00 52 NO MSG/FPB.125 01	MIDI "Constants" (see MIDI CONSTANTS)

FRAMES-PER-BEAT TEMPOS

Frames-Per-Beat tempo calculations present somewhat of a dilemma to the system designer.

The simple logic of this method of tempo specification is that musical beats can be directly mapped to available film frames, thus enabling very precise timing.

However, as film has been running at 24 frames per second for many years, we now have several generations of film composers who have come to know the various frames-per-beat tempos by "feel" as opposed to by calculation. This is true particularly in the US.

For this reason, the Zeta Three allows the frame rate for FRAMES-PER-BEAT calculations to be specified separately from the TIME CODE frame rate. (See MIDI menu "D05 FPB FRM=xx").

The traditional film composer may leave the Zeta set permanently at its default setting of "D05 FPB FRM=24".

European composers may prefer the "D05 FPB FRM=25" setting.

And, allowing for the very real possibility of film running at 30 frames per second in the future, the setting "D05 FPB FRM=30" is also available.

Two other selections are also available which track, or partially track, the current TIME CODE frame rate.

All are summarized in the following table of frame rates to be used in the Frames-Per-Beat calculation . .

	TIME CODE Frame Rate		
	24	25	30
D05 FPB FRM=24	24	24	24
D05 FPB FRM=25	25	25	25
D05 FPB FRM=30	30	30	30
D05 FPB FRM=24/25	24	25	24
D05 FPB FRM=24/25/30	24	25	30

RUNNING THE TEMPO MAP

Return to the "MAIN" MIDI Display, preferrably with Beats-per-Minute in the display.

Press the MIDI switch in the ENABLE section. Keep the Master and Slave disabled.

Press the STOP/CONTINUE switch - your Map should now be "running". Moreover, if you have a sequencer or drum machine attached to the MIDI Out port, it should also be playing. (Don't forget to put the sequencer/drum machine in EXTERNAL clock mode !)

Press the STOP/CONTINUE switch again to stop.

By adjusting the Bar and/or Beat numbers in the display, you may position the sequencer anywhere in the Map and Start from there.

As with the tape machine sections of the Zeta Three, you may capture GOTO points and use the GOTO key (Shift STOP/CONTINUE) to quickly re-position the sequencer. If no GOTO point has been set up (GOTO register shows all dashes), then the sequencer will be cued to Bar 1, Beat 1.

LOCKING TO MASTER TIME CODE

Disable Master, Slave and MIDI sections of the Zeta (using the ENABLE switches).

Set up a tape transport as the Zeta system Master in the usual manner. (You may also use the Generator as a temporary Master by looping its code back into the Master Code Reader, or by assigning it as Master in the Master menus.)

Make sure that the Zeta Three has read Time Code from the Master tape at least once. (If the tape has no Time Code, then stripe some now !)

Stop the tape machine near the beginning of its code.

Locate the "MAIN" MIDI display and set the Bar/Beat section to 0001/01.1 .

Locate the MIDI Offset display and CAPTURE the current Offset.

ENABLE both the Master and MIDI sections.

Play the Master Tape deck.

Your Zeta Three Tempo Map should now "chase" and lock the the Master. If you have a sequencer or drum machine attached, then it too should be playing along.

NOTES:

1. Notice that, when the Master is in rewind or fast forward, the Zeta Three keeps both its Map and the sequencer correctly aligned.

2. If the Master is cued to a point BEFORE the beginning of the Song, then the Bar/Beat display will contain a NEGATIVE quantity of Bars and Beats. The sequencer, on the other hand, will be waiting at Bar 1, Beat 1.

   When the Master plays from a "negative" bar/beat, then the display will count down to the beginning of the Song, and a Continue command will be sent to the sequencer when Bar 1, Beat 1 is reached.

3. If the Master is played or cued to a point AFTER the end of the Song, then the display will show a "+" sign before the Bar number.

4. VERY IMPORTANT ! . . The MIDI section can lock either to the Master or to the Slave tape machines. Locking to the Slave will take place if only the SLAVE and MIDI keys are set to ENABLED (i.e. Master disabled).

   During the switching between "Master master" and "Slave master", the MIDI Offset is automatically adjusted.

   Thus, ANY COMBINATION of the three machines controlled by the Zeta may be operated locked together, simply by ENABLing and DISABLing appropriately.

OVERRIDING THE CURRENT MAP TEMPO

Disable the Master for now.

Locate the "MAIN" Bar/Beat MIDI Display, and set it to show Beats-per-Minute.

Move the cursor to the Beats-per-minute units digit. [D_B120.00 0001/01.1]

Press the UP key.

As expected, the tempo has been changed to 121 beats-per-minute.

In addition, an asterisk has appeared in the center of the display.
[D_B121.00* 0001/01.1]
This indicates that a tempo OVERRIDE is in effect.

No change has been made to the Tempo Map yet, allowing us to audition different tempi first. Note that this new tempo is now being transmitted from the MIDI port.

Pressing the Zeta STOP/CONTINUE key sets the Map (and sequencer) in motion AT THE NEW TEMPO. (Local Start/Stop at the sequencer should also respect the new tempo, provided that receiving of EXTERNAL Clocks has been enabled.)

The override will stay in effect until either it is entered into the Map (see next section), or cleared by pressing CLEAR (Shift CAPTURE). When CLEARed, the display will indicate that overrides have been cancelled, and the tempo will revert to the original Map tempo. [- CANCEL OVERRIDES -]

For comparison between new tempos and Map tempos, the previous override may be recalled by pressing CLEAR again. [- RECALL OVERRIDES -]

ENTERING A NEW TEMPO INTO THE MAP

To change a tempo in the Map, it is necessary to tell the Zeta three things :

(a) The new tempo
(b) Where the new tempo starts
(c) Where it finishes

We have already seen how to set up a tempo OVERRIDE. This new tempo will be used for entry into the Map.

To specify start and end points, the Zeta uses two marker registers "MARK A" and "MARK B". These registers contain the Bar/Beat values for the start and end points for the new tempo.

Using these markers to change tempos is extremely simple. In the following example, we shall change the tempo of the first 8 bars from 120 to 126 BPM . .

Start by setting a tempo override of 126.00 BPM. [D_B126.00* 0001/01.1]

Set the Map to Bar 1, Beat 1 (if not there already). [D_B126.00* 0001/01.1]

Press the CAPTURE key - this sets MARK A. [------ MARK A ------]
Note that the CAPTURE key LED is now flashing. This indicates that we have begun the process of entering the tempo change.

Set the Map to Bar 9. [D_B126.00* 0009/01.1]

Press the CAPTURE key again. [---- CHANGES OK ----]
This action has at the same time set MARK B, and entered the new tempo between MARK's A and B.
i.e. from the beginning of Bar 1
up to (but not including) the beginning of Bar 9.
The CAPTURE key LED has stopped flashing, indicating completion of the change.

Notes :

1. The position of the cursor is not relevant to the MARKing process - it may be anywhere in the display.

2. Once MARK A has been captured and the CAPTURE key LED is flashing, we may abort the change in progress by pressing CLEAR (Shift CAPTURE). [-- CANCEL CHANGES --]
   This does not cancel the tempo OVERRIDE however, and a second press of CLEAR may be required. [- CANCEL OVERRIDES -]

3. All tempos in the above examples may of course be specified using the Frames-per-Beat display.

CHANGING MAP TIME SIGNATURES AND METRONOME CLICK LENGTHS

Time signatures and Click lengths are entered in exactly the same way as tempi. When either is first changed, the OVERRIDE asterisk will appear in the display. The same MARK A, MARK B process is used to enter changes into the Map.

NOTES:

1. Time Signature and Click Length changes CAN ONLY BE MADE AT BAR BOUNDARIES.

2. Click length overrides and changes DO NOT affect the song tempo, but they DO affect the tempo specification. For example, if a 1/4 note click at 120 BPM is changed to an 1/8th not click, then the Beats-per-Minute display will automatically change to 240 BPM. However, there will be NO CHANGE to the actual tempo being produced at the MIDI OUT jack.

3. Time Signature overrides have no effect on the tempo !

4. Combinations of Time Signature, Click Length AND Tempo may all be set up and changed together.

5. Click Lengths MUST match Time Signatures. For example, if you specify a half note click in a 3/4 bar, the Zeta will give you an error message as soon as you attempt to enter this change into the Map . .
   [*17* CLICK/SIG CLASH]

6. Changes in Click Lengths cannot be made across Time Signature changes. Attempting to change the Click Length to 1/8 notes for all 16 bars of the following example will produce the error message . .
   [*22* SIG VARIATION]

```
|| 4   8 bars     | 6    8 bars      ||
|| 4  |------|    | 8   |------|     ||
```

INSERTING, DELETING AND COPYING

These actions require the use of the menus in the "D01 EDIT" menu section. (To get there, press the MENU key while in any of the MIDI Group displays; find the "D01 EDIT ->" menu; press the MENU key again.)

However, setting up the parameters for these actions will usually be done from the "MAIN" MIDI Display. (To return from menus, hit the DISPLAY key !)

Inserting, Deleting and Copying make use of various combinations of the Bar/Beat registers MARK A, MARK B, and a new register, MARK X.

The concept behind MARK X is exceptionally simple - whenever you enter the MIDI menus, the Zeta Three takes your current Bar/Beat song position and loads it into the MARK X register.

In other words, MARK X is equivalent to "current position".

INSERTING BARS

Let's start with an example . .

We wish to insert 4 bars of 3/4 time between bars 8 and 9 of our existing song.

First, stop any Map motion (disabling the Master transport is also good.)

Set a Time Signature OVERRIDE of 03/04.

Set the Bar/Beat section of the display to 0009/01.1 - in other words to the bar that we are going to insert IN FRONT of.

Enter the ".1 INSERT 000 BARS AT X" menu via the "D01 EDIT" menu.

Set that display to ".1 INSERT 004 BARS AT X".

Press the CAPTURE key.

The job is done ! The Zeta will display . . [-- BARS INSERTED ---]

Now let's look quickly at what has just happened . .

Setting the Time Signature OVERRIDE established the type of bar to be inserted. We could also have set Click Length and Tempo overrides, and these would have been used in the inserted bars. As our example specified no Click Length or Tempo overrides, the Zeta used the Click Length and Tempo values which were in effect AT THE INSERT POINT (i.e. at the beginning of bar 9).

Moving the Bar/Beat display to bar 9 allowed the Zeta to load "Bar 9" into the MARK X register as soon as we entered the EDIT menus.

Adjusting to "INSERT 004 BARS" established the number of bars to insert.

Pressing the CAPTURE key inserted 4 bars of 3/4 at MARK X, i.e. at the bar 9 bar line.

Notes:

1. Only quantities of BARS may be inserted. Insertion of single beats is not supported.

2. To add an "Intro" to an existing song, insert bars at Bar 1, Beat 1.

3. To extend an existing song, insert bars at the first bar which has a "+" sign.

4. After any insertion, the Zeta will position its Map at the END of the inserted section, ready to insert more bars, perhaps with different tempos or time signatures. It is important to realize that all bar numbers AFTER the insertion will have been incremented by the number of bars inserted.

   In our example, the Zeta positioned itself, after the insertion, at bar 13. Bars 9,10,11 and 12 are the new 3/4 bars, and bar 13 corresponds to what used to be bar 9. In the same way, what used to be bar 10 has become bar 14, and so on.

5. It is NOT essential, before inserting bars, to set the Bar/Beat display to EXACTLY the beginning of the bar at the insertion point. The Zeta will ignore any beat or 1/16 note values and load only the BAR number when it loads MARK X.

DELETING BARS

Once again, an example . .

We wish now to delete bars 5,6,7 and 8.

At the "MAIN" display, set Bars/Beats to 0005/01.1
i.e. the start of the section to be deleted.
Press the CAPTURE key.
The Zeta will display . . [------ MARK A ------]

This is very similar to the process of changing
tempos, click lengths and time signatures.

Set Bars/Beats to 0009/01.1, the end of the section
to be deleted.
Press the CAPTURE key.
The Zeta Three will display . . [------ MARK B ------]

This is now different to changing tempos etc,
because NO OVERRIDES were specified.

Enter the ".2 DELETE A -> B" menu via the "D01 EDIT" menu.

Press the CAPTURE key. Zeta displays . . [--- BARS DELETED ---]

COPYING BARS

This process starts like a deletion, and ends like an insertion !

Example . .

We wish to copy bars 5,6,7 and 8 (our 3/4 bars, if you have followed the previous examples) to a spot between bars 16 and 17.

At the "MAIN" display, set Bars/Beats to 0005/01.1
i.e. the start of the section to be copied.
Press the CAPTURE key.
The Zeta Three will display . . [------ MARK A ------]

Set Bars/Beats to 0009/01.1, the end of the section
to be copied.
Press the CAPTURE key.
The Zeta Three will display . . [------ MARK B ------]

Set Bars/Beats to 0017/01.1, the bar IN FRONT OF
WHICH we want to insert our copied section.

Enter the ".3 COPY A -> B TO X" menu via the "D01 EDIT" menu.

Press the CAPTURE key. Zeta displays . . [--- BARS COPIED ----]

Note:

1. Copying does not delete the bars copied from.

LEARNING TEMPOS

There are 4 distinct ways to learn tempos and enter them into the Tempo Map.

All 4 modes require a REFERENCE TIME against which beat lengths may be measured.

If the Zeta is in UNRESOLVED mode, then the selection of this reference time is made, with one exception, in exactly the same way that a Master is selected for the MIDI section to chase.

If the Zeta is in RESOLVED mode, then the reference time will actually be the frame rate being used for the resolving process (e.g. video sync). Starting times (hence Offsets) will however still be calculated relative to the unresolved mode reference time.

Here is a chart showing the (unresolved mode) REFERENCE TIME for each configuration of the ENABLE switches . .

------ ENABLE ------			
MASTER	SLAVE	MIDI	REFERENCE TIME
-	-	-	Master Time Code
M	-	-	Master Time Code
M	S	-	Master Time Code
M	-	D	Master Time Code
M	S	D	Master Time Code
-	S	-	Slave Time Code
-	S	D	Slave Time Code
-	-	D	GENERATOR Time Code

MODE 1. AVERAGED CAPTURE TAP . .

This mode is quite unlike the other three modes in that no tempos are actually entered into the Map (at least not automatically).

In this mode, the operator may tap a tempo on the CAPTURE key, and a tempo OVERRIDE will be set up, loaded with the average of the tapped tempo.

This tempo may subsequently be entered manually into the Map (see ENTERING A NEW TEMPO INTO THE MAP).

Example . .

Enter the ".1 AVRG CAPTURE TAP" menu via the "D03 LEARN MODE" menu. Press the CAPTURE key to select the mode. (Note: It is not necessary to establish the learning mode every time learning is to take place. The mode stays in effect until changed.)

Return to the "MAIN" Bar/Beat display.

Stop all tape transport and Map motion.

Disable Master and Slave, enable the MIDI section, using the ENABLE keys. (The Generator now provides the reference time.)

Enable the LEARN mode by pressing the Shift MIDI ENABLE key. The red LED in this key will flash until learning is commenced.

Tap the desired tempo at the CAPTURE key.

Notice that a tempo OVERRIDE has been set up, and the averaged tempo is being displayed on the Beats-per-Minute (or Frames-per-Beat) display. The red LED in the MIDI ENABLE key is lit solidly, indicating that the learn mode has begun.

To end the learn mode, simply stop tapping the CAPTURE key, the red LED in the MIDI key will extinguish shortly thereafter.

You now are left with a tempo OVERRIDE exactly as if you had just set it manually, only its value is the average of the tempo just tapped.

MODE 2. EXACT CAPTURE TAP . .

This mode appears similar to Mode 1, except that the tempos tapped at the CAPTURE key are entered DIRECTLY into the Tempo Map.

Example . .

We wish to manually tap a tempo starting from bar 9, beat 3 and finishing at the beginning of bar 11 (i.e. return to the original tempo at bar 11). This example assumes that bars 9 and 10 are both 4/4 and have a Click Length of a quarter note (04).

Enter the ".2 EXACT CAPTURE TAP" menu via the "D03 LEARN MODE" menu. Press the CAPTURE key to select the mode.

Return to the "MAIN" Bar/Beat display.

Stop all tape transport and Map motion.

Disable Master and Slave, enable the MIDI section, using the ENABLE keys. (The Generator now provides the reference time.)

Set the BAR/BEAT display to 0009/03.1 (our starting point).

Enable the LEARN mode by pressing the Shift MIDI ENABLE key. The red LED in this key will flash until learning is commenced.

Hit the Capture key 7 times at the desired tempo(s).

```
i.e.  Bar 9,  Beat 3
                   4
      Bar 10, Beat 1
                   2
                   3
                   4
      Bar 11, Beat 1
```

The final "hit" at the beginning of bar 11 is necessary for the Zeta to complete the measurement of Bar 10, beat 4.

Notice that, while the learn was in progress, the Bar/Beat display counted along with the CAPTURE key taps.

Also, if the display had been set to Beats-per-minute or Frames-per-beat, then it will have switched automatically to the Time Signature display for the duration of the learning process.

You may now step back through the Map and observe the new tempos which have been created.

Notes:

1. This type of learning follows the Time Signature and Click Length patterns already set up in the Tempo Map. If the Time Signatures in our example had been 3/4, then only 5 taps would have been required to take us to the beginning of bar 11. On the other hand, if the Time Signatures had remained at 4/4, but the Click Length had been changed to an 1/8th note, the 13 taps would have been required.

2. In this mode, the Map can be RUNNING, even running LOCKED, when learning is started.
   For example, it may be necessary to manually tap a tempo to match some action on a Master video. To do this, lock the MIDI section to the Master tape as usual, but have Learn Mode enabled (Shift MIDI) - the red MIDI key LED will be flashing.
   When the video (and the sequencer) arrives at the starting beat, simply commence tapping the CAPTURE key as before.
   The sequencer will be stopped during the learn sequence.

3. Learning commences from nearest Click boundary to the bar and beat on display when the CAPTURE key is tapped for the first time.

4. Learning finishes at the bar and beat that is on display when the CAPTURE key is hit for the last time.

MODE 3. MIDI CLOCK IN . .

This mode allows the Zeta to learn tempos from MIDI Clocks appearing at its MIDI In port.

Method . .

Create or load a sequence in your sequencer (or drum machine).

Connect the sequencer's MIDI Out to the Zeta Three MIDI In.

Enter the ".3 MIDI CLOCK IN" menu via the "D03 LEARN MODE" menu. Press the CAPTURE key to select the mode.

Return to the "MAIN" Bar/Beat display.

Stop all tape transport and Map motion.

Disable Master and Slave, enable the MIDI section, using the ENABLE keys. (The Generator now provides the reference time.)

Enable the LEARN mode by pressing the Shift MIDI ENABLE key. The red LED in this key will flash until learning is commenced.

Press the START key on the SEQUENCER.

Learning will begin from Bar 1, Beat 1 (unless the sequencer does not transmit a MIDI "Start" command - see Notes).

Learning will cease when a MIDI "Stop" command is received from the sequencer, OR IF MIDI LEARN IS DISABLED by the ENABLE keys.

Notes:

1. MIDI Clock In learning does not change any Time Signature or Click Length information already set up in the Tempo Map. A new tempo may be learned for each 1/16th note.

2. There are two kinds of MIDI start commands which may be issued by a sequencer - "Start" and "Continue". If "Start" is issued, then the Zeta will commence learning at Bar 1, Beat 1. If "Continue" is sent, then learning will commence at the 1/16 note being displayed when "Continue" is received.

   The Zeta is not currently recognising Song Pointer commands received at its MIDI In port.

MODE 4. AUX IN TIMEBASE . .

This mode allows the Zeta to learn tempos from a frequency appearing at its AUX IN jack. (Typically 96 cycles per beat from a non-MIDI device.)

Method . .

Connect the frequency source to the Zeta Three AUX IN.

Select a TIMEBASE FREQUENCY from the "D12 TIMEBASE" menus.

Enter the ".4 AUX IN TIMEBASE" menu via the "D03 LEARN MODE" menu. Press the CAPTURE key to select the mode.

Return to the "MAIN" Bar/Beat display.

Stop all tape transport and Map motion.

Disable Master and Slave, enable the MIDI section, using the ENABLE keys. (The Generator now provides the reference time.)

Enable the LEARN mode by pressing the Shift MIDI ENABLE key. The red LED in this key will flash until learning is commenced.

Start the frequency source.

Learning will commence at the 1/16 note being displayed when the Zeta receives its first frequency information at AUX IN.

Learning will cease when the input frequency stops, or if MIDI LEARN IS DISABLED by the ENABLE keys.

Notes:

1. TIMEBASE learning does not change any Time Signature or Click Length information already set up in the Tempo Map. A new tempo may be learned for each 1/16th note.

2. If the input frequency is running all the time, then learning will start as soon as LEARN is ENABLED.

COMMON CONSIDERATIONS FOR MODES 2, 3, AND 4 . .

1. Learning relative to the Generator is useful if tapes are being striped with Time Code at the same time as Tempo learning is taking place.
   If the Master and/or Slave tapes already have Time Code, then it would be advisable to learn all tempos relative to Master Code.

2. If learning does commence at Bar 1, Beat 1 (and only if), then the REFERENCE TIME CODE current when the first beat is received will be entered into the START TIME register, and the appropriate Offset will be calculated.

3. Learning may be ABORTED at any time by pressing CLEAR (Shift CAPTURE). No changes will have been made to the Map if this is done.

4. Learning may be STOPPED at any time by disabling the LEARN mode, i.e. by pressing Shift MIDI ENABLE.
   In this case, all tempos learned, up to the current beat, will be entered into the Map.

LEARN QUANTIZATION

The Zeta Three is always applying a quantization (averaging) routine to the learning process.

However, the DEGREE of quantization may be adjusted via the "D04 LEARN QNTIZE=" menu. When adjusted to its minimum ("MIN"), quantization is effectively disabled.

The trade-offs when using more or less quantization are as follows . .

If very little quantization is employed, then all the subtleties of the tempo to be measured will be captured.
HOWEVER, this will also use up the maximum amount of space in the Zeta's Tempo Map memory.

(It is also possible to capture "too much" subtlety ! For example, when learning from MIDI CLOCK IN, it is highly unlikely that a "constant" tempo created within a sequencer will appear quite so constant after it has been processed by the sequencer, transmitted on the MIDI line, and again processed by the Zeta. If minimum quantization is used for this learn mode, then the Zeta will likely record a series of very slightly fluctuating tempos - for example, 148.50, 148.64, 148.50, 148.64 and so on - thus needlessly using memory space for a tempo which should be specified ONCE as 148.57 !)

If MAXIMUM quantization is employed, then the resultant Map will clearly use a minimum of memory space, but only gross tempo changes will be recognized. All other changes will be averaged with the tempos coming before and after the change.

We have found so far that the default quantization value of "8" is good for MIDI CLOCK and TIMEBASE learning.

Higher values (up to MAX) are good for CAPTURE key tapping. In fact, learn mode 1 (AVERAGED CAPTURE TAP) always uses maximum quantization.

THE TIMEBASE OUTPUT

A controlled frequency, measured in cycles per beat, may be transmitted from the AUX OUT jack of the Zeta.

First, the 'Tip' of the AUX OUT jack must be assigned to the TIMEBASE generator via the Zeta System menu "Z10 XOUT TIP=TIMBASE".

Second, the desired TIMEBASE frequency must be selected in the MIDI menus "D12 TIMEBASE ->".

This frequency will be started and stopped at the same times as start and stop commands are transmitted from the MIDI Out port.

SAVING THE TEMPO MAP TO TAPE

To SAVE the current Tempo Map to tape, follow these steps . .

1. Connect the Zeta Three Generator output to a tape recorder input.
2. Enter the ".1 SAVE TO TAPE" menu via the MIDI menu "D13 MAP LOAD/SAVE ->".
3. Put the tape recorder into record on the appropriate track.
4. Press the CAPTURE key.
   The Zeta will display . . [---- SAVING MAP ----]
   And shortly after . . [--- MAP SAVED OK ---]
5. Press the CAPTURE key again
   - we recommend saving two copies.

Note: The Generator may be running or stopped while saving the Map. (Map information is stored in MIDI File Format in the Generator User Bits.)

The save may be aborted at any time by pressing CLEAR (Shift CAPTURE). Zeta displays . . [- MAP SAVE ABORTED -]

To LOAD a Tempo Map from tape . .

1. (a) Connect the output of a tape track containing the desired Map to the MASTER CODE Input, and ENABLE the Master ONLY.

OR (b) Connect the output of a tape track containing the desired Map to the SLAVE CODE Input, and ENABLE the Slave ONLY.

2. Enter the ".2 LOAD FROM TAPE" menu via the MIDI menu "D13 MAP LOAD/SAVE ->".

3. Press the CAPTURE key. ***
   Zeta displays . . [----- WAITING ------]

4. Put the tape machine into Play.
   When the Map Header arrives . . [--- LOADING MAP ----]
   When LOADing is complete . . [-- MAP LOADED OK ---]

Note: The load may be aborted at any time by pressing CLEAR (Shift CAPTURE). Zeta displays . . [- MAP LOAD ABORTED -]

*** IMPORTANT: As soon as the CAPTURE key is pressed to initiate a load from tape, ALL CONTENTS OF THE CURRENT MAP ARE DESTROYED.

MERGING

The Merge feature of the Zeta Three, if enabled, may be set to merge only selected types of MIDI messages. (See the "D06 MERGE=" display.)

1. "REAL TIME" - MIDI System Real Time messages which can be merged by the Zeta are . .

   Start
   Continue
   Stop
   Active Sensing

   Timing Clocks will only be merged if the Zeta itself is NOT issuing Clocks - i.e. if the "D07 MIDI OUT=" menu is set to "OFF".

2. "NO REAL TIME" - All messages received EXCEPT the above System Real Time messages will be merged.

3. "ALL" - merge all messages.
   The same restriction applies to MIDI Timing Clocks.

13 COMPUTER PORT [Adams-Smith Protocols]

[Software Revision 3.50]
[Document revised December 20, 1989]

13.1 INTRODUCTION

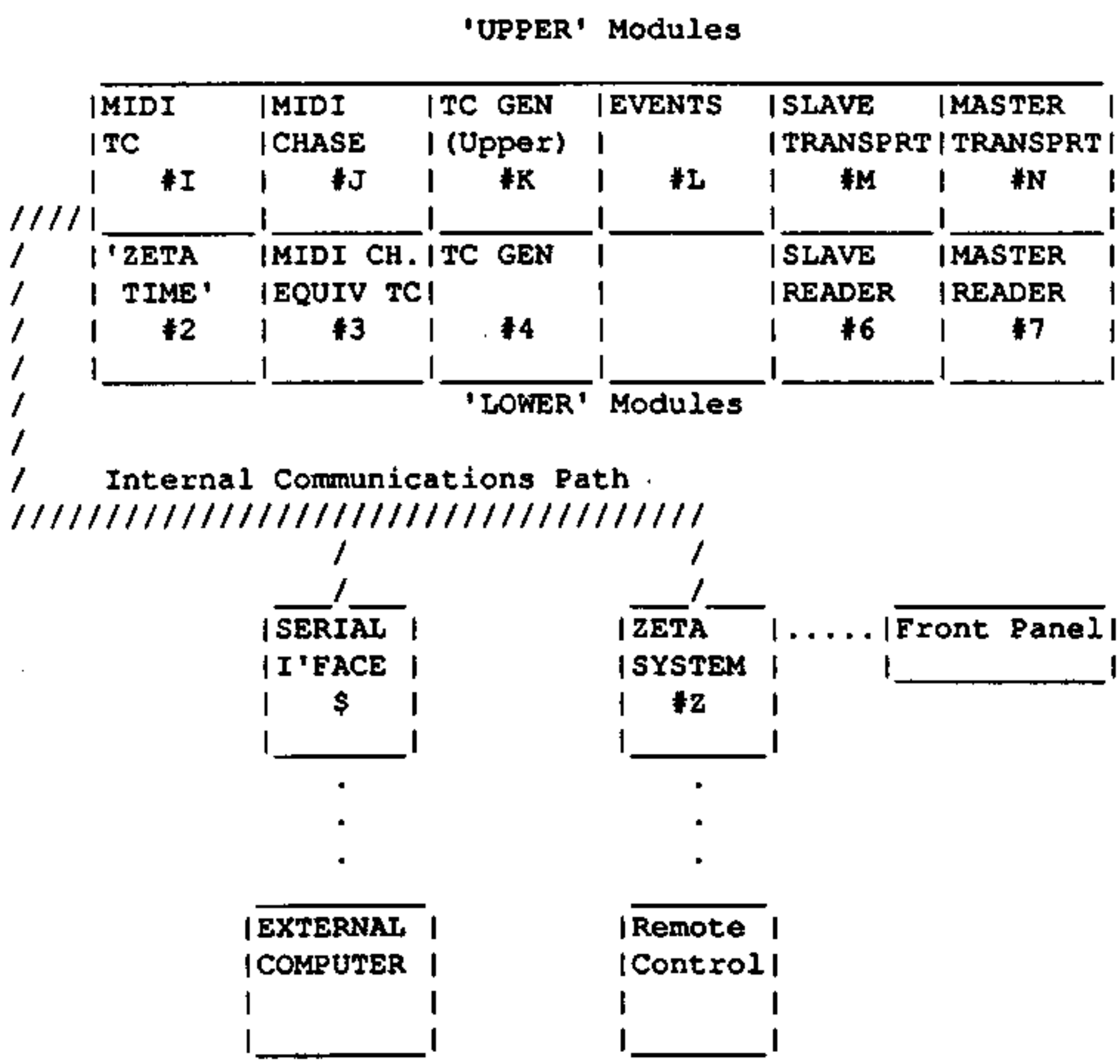
Commands sent to the Zeta III consist entirely of ASCII characters with no special formatting.

Responses from the Zeta III are also fully ASCII, but each response string is terminated by a Carriage Return / Line Feed pair for use with dumb terminals.

Communications are completely asynchronous, and require no handshaking. Use of X-ON and X-OFF is optional.

13.1.1 ZETA INTERNAL STRUCTURE

The Zeta appears via its Serial Interface as a collection of addressable software modules. There are currently 11 modules defined, plus a Zeta control module ("Z-module") and the Serial Interface module itself.



13.1.2 COMMANDS

In order to send commands to any particular module, the module must first be addressed. With the exception of the Serial Interface, this is done by sending a '#' character, followed by the address of the module. (Note: Single quotes are included in this text for clarity only, and are not to be sent as part of the command). The Serial Interface is addressed simply with the '$' character.

e.g. '#M' directs all subsequent commands to the Slave Transport module.
'$' directs all subsequent commands to the Serial Interface module.

The Zeta control module itself may also be addressed with '#Z', thus making available a number of time-saving system commands.

Module commands consist of ASCII command characters, some of which may be preceeded by an ASCII numeric string.

e.g. 'P' is a Play command when directed to either of the transports.

'01020304T' loads the time value 1 hour, 2 minutes, 3 seconds and 4 frames when directed to the Generator.

EXAMPLES

1.	Slave Transport Play command . .	#MP
2.	Master Transport Rewind command . .	#NR
3.	Load Generator Time Code . .	#401000000T
4.	Load Zeta System In Point . .	#Z2359592400I
5.	Load Serial Interface "Constant" #0 . .	$0033K

NOTES

1. Some commands, especially to the "Upper" modules (see diagram), take on different meanings depending upon whether the immediately preceding character was NUMERIC or not.

e.g. 'Q', when directed to either transport, is a "Cue to the previously loaded cue point" command, whereas
'0100000000Q' is the command which loads the cue point.

Consequently, to load a cue point of one hour and go to it, send the string '0100000000QQ'.

2. Generally, loading time code values to "Lower" modules requires the transmission of 8 digits (2 each for hours, minutes, seconds, and frames). However, loading to "Upper" modules, or to the "Z-module", requires 10 digits to be sent, the final two defining a subframe value (100ths), whether subframes are in fact used or not.

13.1.3 NUMERIC ENTRY AND THE "X" COMMAND

The Zeta Serial Interface contains a 14 digit internal numeric buffer into which all received numeric characters (hex 30 thru 39) are entered, calculator style.

Upon receipt of a subsequent command, the most recent numeric string will be handed on to the module to which the command is directed (including internal Serial Interface use).

There are other commands available which will fill the buffer with zeroes, fill it with 'blank' characters (hex F's), or backspace the entry by one position (see section 13.2.4 Serial Interface Command Set). Note that, for the sake of subsequent commands, these special numeric buffer commands are treated as 'numeric' entries.

However, one other command deserves serious attention, and that is the 'X' command.

Using ASCII characters only, it is difficult to enter hexadecimal numbers into the numeric buffer and not have them confused with alphabetic commands A thru F.

With one major exception, the character 'X' is used as a prefix to the letters A thru F if they are to be entered into the numeric buffer as hex values. The following rules must be adhered to:

For all modules other than the Serial Interface, the letter 'X' must precede any and every alphabetic character ('A' thru 'F') which is to be entered into the numeric buffer as a hexadecimal value.

For data directed internally to the Serial Interface, the letter 'X' must NOT precede the alphabetic characters 'A' thru 'F', as they will ALWAYS be interpreted as hexadecimal ('X' is in fact used by the Serial Interface as a data request command).

For example, to enter hex 3B to Master transport Constant #3, we must transmit '#M033XBK' (address 'M', constant #03, value 3B, enter constant command 'K').
On the other hand, to enter hex 3B to the Serial Interface Constant #1, we must transmit '$013BK'.

13.1.4 RESPONSES

The Zeta may be commanded to send data back to the External Computer.

It is first necessary to define where the data is to come from (i.e. from which module).

Next, the Serial Interface must be instructed to transmit data from one of three available data fields:

TIME
USER BITS
or
STATUS.

e.g. '#6$t' will return time code from the Slave Reader.

Although the actual format of the returned data may be selected by setting "Constants" (operating parameters) in the Serial Interface, the **default** format response to the above request will be :

'A HH:MM:SS:FF<CR><LF>'

where *<CR><LF>* represent Carriage Return and Line Feed respectively.

13.1.5 MULTIPLEXED REGISTERS

Modules at addresse 'J' thru 'N' (i.e. the 'Upper' set) multiplex the data available in their Time Code register.

e.g. The Slave Transport Synchronizer makes a number of registers available through the Time Code area. As these registers need to be called up ('attatched to the bus' in System 2600 terminology) before their data can be collected, each has been assigned a single call up letter:

'l'	LOCATE POINT
'm'	MASTER TIME CODE
'n'	END POINT
'q'	CUE POINT
't'	SELF TIME CODE
'|'	EVENTS TRIGGER SOURCE TIME ('|' = vertical line)
'e'	LOCK ERROR
'f'	ACTUAL OFFSET (slave - master)
'i'	IN POINT
'j'	OUT POINT
'o'	REQUESTED OFFSET
'0v'	EVENT 0 = IN POINT (same as 'i')
'1v'	EVENT 1 = OUT POINT (same as 'j')
<const #>'k'	CONSTANTS

The following request would cause the return of the Slave lock error register:

#Me$t

(Module 'M'; multiplexed register 'e'; address Serial Interface '$' for Time data request 't').

13.1.6 SYSTEM MASTERS

The concept of "System Master" is important to Zeta operation, **particularly when the Zeta Control module (#Z) is being addressed**. This is due to the fact that the Zeta controls three devices which may be synchronized in all possible combinations.

The "System Master" will most often be the Master transport, for example, when Slave is chasing Master, or MIDI and Slave are chasing the Master, or when the Master transport is simply enabled by itself.

However, if the Master transport is not enabled, and the MIDI device is chasing the Slave transport, then the Slave is said to be the current "System Master".

By the same token, if the MIDI chase section is enabled by itself, then it also becomes the System Master, even though no other devices are chasing it.

On the front panel ENABLE switches, the "System Master" is the LEFTMOST enabled device.

Device enables may be controlled via the Zeta control section by using the 'Z' command, which closely mirrors the function of the front panel ENABLE switches (see Section 13.11.16).

The current System Master may also be controlled via the Zeta control section using normal transport commands like Play, Stop, Rewind, Cue to GOTO point, etc. Of course, all chasing slaves will then follow, thus providing System control through the "System Master".

13.1.7 RECOMMENDED SERIAL INTERFACE SETUP

Many functions of both the Zeta Serial Interface and its internal software modules are included solely for backward compatability with the 2600 System.

In its default mode, the Interface is ideally set up for operation with a dumb terminal, and using your computer in this fashion is probably the best method of testing and getting the 'feel' of the Zeta commands and responses.

Dumb terminal mode therefore requires no special set up.

However, higher speed systems can benefit from other options within the Serial Interface.

In order to avoid time spent determining a configuration for the Serial Interface (i.e. how to set up its "Constants" or operating parameters), we include here a **recommended** setup.

Just send this string . .

$0289K0105K0033K

. . and you will invoke the following modes:

1. BREAK character recognition by the Serial Interface. .
2. Transmission of NAK character whenever receive errors are encountered in the Serial Interface.
3. Data transmission from the Zeta will continue even if there have been receive errors (default mode would be to halt transmission).
4. Transmission of the LINE FEED character at the end of every Zeta response will be inhibited (i.e. termination by Carriage Return only).
5. 'Extra status' will be included in all transmissions of status data (commands 'S','s').
6. All data will be transmitted in "COMPRESSED FORMAT - ALTERNATE MODE".
7. The Serial Interface will transmit X-ON and X-OFF to control data transmissions from the External Computer.
8. The Serial Interface will recognize X-ON and X-OFF protocols from the External Computer, and use them to control Response Transmissions.

Note: See also Sections 13.3, 13.4, 13.6, and 13.9.

13.1.8 HARDWARE CONSIDERATIONS

The choice of RS-422 or RS-232 is made with a connection and a jumper inside the Zeta III.

With the chassis cover removed, locate the ribbon cable originating at the 9-pin COMPUTER connector on the rear panel.

For RS-422 operation, the other end of this cable must be plugged into connector "J17 RS-422" on the main printed circuit board.
In addition, jumper JP3 must be positioned to connect the two of its three pins which are adjacent to connector J17.
i.e. Both the connector and the jumper are "towards the FRONT" of the cabinet.

For RS-232 operation, the ribbon cable must be plugged into connector "J16 RS-232".
Similarly, jumper JP3 must connect the two of its pins closest to J16.
i.e. Both the connector and the jumper "towards the REAR".

When using RS-422, pin connections are as follows:

1	FRAME GROUND	6	TRANSMIT Common
2	TRANSMIT "A"	7	TRANSMIT "B"
3	RECEIVE "B"	8	RECEIVE "A"
4	RECEIVE Common	9	FRAME GROUND
5	-		

NOTES

1. The "A" line of each balanced pair is inverted relative to UART inputs/outputs, similar to a regular RS-232 transmit/receive line.
   The "B" line is the non-inverted signal.

2. In order to achieve some degree of ground isolation, the RS-422 cable may be wired like a balanced audio cable.
   i.e. with ground/shield connection at one end of the line only.

When using RS-232, connections are as follows:

1	-	6	DATA SET READY (DSR)
2	RECEIVED DATA (RXD)	7	REQUEST TO SEND (RTS)
3	TRANSMIT DATA (TXD)	8	CLEAR TO SEND (CTS)
4	DATA TERMINAL READY (DTR)	9	-
5	SIGNAL GROUND		

NOTES

1. REQUEST TO SEND (Pin 7/Output) should be connected to the external computer's CLEAR TO SEND (CTS) input.
   When asserted, the RTS output indicates that the ZETA-THREE is ready to receive data (i.e. there is plenty of room in the ZETA's data buffer).
   If RTS is not asserted, then the external computer should refrain from further transmissions, as the ZETA-THREE buffer is about to overflow.

   If this line is not used, then the ZETA-THREE must be configured (Section 13.6) to transmit X-ON and X-OFF protocols to the external computer. (X-ON is transmitted at the time that the RTS line is asserted, X-OFF when it is de-asserted.)

2. CLEAR TO SEND (Pin 8/Input), when asserted enables the ZETA-THREE to transmit data. If not connected, then it will be asserted internally and the ZETA-THREE will transmit without restriction.

   The external computer may also control the ZETA-THREE's transmissions with X-ON and X-OFF protocols (the ZETA-THREE must first be enabled to receive them - see Section 13.6).

3. DATA TERMINAL READY (Pin 4/Output) is always asserted, as long as the ZETA-THREE is powered-up, and may be connected to the External Computer DSR (Data Set Ready) input.

4. DATA SET READY (Pin 6/Input), when asserted, enables the ZETA-THREE receive channel. If DSR is not connected, then it will be asserted internally.

13.2 CHARACTER SET RECOGNIZED BY SERIAL INTERFACE MODULE

NOTE: Throughout this text, square brackets are used to delimit variable command parameters, and must **not** be transmitted as part of the command (unless otherwise indicated).

13.2.1 IMMEDIATE CHARACTERS

Immediate characters are detected as soon as they are received, bypassing the receive buffer completely.

BREAK		RESET CHANNEL TO IDLE STATE The Serial Interface will only recognize a Break character if enabled to do so by via Constant 2. A Break character is defined as a spacing state of greater than one character length. The idle state resulting from a Break is very similar to the normal power up state - Receive and Transmit buffers are completely emptied, as is the Procedure stack - the 'last' device address is reset to 0, and the current data type to 'T'. The actual current address is the Serial Interface itself - all errors are cleared, all current status is reset. However, no Constants, Procedures or Groups are lost.
X-ON X-OFF	11h 13h	The Serial Interface can be set to recognize X-ON and X-OFF characters in the received data stream and to control its transmissions accordingly. This is set up by Constant 0, bit 0. NOTE: X-ON AND X-OFF MUST BE SINGLE CHARACTER COMMANDS, AND CANNOT BE SENT USING THE '@Q' OR '@S' REPRESENTATIONS.
NAK	15h	The NAK command is always recognized by the Serial Interface, and its effect is identical to the CTRL R (CLEAR ERRORS) command. The difference is that NAK does not have to be queued up in the receive buffer, and that it is not necessary to have addressed the Serial Interface ($). Also, no Wakeup message will be sent following a NAK command. NOTE: NAK MUST BE TRANSMITTED AS A SINGLE CHARACTER, AND CANNOT USE THE '@U' REPRESENTATION.

13.2.2 ENTRY ALTERING CHARACTERS

@ <char>	40h	CONVERTS CHARACTER TO CTRL CONVERTS CHARACTER TO LOWER CASE These two commands are always operational, and perform their conversions as the character is taken from the receive buffer. e.g. '@C' is equivalent to CTRL C (03hex) '}L' is equivalent to 'l'
} <char>	7Dh	

13.2.3 PRIVELIDGED COMMANDS

The following characters operate independantly of the current address pointer i.e. the address pointer set with '#', '%', '!' and '$' commands.

ESC	1Bh	ABORT CURRENT TRANSMISSIONS. Iterations ('I','J') are discontinued. Transmit buffer is emptied
! [a]	21h	SPECIFY DEVICE AS RESPONSE ONLY ADDRESS a : address This is almost identical to the '#' command, except that it sets up the current module address for RESPONSES ONLY (data collection), and does not affect the current command address.
# [a]	23h	SPECIFY DEVICE ADDRESS a : address The address indicated is set up both as the module address to which subsequent module commands should be sent, and as the module address from which data should be collected following a $S thru $Z (or $s through $z) command.
$	24h	SPECIFY SERIAL I'FACE AS COMMAND ADDRESS All commands will now be directed to the Serial Interface itself. Note however that many of these commands will direct the Serial Interface to collect and transmit data from the most recently addressed module. e.g. To collect timecode data from a module at address 2, send: #2$T
% [a]	25h	SPECIFY DEVICE AS COMMAND ONLY ADDRESS a : address This is almost identical to the '#' command, except that it sets up the current module address for COMMANDS ONLY, and does not affect the current data collection address.

& [p]	26h	EXECUTE PROCEDURE p : Procedure name See Procedure section.
^	5Eh	GROUP PREFIX (ONLY FOLLOWING '#','%','!') Instead of addressing a single module with the '#', '%' or '!' commands, a group may be addressed by transmitting . . # ^ <GROUP NAME> or % ^ <GROUP NAME> See Group section.
"	22h	reserved
@M <CR>	0Dh	reserved
'	27h	reserved
@G	07h	reserved

13.2.4 SERIAL INTERFACE COMMAND SET

These commands are unique to the Serial Interface, and the Serial Interface must first be addressed with the '$' command before they will be recognized.

COLLECT AND TRANSMIT MODULE DATA

S thru Z	53h- 5Ah	COLLECT AND TRANSMIT DATA FROM THE LAST ADDRESSED MODULE Each of these commands specifies that one type, or combination of types, of data be read from the module and sent to the External Computer in one of the various data formats described in the Data Format section. Data types are . . S Module status T Time code register U User bits register V Combined time code and user bits W Combined time code, user bits and a shortened version of status X Expanded time code, including subframes Y Combined time code and short status Z Combined user bits and short status The actual content of the data returned is dependant upon the module being interrogated. These data forms are described in section 13.10.

s thru z	73h- 7Ah	SAME AS S-Z
I	49h	ITERATE LAST TRANSMISSION The last data transmission will be repeated continuously until a 'Q' command is issued to the Serial Interface.
J	4Ah	ITERATE SYNC'D TO FRAME EDGES Same as 'I', but each transmission must be triggered by a fresh Master frame edge. This will often produce smoother displays, as adjacent transmissions of identical data are avoided.
Q	51h	QUIT ITERATION ('I' OR 'J')
L	4Ch	reserved

SERIAL INTERFACE INTERNAL DATA ACCESS

[ccvv] K	4Bh	ENTER SERIAL INTERFACE CONSTANTS cc = constant number vv = new value Similar to Synchronizer Constant entry. See Constants section.
e	65h	TRANSMIT SERIAL INTERFACE ERRORS AND STATUS Error and status bits are defined in the Errors and Status section.
k	6Bh	TRANSMIT SERIAL INTERFACE CONSTANTS See Constants section.
r	72h	TRANSMIT ZETA-THREE MODEL IDENTIFICATION AND SOFTWARE REVISION NUMBER The main body of this response will contain:

12 characters	*'ADAMS-SMITH'*	
8 characters	*'ZETA III'*	Model
4 characters	' '	Extension
4 characters	*'Rev'*	
4 characters	*'x.xx'*	Software

ERROR HANDLING

@R	12h	RESET ERRORS. All error bits are cleared. Transmission is resumed (if halted). A Wakeup message is sent if requested by Constant 0. See also the CTRL U / NAK command.

PROCEDURE DEFINITION AND READ BACK

P [p]	50h	READ BACK PROCEDURE DEFINITION p : Procedure name See Procedure section.
[	5Bh	BEGIN PROCEDURE DEFINITION See Procedure section.
]	5Dh	END PROCEDURE DEFINITION See Procedure section. Note: Square brackets constitute the actual commands here, and are not simply textual delimiters.

GROUP DEFINITION AND READ BACK

(	28h	BEGIN GROUP DEFINITION See Group section.
)	29h	END GROUP DEFINITION See Group section.
G [g]	47h	READ BACK GROUP DEFINITION g : Group name See Group section.

NUMERIC ENTRY TO NUMERIC INPUT BUFFER

@H	08h	BACKSPACE NUMERIC BUFFER.
0 thru 9	30h-39h	ENTER NUMERICS TO INPUT BUFFER
A thru F	41h-46h	ENTER HEX NUMERICS TO INPUT BUFFER **Note that the 'X' hex number prefix, while ALWAYS necessary for other modules, is never used by the Serial Interface. In fact, 'X' will produce expanded data from the most recently addressed module, so care should be exercised.**
_	5Fh	ZERO NUMERIC INPUT BUFFER
~	7Eh	BLANK NUMERIC INPUT BUFFER (Fill with Hex F's)
DEL	7Fh	BACKSPACE NUMERIC INPUT BUFFER

13.3 CHARACTER SET TRANSMITTED BY SERIAL INTERFACE

13.3.1 SERIAL CONTROL

X-ON	11h	
X-OFF	13h	The Serial Interface may be set to transmit these characters by Constant 0, bit 1. Note that Constant 0 provides separate control over sending and receiving X-ON,OFF. When enabled, X-OFF will be transmitted whenever the Serial Interface receive buffer contains over 150 characters, and X-ON will be issued when the contents drop below 100 characters. This corresponds to the disable/enable action on the RTS line of the RS-232 hook up. Note that once X-OFF or X-ON has been transmitted once, neither will be transmitted again until the buffer condition changes. Transmission of X-ON,OFF will not be disabled by the receipt of an X-OFF character from the other end of the serial link.

13.3.2 SPECIAL CHARACTERS

<BEAT>	04h	SERIAL METRONOME BEAT
<BAR>	05h	SERIAL METRONOME BEAT AT BAR LINE
<BEAT-PRE>	06h	SERIAL METRONOME BEAT BEFORE START OF SONG
<BAR-PRE>	07h	SERIAL METRONOME BEAT AT BAR LINE, BEFORE SONG These special characters, based on the progress of the MIDI section Tempo Map, are transmitted whenever enabled to do so by Serial Interface constant number 2.
NAK	15h	The Serial Interface may be requested, via Constant 2, to transmit a NAK character whenever a receive error occurs (viz. parity, UART buffer overrun, framing, Serial Interface receive buffer full). This NAK corresponds to the disabling of the DTR line in the RS-232 hook up. Like X-ON and X-OFF, NAK will not be inhibitted by the receipt of an X-OFF.
<FRAME>	1Ch	FRAME EDGE CHARACTER In similar fashion, the Serial Interface may be set to transmit a special character after every Master frame edge.

13.3.3 TERMINATORS

@J (LF)	0Ah	
@M (CR)	0Dh	All transmissions from the Serial Interface, with the exception of the above Serial Control and Special characters, will be terminated by the <LF><CR> pair. The Line Feed may however be disabled via Constant 1.

13.3.4 MESSAGE CHARACTER SET

space thru DEL	20h-7Fh	Serial Interface transmissions will contain only these standard ASCII characters except as outlined above.

13.4 DATA FORMATS

These symbols are used in the tables below . .

A	Module Address
r	Space if A = 0-7, register i.d. if A = G thru N, or Z
HH	Hours
MM	Minutes
SS	Seconds
FF	Frames
XX	Subframes
BB	User bits data
DD	Module status data
e	Serial Interface errors ident
k	Serial Interface constants ident
NN	Serial Interface internal data
xx	Software Revision number
^	Space (20h)

13.4.1 FULL FORMATS

These are the default formats of the Serial Interface.

Data Rqst	Data Format (^ indicates space)	Description
T	*A^HH:MM:SS:FF<CR><LF>*	Time
U	*A^BB^BB^BB^BB<CR><LF>*	User bits
S	*A^DD.DD.00.00<CR><LF>*	Status
S	*A^DD.DD.DD.DD<CR><LF>*	Status (Extended via Constant 1)
X	*A^HH:MM:SS:FF.XX<CR><LF>*	eXpanded, includes subframes
V	*A^HH:MM:SS:FF^^BB^BB^BB^BB<CR><LF>*	Combined Time and User Bits
W	*A^HH:MM:SS:FF^^BB^BB^BB^BB^^DD.DD<CR><LF>*	Combined Time, User Bits and short Status
Y	*A^HH:MM:SS:FF^^DD.DD<CR><LF>*	Combined Time and short Status
Z	*A^BB^BB^BB^BB^^DD.DD<CR><LF>*	Combined User Bits and short Status
Any	*$^NN^NN^NN^NN<CR><LF>*	Serial Interface internal data (see 13.5, 13.6)
Any	*$^ADAMS-SMITH^ZETA^III^^^^Rev^x.xx<CR><LF>*	Model and Software Revision

13.4.2 FULL FORMATS - ALTERNATE MODE

These formats may be invoked via Constant 0 MSD.

Data Rqst	Data Format (^ indicates space)	Description
T	*ArHH:MM:SS:FF<CR><LF>*	Time
U	*ArBB^BB^BB^BB<CR><LF>*	User bits
S	*ArDD.DD.00.00<CR><LF>*	Status
S	*ArDD.DD.DD.DD<CR><LF>*	Status (Extended via Constant 1)
X	*ArHH:MM:SS:FF.XX<CR><LF>*	eXpanded, includes subframes
V	*ArHH:MM:SS:FF^^BB^BB^BB^BB<CR><LF>*	Combined Time and User Bits
W	*ArHH:MM:SS:FF^^BB^BB^BB^BB^^DD.DD<CR><LF>*	'Combined Time, User Bits and short Status
Y	*ArHH:MM:SS:FF^^DD.DD<CR><LF>*	Combined Time and short Status
Z	*ArBB^BB^BB^BB^^DD.DD<CR><LF>*	Combined User Bits and short Status
e	*$eNN^NN^NN^NN<CR><LF>*	Errors and status (see 13.5)
k	*$kNN^NN^NN^NN<CR><LF>*	Constants (see 13.6)
Any	*$rADAMS-SMITH^ZETA^III^^^^Rev^x.xx<CR><LF>*	Model and Software Revision

13.4.3 COMPRESSED FORMATS

These formats may be requested via Constant 0 MSD.

Data Rqst	Data Format	Description
T	*HHMMSSFF<CR><LF>*	Time
U	*BBBBBBBB<CR><LF>*	User bits
S	*DDDD0000<CR><LF>*	Status
S	*DDDDDDDD<CR><LF>*	Status (Extended via Constant 1)
X	*HHMMSSFFXX<CR><LF>*	eXpanded, includes subframes
V	*HHMMSSFFBBBBBBBB<CR><LF>*	Combined Time and User Bits
W	*HHMMSSFFBBBBBBBBDDDD<CR><LF>*	Combined Time, User Bits and short Status
Y	*HHMMSSFFDDDD<CR><LF>*	Combined Time and short Status
Z	*BBBBBBBBDDDD<CR><LF>*	Combined User Bits and short Status
Any	*NNNNNNNN<CR><LF>*	Serial Interface internal data (see 13.5, 13.6)
Any	*ADAMS-SMITH^ZETA^III^^^^Rev^x.xx<CR><LF>*	Model and Software Revision

13.4.4 COMPRESSED FORMATS - ALTERNATE MODE

These formats are invoked when both Compressed and Alternate bits are set in Constant 0 MSD.

Data Rqst	Data Format (^ indicates space)	Description
T	*ArTHHMMSSFF<CR><LF>*	Time
U	*ArUBBBBBBBB<CR><LF>*	User bits
S	*ArSDDDD0000<CR><LF>*	Status
S	*ArSDDDDDDDD<CR><LF>*	Status (Extended via Constant 1)
X	*ArXHHMMSSFFXX<CR><LF>*	eXpanded, includes subframes
V	*ArVHHMMSSFFBBBBBBBB<CR><LF>*	Combined Time and User Bits
W	*ArWHHMMSSFFBBBBBBBBDDDD<CR><LF>*	Combined Time, User Bits and short Status
Y	*ArYHHMMSSFFDDDD<CR><LF>*	Combined Time and short Status
Z	*ArZBBBBBBBBDDDD<CR><LF>*	Combined User Bits and short Status
e	*$e^NNNNNNNN<CR><LF>*	Errors and status (see 13.5)
k	*$k^NNNNNNNN<CR><LF>*	Constants (see 13.6)
Any	*$r^ADAMS-SMITH^ZETA^III^^^^Rev^x.xx<CR><LF>*	Model and Software Revision

13.4.5 WAKE-UP MESSAGE FORMAT

This message will be transmitted after each power up sequence, and after each CTRL R error reset, if so enabled by Constant 0 LSD. The message format is identical to the main body of the MODEL AND SOFTWARE REVISION response.

ADAMS-SMITH^ZETA^III^^^^Rev^x.xx<CR><LF>

13.5 SERIAL INTERFACE ERRORS AND STATUS

The current status and error status of the Serial Interface itself may be examined through the use of the 'e' command. Eight digits are returned, beginning with digit 7 and ending with digit 0. Only the middle 4 digits, 2 through 5, are used.

At the time when a serial receive error occurs, the Serial Interface will save the error status incurred in a 'latch' register, which may later be read regardless of time intervening between the error and the status read back. Note that once an error has been 'latched', no further latching will occur on the detection of subsequent errors. In other words, only the first error is latched and held until status is read. A 'latched' bit may be found in digit 2 of the status read back. If this is set, then the status read relates to some previous error condition. After this latched status has been read, the bit will be reset, allowing current status to be accessed. Digits 2 and 3 are, furthermore, both completely reset to zero after any read of current status.

Digit 5 : Serial Receive Errors - MSB

8 bit : 1=Receive buffer is full - characters will be lost.
4 bit : 1=Receive framing error has occured.
2 bit : 1=Receive over-run error, viz new character arrived before previous character(s) had cleared the UART.
1 bit : 1=Receive parity error has occured.

Digit 4 : Serial Receive Errors - LSB

8 bit : 0
4 bit : 0
2 bit : 1=Receive buffer filling, RTS disabled (RS-232), X-OFF transmitted if enabled.
1 bit : 0

Digit 3 : Temporary Status (cleared after each read) - MSB

8 bit : 1=Serial Interface has been reset since this status last read.
4 bit : 1=Group block overflow.
2 bit : 0
1 bit : 1=Attempt to address undefined Group.

Digit 2 : Temporary Status (cleared after each read) - LSB

8 bit : 1=This status read reflects the status incurred by the first serial receive error which has been detected since status was last read. The next status read will show current status, with this bit reset.
4 bit : 1=Procedure block overflow.
2 bit : 1=Illegal character in Procedure definition.
1 bit : 1=Call to undefined procedure.

13.6 SERIAL INTERFACE CONSTANTS

Serial Interface operation is controlled via stored (non-volatile) constants which may be set and read directly via the serial line.

There are only four constants, and they may be read back, without being addressed, with the command 'k'. Constants are set with command 'K', which will take the last four digits entered to the numeric input buffer, use the first two as a Constant number, and the second (most recent) two as the value to load.
e.g. $0140K loads 40 hex to Constant number 1.

Constants are read back in the order 3,2,1,0 with the most significant digit (MSD) preceding the least significant (LSD).

Constant 3: currently unused.

Constant 2 MSD:

8 bit : 1=Enable BREAK character recognition by the Serial Interface.
4 bit : 0
2 bit : 1=Enable automatic transmission of Special Frame Edge character at every "System Master" frame edge.
1 bit : 1=Enable transmission of Special Serial Metronome characters (see 13.3.2).

Constant 2 LSD:

8 bit : 1=Enable transmission of NAK character whenever receive errors are encountered. See Error Responses section.
4 bit : 0
2 bit : 0
1 bit : 1=Enable data transmission to continue despite receive errors. See Error Responses section.

Constant 1 MSD:

8 bit : 0
4 bit : 0
2 bit : 0
1 bit : 0

Constant 1 LSD:

8 bit : 0
4 bit : 1=Include 'extra status' in all transmissions of status data (commands 'S','s'). This will replace the last four digits, normally all zeroes, with further bus status.
2 bit : 0
1 bit : 1=Disable transmission of LINE FEED character.

Constant 0 MSD:

8 bit : 0
4 bit : 0
2 bit : 1=Transmit all data in COMPRESSED mode.
(See Data Format section.)
1 bit : 1=Transmit all data in ALTERNATE mode.
(See Data Format section.)

Constant 0 LSD:

8 bit : 0
4 bit : 1=Send Wakeup message after each power up or CTRL R reset.
2 bit : 1=Enable the Serial Interface to transmit X-ON and X-OFF to control data transmissions from the External Computer.
1 bit : 1=Enable Serial Interface reception of X-ON, X-OFF protocols to control its transmitter.

13.7 PROCEDURES

A Procedure is a command sequence transmitted to the Serial Interface to be stored internally and executed on demand by transmission of a Procedure execution command.

13.7.1 PROCEDURE DEFINITION

The format for Procedure definition is as follows . .

[<Procedure name> <Any command string>]

Procedure name	may be any ASCII character with the exception of NULL (00 hex). However, only the least significant 6 bits will be used (63 names). (Thus CTRL A and 'A' will be equivalent, as will '0' and 'p' etc.)
Any command string	means what it says with the exception of the character '['. i.e. Procedure definitions may not be nested. However, a Procedure may contain execution calls to any other Procedure(s). If '[' is encountered, the illegal Procedure character flag in Error/Status digit #2 will be set, and the character will be ignored.

There exists also the possibility that the new Procedure definition will not fit into the space remaining in the internal Procedure block. If this should occur, the new string will not be entered at all, and the Procedure overflow bit will be set in Error/Status digit #2.

13.7.2 PROCEDURE DELETION

Any single Procedure may be deleted with the commands:

[<Procedure name>]

All procedures may be deleted using as ASCII Null:

[<ASCII NULL>]

<ASCII NULL> is of course 00 hex, which may also be entered manually with '@ <space>', or '@@'.
e.g. [@@] will delete all Procedures.

13.7.3 PROCEDURE EXECUTION

Procedures are executed with the string . .

& <Procedure name>

Procedures may be nested to a level of 17. That is, a Procedure may call another Procedure which may call another and so on. Any Procedure calls beyond a nesting level of 17 will simply be ignored.

If an execution call is made to an undefined Procedure, then the Procedure error bit will be set in Error/Status digit #2, and the call will be ignored.

13.7.4 PROCEDURE READ BACK

The contents of a Procedure definition may be transmitted back to the External Computer via the following command . .

P <Procedure name>

The string returned will correspond exactly to the string used to define the Procedure in the first place, including the square brackets. However, all Control characters (viz less than 20 hex) will be replaced with an '@' <upper case> pair.

Also, if the definition is greater then 100 characters long, only the first 100 characters will be transmitted, and the closing square bracket will be dropped to indicate the truncation.

If the Procedure is in fact undefined, then only the square brackets will be returned.

i.e.

Normal Procedure:	[<Name> <Command string>]
Long Procedure:	[<Name> <Truncated command string>
Undefined Procedure:	[]

13.8 GROUPS

Modules may be gathered into Groups for addressing purposes, and commands may thus be directed to more than one module. In addition, data may be collected from a Group of modules, and transmitted as if separate data requests had been issued.

Groups are defined and stored in very much the same manner as Procedures.

13.8.1 GROUP DEFINITION

The format for Group definition is as follows . .

(<GROUP NAME> <Any module addresses>)

Group name	may be any ASCII character with the exception of NULL (00 hex). However, only the least significant 6 bits will be used (63 names). (Thus CTRL A and 'A' will be equivalent, as will '0' and 'p' etc.)
Any module addresses	are standard module addresses, as used in the '#','%' and '!' commands, and must be entered one immediately following the other.

Note that the list of module addresses cannot contain other Group names, and thus that Groups may not be nested.

Just like Procedures, if a new Group definition will not fit into the space remaining in the internal Group definition block, the new Group will not be entered at all, and the Group overflow bit will be set in Error/Status digit #3.

13.8.2 GROUP DELETION

Any single Group may be deleted with the commands:

(<Group name>)

All groups may be deleted using an ASCII Null:

(<ASCII NULL>)

<ASCII NULL> is of course 00 hex, which may also be entered manually with '@ <space>', or '@@'.
e.g. (@@) will delete all Groups.

13.8.3 GROUP ADDRESSING

Groups are addressed with the string . .

^ <Group name>
or
% ^ <Group name>
or
! ^ <Group name>

If an attempt is made to address an undefined Group, then the Group error bit will be set in Error/Status digit #3. This will take place both at the time the address is called for (at the time of the '#', '%' or '!' command), and any time an attempt is made to subsequently send a command to, or collect data from, that undefined Group of modules.

Note: When collecting and transmitting data from a Group of modules, the Serial Interface makes no attempt to guarantee that every data request is satisfied for every member of the Group, as it does for individual module data requests. Rather, to avoid seriously overloading the output buffer, the Group data requests are treated much like Iteration requests - when there is room in the output buffer, the data for the next member of the Group is assembled and transmitted. If a second data request (S-Z command etc.) is made for the same Group, before completion of the previous transmission for the Group, then the new request will simply overwrite the old request for the modules whose first request data has not yet been transmitted.

13.8.4 GROUP READ BACK

The contents of a Group definition may be transmitted back to the External Computer via the following command . .

G <Group name>

The string returned will correspond exactly to the string used to define the Group in the first place, including the parentheses. However, all Control characters (viz those less than 20 hex) will be replaced with an '@' <upper case> pair.

Also, if the definition is greater then 100 characters long, only the first 100 characters will be transmitted, and the closing parenthesis will be dropped to indicate the truncation.

If the Group is in fact undefined, then only the parentheses will be returned.

i.e.

Normal Group:	(<Name> <Addresses>)
Long Group:	(<Name> <Truncated addresses>
Undefined Group:	()

13.9 ERROR RESPONSES

The default response to receive errors (Parity errors, UART receiver overrun, Framing errors, and Serial Interface receive buffer full errors) is simply to **halt all transmissions** from the Serial Interface. The only response to this condition is to send a $ <CTRL R> sequence to clear the error state and resume transmission.

Some other alternatives are provided:

1. The NAK character can be enabled as an error indicator transmitted **from** the Serial Interface. Having received a NAK, the External Computer may request an Error/Status report from the Serial Interface, which will indicate the error condition, which has in fact been latched at the time of the error. (See 13.5 Serial Interface Errors and Status.)

2. The NAK character may also be transmitted **to** the Serial Interface to clear errors, in a similar fashion to the CTRL R command. However, the NAK command completely bypasses the receive buffer, and does not require the Serial Interface to be currently addressed via the '$' command, and so is more immediate and also guaranteed to clear even buffer full conditions, although the state of the buffer of course remains unchanged until the Serial Interface resumes command processing.
Note: Neither the NAK character nor CTRL R will reset the 'latched' copy of the error status which has been incurred.

3. In addition, a BREAK character may be recognized by the Serial Interface, if enabled by Constant 2 MSD, in which case, all errors will be cleared and the Serial Interface channel will return to a state very similar to the normal power up condition. Most importantly, both the receive and transmit buffers will be completely flushed.

4. The response of halting Serial Interface transmissions after detecting a receive error may be over-ridden by setting the appropriate bit in Constant 2 LSD. This assumes that the External Computer is going to use some other means (NAK !) to detect that an Serial Interface receive error has occurred.

13.10 CONTENT OF DATA READ BACK FROM ZETA-THREE

Note: SYSTEM 2600 EQUIVALENT ADDRESSES . .

'LOWER' Modules		'UPPER' Modules	
0	reserved	G	reserved
1	reserved	H	reserved
2	ZETATIME	I	MIDI TIME CODE GENERATOR
3	MIDI CHASE EQUIV. TIME	J	MIDI CHASE
4	GENERATOR	K	GENERATOR Upper
5	reserved	L	EVENTS
6	SLAVE READER	M	SLAVE TRANSPORT
7	MASTER READER	N	MASTER TRANSPORT
		Z	ZETA SYSTEM

EXAMPLES

To request data from a 'lower' module, first the module must be specified, then the Interface section must be told to transmit the appropriate data.
e.g.'#7$T' causes the transmission of time data from the Master Reader. '#4$S' produces Status information from the Generator.

Requesting data from an 'upper' module may also require that a register is specified, as many registers may be multiplexed into the single time code area.

e.g.'#Me$X' returns 10 digits (eXtended data) of lock error information from the Slave Synchronizer.
In this case, the 'e' (error) register becomes the CURRENTLY ATTACHED REGISTER, which may also be referred to by the
Trap and Increment/Decrement commands.

'#M$U' on the other hand, returns the 'user bits' area from the Slave device, 6 digits of which are never
multiplexed. Note that the 'user bits' area of the 'upper' modules is used as a general purpose status area.

DATA FORMATTING CODES USED IN THIS TEXT

Note: Upper case denotes BCD digits.
Lower case denotes individual bits.

TIME

	HH	:	*MM*	:	*SS*	:	*FF*

USER BITS

	BB	*BB*	*BB*	*BB*
or	*aaaabbbb*	*ccccdddd*	*eeeeffff*	*gggghhhh*

STATUS

	DD .	*DD* .	*DD* .	*DD*
or	*ssssttt*t.	*uuuuvvvv*.	*wwwwxxxx*.	*yyyyzzzz*

13.10.2 ZETATIME (address 2)

See 13.10.6 READERS

13.10.3 MIDI CHASE EQUIVALENT TIME (address 3)

TIME

HHMMSSFF	Equivalent Time Code

USER BITS

aaaa-dddd	4 digit current bar number
eeeeffff	Current beat number (within bar)
gggghhhh	Current 1/16 note number (within beat)

STATUS

ssss	0
tttt	0
uuuu	Requested time code mode . . 0 = 30fps 4 = 24fps 5 = 25fps
vvvv	0
wwww-zzzz	not used

13.10.4 GENERATOR (Lower: address 4)

TIME	
HHMMSSFF	Generated Time Code
USER BITS	
aaaa-hhhh	Generated User Bits
STATUS	
ssss	0
tttt	bit 8
	bit 4 = generated drop frame bit
	bit 2 = generated user bits group flag 1
	bit 1 = generated user bits group flag 0
uuuu	Requested time code mode . .
	0 = 30fps
	4 = 24fps
	5 = 25fps
vvvv	bit 8 = counting (not jamming)
	bit 4
	bit 2
	bit 1 = frame locked
wwww	bit 8 = data dump buffer is full
	bit 4 = data dump to user bits in progress
	bit 2 = frame locking to current Master (else crystal)
	bit 1 = generator incrementing
xxxx	bit 8 = counting inhibited by stopped source (jam) code
	bit 4 = time jamming
	bit 2 = user bits jamming
	bit 1 = reverse direction (generated time code)
yyyyzzzz	not used

13.10.6 READERS (Slave at address 6, Master at address 7)

Note: ZETATIME (address 2) also follows this format.

TIME

HHMMSSFF	Time Code from tape

USER BITS

aaaa-hhhh	User Bits from tape

STATUS

ssss	bit 8
	bit 4
	bit 2
	bit 1 = tach updating (will be 0Ch ?)
tttt	bit 8 = color frame bit from time code
	bit 4 = drop frame bit .. .. ..
	bit 2 = user bits group flag 1 .. .. ..
	bit 1 = user bits group flag 0 .. .. ..
uuuu	Requested time code mode . .
	0 = 30fps
	4 = 24fps
	5 = 25fps
vvvv	Time code mode from tape . .
	0 = 30fps (not = Requested mode)
	4 = 24fps (.. .. ..)
	5 = 25fps (.. .. ..)
	F = same as Requested time code mode
wwww	bit 8 = reverse direction (tach update)
	bit 4
	bit 2 = no tach
	bit 1 = valid (current) time code
xxxx	bit 8 = time code at sampling speeds (above approx 2x play)
	bit 4 = tach updating
	bit 2 = tach rate and direction have been learnt
	bit 1 = reverse direction (time code)
yyyyzzzz	not used

13.10.11 MIDI CHASE (address J)

TIME

HHMMSSFF	MULTIPLEXED DATA Requested with Lower Case ident's :

8 digit timecode
(User Bits *gggghhhh* = 00)

'm'	[6Dh]	MASTER TC
'q'	[71h]	CUE POINT (time code)
't'	[74h]	EQUIVALENT SELF TC

10 digit timecode
(Subframes in User Bits *gggghhhh*)

'e'	[65h]	ERROR
'f'	[66h]	ACTUAL OFFSET (slave - master)
'o'	[6Fh]	REQUESTED OFFSET
's'	[73h]	START TIME

Bar/Beat

'a'	[61h]	MARK A
'b'	[62h]	MARK B
'r'	[72h]	BAR/BEAT CUE POINT
'u'	[75h]	CURRENT BAR/BEAT
'x'	[78h]	MARK X

HHMM	4 digit bar number
SS	Beat number (within bar)
FF	1/16 note number (within beat)
gggghhhh	00

Special

'0v'	[30h,76h]	Beats per Minute

HHM	000
MSS	BPM
FF	BPM fraction (/100)
gggg	bit 8 = Click override bit 4 = Time Signature override bit 2 = Frames per Beat override bit 1 = Beats per Minute override
hhhh	0

'1v'	[31h,76h]	Frames per Beat

HHM	000
MS	FPB
SFF	FPB fraction (/1000)
gggg	see Beats per Minute
hhhh	1

'2v'	[32h,76h]	Time Signature
	HH	00 = Normal Click value 01 = Dotted Click value
	MM	Click value
	SS	Time Signature numerator
	FF	Time Signature denominator
	gggg	see Beats per Minute
	hhhh	2
'3v'	[33h,76h]	Metronome Click value
	HHMMSSFF	see Time Signature
	gggg	see Beats per Minute
	hhhh	3

Constants

<const #>'k' [??h,??h,6Bh]

HH	requested constant + 3
MM	requested constant + 2
SS	requested constant + 1
FF	requested constant
ggghhhh	# of requested constant

USER BITS

aaaabbbb	ASCII tally of current pseudo-transport motion . . 'P'=play 'S'=stop 'F'=fast forward 'R'=rewind
ccccdddd	ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.
eeee	bit 8 = no cue point loaded bit 4 = 0 bit 2 = not slewing bit 1 = not cueing
ffff	bit 8 = not locked bit 4 = disabled bit 2 = parked bit 1 = same as bit 2

STATUS

ssss	Currently attached bank number (0 thru 6)
tttt	Number of available map banks (1,3,4 or 7)
uuuu	bit 8 = not tight locked (narrow window) bit 4 = 1 bit 2 = resolve mode bit 1 = chase mode
vvvv	bit 8 = 0 bit 4 = 0 bit 2 = map positioned after end of song bit 1 = map positioned before beginning of song
wwwwxxxx	ASCII tally of most recent transport command ('P'=play, etc.)
yyyyzzzz	Most recent map edit error code

13.10.12 GENERATOR (Upper: address K)

TIME

HHMMSSFF	MULTIPLEXED DATA Requested with Lower Case ident's . .

8 digit timecode
(User Bits *gggghhhh* = 00)

'r'	[72h]	PRESET REGISTER
't'	[74h]	GENERATED TC
'u'	[75h]	GENERATED USER BITS
'y'	[79h]	TC JAM SOURCE CODE
'z'	[7Ah]	USER BITS JAM SOURCE CODE

Constants

<const #>'k' [??h,??h,6Bh]

HH	requested constant + 3
MM	requested constant + 2
SS	requested constant + 1
FF	requested constant
gggghhhh	# of requested constant

USER BITS

aaaabbbb	ASCII tally of current generator action 'P'=run 'S'=stop 'R'=run reverse
ccccdddd	ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.
eeee	bit 8 = 1 bit 4 = 0 bit 2 = 1 bit 1 = 1
ffff	bit 8 = not frame locked bit 4 = master disabled when gen=master only bit 2 = stopped bit 1 = same as bit 2

STATUS

ssssttt	not used
uuuu	bit 8 = not frame locked bit 4 = 0 bit 2 = system resolve mode bit 1 = generator frame lock mode
vvvv-zzzz	not used

13.10.13 EVENTS (address L)

TIME

HHMMSSFF MULTIPLEXED DATA
Requested with Lower Case ident's

10 digit timecode
(Subframes in User Bits *gggghhhh*)
n'v' [3?h,76h] EVENT n

Special
'w' [77h] EVENTS STATUS

HH	event 9, event 8
MM	event 7, event 6
SS	event 5, event 4
FF	event 3, event 2
gggg	event 1
hhhh	event 0

For each digit
bit 8 = event tripped
bit 4 = event armed
bit 2 = continuous output active
bit 1 = momentary output active

	bit 8	bit 4
At arming	0	1
After trip time	1	1
After disarm	x	0 (No change)

Note 1. Momentary output may not correspond to continuous output if disarm state is inverted.
2. Continuous output always reset at arming time.
3. Momentary output unaffected by arming.

Constants
<const #>'k' [??h,??h,6Bh]

HH	requested constant + 3
MM	requested constant + 2
SS	requested constant + 1
FF	requested constant
gggghhhh	# of requested constant

USER BITS

aaaabbbb	ASCII number for event currently 'attached' (relevant to 'v' register request)
ccccdddd	ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.
eeee	bit 8 = 1 bit 4 = 0 bit 2 = 1 bit 1 = 1
ffff	0

STATUS

ssss	0
tttt	0
uuuu	bit 8 = 1 bit 4 = 1 bit 2 = 0 bit 1 = 0
vvvv	0
wwww-zzzz	not used

13.10.14 SLAVE TRANSPORT (address M)

TIME
HHMMSSFF MULTIPLEXED DATA
Requested with Lower Case ident's :

8 digit timecode
(User Bits *gggghhhh* = 00) . .

'l'	[6Ch]	LOCATE POINT
'm'	[6Dh]	MASTER TC
'n'	[6Eh]	END POINT
'q'	[71h]	CUE POINT
't'	[74h]	SELF TC
'|'	[7Ch]	EVENTS TRIGGER SOURCE

10 digit timecode
(subframes in User Bits *gggghhhh*)

'e'	[65h]	ERROR
'f'	[66h]	ACTUAL OFFSET (slave - master)
'i'	[69h]	IN POINT
'j'	[6Ah]	OUT POINT
'o'	[6Fh]	REQUESTED OFFSET
'0v'	[30h,76h]	EVENT 0 = IN POINT (same as 'i')
'1v'	[31h,76h]	EVENT 1 = OUT POINT (same as 'j')

Constants
<const #>'k' [??h,??h,6Bh]

HH	requested constant + 3
MM	requested constant + 2
SS	requested constant + 1
FF	requested constant
gggghhhh	# of requested constant

Transport Identification: 'x' [78h]
(User Bits *gggghhhh* = 00)

HHMM		0000
SSFF		0ddd=Transport code (ddd=000 thru 999)
	or	100n=User area code (n=0 thru 9)

USER BITS

aaaabbbb	ASCII tally of current transport motion 'P'=play 'S'=stop 'F'=fast forward 'R'=rewind
ccccdddd	ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.
eeee	bit 8 = no cue point loaded bit 4 = rolling back bit 2 = not slewing bit 1 = not cueing
ffff	bit 8 = not locked bit 4 = disabled bit 2 = parked bit 1 = same as bit 2

STATUS

ssss	bit 8 = record out armed bit 4 = record in armed bit 2 = rehearse out armed bit 1 = rehearse in armed Note: In and Out always armed/disarmed together.
tttt	bit 8 = record completion bit 4 = record status bit 2 = rehearse completion bit 1 = rehearse status

	bit 8	bit 4
At arming	0	0
During Rec/Reh	0	1
After punch out	1	0
After disarm	x	x (No change)

uuuu	bit 8 = not tight locked (narrow window) bit 4 = not lock hold bit 2 = resolve mode bit 1 = chase mode
vvvv	bit 8 = external tallies in use (via constant) bit 4 = external record tally active bit 2 = external 'spare' tally active bit 1 = not used
wwwwxxxx	ASCII tally of most recent transport command ('P'=play, etc.)
yyyy	not used
zzzz	bit 8 = 0 bit 4 = 0 bit 2 = 0 bit 1 = over end point

13.10.15 MASTER TRANSPORT (address N)

TIME

HHMMSSFF MULTIPLEXED DATA
Requested with Lower Case ident's :

8 digit timecode
(User Bits *gggghhhh* = 00) . .

'n'	[6Eh]	END POINT
'q'	[71h]	CUE POINT
't'	[74h]	SELF TC

10 digit timecode
(subframes in User Bits *gggghhhh*)

'e'	[65h]	ERROR (usually distance from cue point)
'i'	[69h]	IN POINT
'j'	[6Ah]	OUT POINT
'0v'	[30h,76h]	EVENT 0 = IN POINT (same as 'i')
'1v'	[31h,76h]	EVENT 1 = OUT POINT (same as 'j')

Constants

<const #>'k' [??h,??h,6Bh]

HH	requested constant + 3
MM	requested constant + 2
SS	requested constant + 1
FF	requested constant
gggghhhh	# of requested constant

Transport Identification: 'x' [78h]
(User Bits *gggghhhh* = 00)

HHMM		0000
SSFF		0ddd=Transport code (ddd=000 thru 999)
	or	100n=User area code (n=0 thru 9)

USER BITS

aaaabbbb ASCII tally of current transport motion
'P'=play
'S'=stop
'F'=fast forward
'R'=rewind

ccccdddd ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.

eeee
bit 8 = no cue point loaded
bit 4 = rolling back
bit 2 = 1
bit 1 = not cueing

ffff
bit 8 = 1
bit 4 = disabled
bit 2 = parked
bit 1 = same as bit 2

STATUS

ssss

bit 8 = record out armed
bit 4 = record in armed
bit 2 = rehearse out armed
bit 1 = rehearse in armed
Note: In and Out always armed/disarmed together.

tttt

bit 8 = record completion
bit 4 = record status
bit 2 = rehearse completion
bit 1 = rehearse status

	bit 8	bit 4
At arming	0	0
During Rec/Reh	0	1
After punch out	1	0
After disarm	x	x (No change)

uuuu

bit 8 = 1
bit 4 = 1
bit 2 = 0
bit 1 = 0

vvvv

bit 8 = external tallies in use (via constant)
bit 4 = external record tally active
bit 2 = external 'spare' tally active
bit 1 = not used

wwwwxxxx

ASCII tally of most recent transport command ('P'=play, etc.)

yyyy

not used

zzzz

bit 8 = 0
bit 4 = 0
bit 2 = 0
bit 1 = over end point

13.10.16 ZETA SYSTEM (address Z)

TIME

HHMMSSFF MULTIPLEXED DATA
Requested with Lower Case ident's :

8 digit timecode
(User Bits *gggghhhh* = 00) . .

'n'	[6Eh]	SYSTEM END POINT
'p'	[70h]	SYSTEM PREROLL TIME
'q'	[71h]	SYSTEM GOTO POINT

10 digit timecode
(Subframes in User Bits *gggghhhh*)

'i'	[69h]	SYSTEM IN POINT
'j'	[6Ah]	SYSTEM OUT POINT
'u'	[75h]	CURRENT EVENT (front panel selected)
n'v'	[3?h,76h]	EVENT n (n=0 thru 9)

Bar/Beat

'r' [72h] BAR/BEAT CUE POINT (read only - calc'd when Cue point loaded)

HHMM	4 digit bar number
SS	Beat number (within bar)
FF	1/16 note number (within beat)
gggghhhh	00

USER BITS

aaaa Hardware AUX-IN Tip assignment:
0 = AUX 1
1 = TIMEBASE OUTPUT
2 = OFF

bbbb Hardware AUX-IN Ring assignment:
0 = AUX 2
1 = OFF

ccccdddd ASCII UPPERCASE representation of lower case ident of data currently loaded into the multiplexed time code area.

eeeeffff not used

STATUS

ssss Current system Enables (this status goes hand in hand with the 'Z' command):

z	Mast	Slv	Midi	System Master
0	-	-	-	Master
1	-	-	x	Midi
2	-	x	-	Slave
3	-	x	x	Slave
4	x	-	-	Master
5	x	-	x	Master
6	x	x	-	Master
7	x	x	x	Master

tttt	Current system Solos (this status corresponds to the 'XZ' command). Uses the same bit pattern as system Enables, but the only valid cobinations are 0,1,2,4.
uuuu	Master Address: 4=master transport 7=generator
vvvv	Resolve Mode Reference (if system resolved): 2=external 5=video
wwww	bit 8 = system resolved bit 4 = rehearse mode bit 2 = master transport controls disassigned bit 1 = master transport connector = AUX-OUT 3-10
xxxx	bit 8 = system under remote control bits 4,2,1 = system time code count mode and rate: 0 = 30 frames/sec 2 = 29.97 frames/sec NON-DROP-FRAME 3 = 29.97 frames/sec DROP-FRAME 4 = 24 frames/sec 5 = 25 frames/sec
yyyy	Cycle mode: 0 = OFF 1 = CYCLE 2 = AUTO REWIND 3 = AUTO STOP
zzzz	Auto-Edit mode: 0 = OFF 1 = AUTO-EDIT 2 = AUTO-EDIT WITH TALENT CUEING

13.11 ADAMS-SMITH PROTOCOL COMMANDS FOR ZETA-THREE

Note:
1. SYSTEM 2600 EQUIVALENT ADDRESSES . .

'LOWER' Modules		'UPPER' Modules	
0	reserved	G	reserved
1	reserved	H	reserved
2	ZETATIME	I	MIDI TIME CODE GENERATOR
3	MIDI CHASE EQUIV. TIME	J	MIDI CHASE
4	GENERATOR	K	GENERATOR Upper
5	reserved	L	EVENTS
6	SLAVE READER	M	SLAVE TRANSPORT
7	MASTER READER	N	MASTER TRANSPORT
		Z	ZETA SYSTEM

2. Conventions . .

(i) All characters and numerics in ASCII format

(ii) Control characters are indicated by a preceding @ e.g. @X for control X (1Ah) etc. (These may of course be transmitted as a single control character, or as the '@character' combination indicated.)

(iii) Square brackets are used to enclose symbols representing specific character choices. e.g. [hhmmssff] representing 8 ASCII digits of time code data; [n] representing a single ASCII digit whose value is explained in the text; etc.

13.11.2 ZETATIME (address 2)

O	CALC ZETATIME OFFSETS FROM SLAVE OFFSET - RESERVED
[hhmmssffxx] V	CALC ZETATIME OFFSET FROM OFFSET OVERRIDE - RESERVED
[n] Y	LOAD ZETATIME SOURCE ADDRESS n : 7=master 6=slave

13.11.3 MIDI CHASE EQUIVALENT TIME (address 3)

No commands

13.11.4 GENERATOR (address 4)

@O	LOAD TIME CODE FROM PRESET REGISTER
@T	LOAD PRESET REGISTER FROM TIME CODE
[n] @Y	ENABLE/DISABLE TIME CODE JAM n : 0 = disable 1 = enable
[n] @Z	ENABLE/DISABLE USER BITS JAM n : 0 = disable 1 = enable
[g] C	LOAD USER BITS GROUP FLAGS g = flags (0 thru 3 only)
[ff] F	LOAD COUNT MODE (FRAME RATE)

Mode	ff
24	24
25	25
30	30
30DF	70 or 69
29.97NON-DROP	29

[n] G	LOAD SYNC SOURCE n : 0 = Internal Crystal 1 = Follow master frame edge
[n] H	GENERATOR HOLD/RUN/RUN REVERSE n : 0 = Run 1 = Hold 2 = Run reverse
[ccvv] M	LOAD CONSTANT cc = constant number vv = new value
[hhmmssff] P	LOAD TC PRESET REGISTER
[hhmmssff] T	LOAD MAIN TIME CODE
[abcdefgh] U	LOAD USER BITS
[ua] Y	LOAD TIME CODE JAM ADDRESS u : bit 8 = jam to user bits bit 4 = 0 bit 2 = 0 bit 1 = 0 a : address of source code (0 thru 7 only)

[ua] Z

LOAD USER BITS JAM ADDRESS

u : bit 8 = jam to user bits
bit 4 = 0
bit 2 = 0
bit 1 = 0

a : address of source code (0 thru 7 only)

[dddddddnn] @X

LOAD USER BITS DUMP BUFFER

dddddddd : Data

nn : Offset into buffer
00 = Buffer start
01 = Buffer + 4 bytes
.
.
07 = Buffer + 28 bytes

Note: The buffer is 32 bytes long (64 digits), and may be fully loaded with eight of these commands.

[gnn] @X

BEGIN USER BITS DUMP

g : User bits group flags to be transmitted for the duration of the dump. (0 thru 3 only)

nn : 08

13.11.6 READERS (Slave at address 6, Master at address 7)

@O	LOAD TIME CODE FROM PRESET REGISTER
@T	LOAD PRESET REGISTER FROM TIME CODE
L	RE-LEARN TACH RATE
[hhmmssff] P	LOAD TC PRESET REGISTER
[hhmmssff] T	LOAD CTL/TACH UPDATE PRESET
[n] U	ENABLE/DISABLE U'BITS COMPENSATION n : 0 = disable 1 = enable

13.11.10 MIDI TIME CODE GENERATOR (address I)

D	DISABLE MIDI TIME CODE OUTPUT
E	ENABLE MIDI TIME CODE OUTPUT

13.11.11 MIDI CHASE (address J)

NOTE:
1. Many commands take on different meanings depending upon whether the immediately preceding character was NUMERIC or not. Numeric characters are 0 thru 9, XA thru XF, backspace (@H : 08h), DEL (7Fh), _ (5Fh), and ~ (7Eh).

2. While loading a Map to or from Tape, only the following commands will be honored :

[n] @T	SAVE/LOAD MAP TO TAPE
D	DISABLE SEQUENCER CONTROL
K	KILL SEQUENCER
[ccvv] K	LOAD CONSTANT
S	STOP

3. While in Learn mode, only the following commands will be honored :

[ccvv] K	LOAD CONSTANT
@A	ARM LEARN MODE
@C	ABORT LEARN MODE
@D	DISARM LEARN MODE

@A — ARM LEARN MODE

[n] @B — REMOTE BEEPER AND MIDI METRONOME ENABLES
n : 0=OFF
1=REMOTE BEEPER (INCLUDING COUNT IN)
2=REMOTE BEEPER COUNT IN ONLY
3=MIDI METRONOME (INCLUDING COUNT IN)
4=MIDI METRONOME COUNT IN ONLY
5=REMOTE AND MIDI (INCLUDING COUNT IN)
6=REMOTE AND MIDI COUNT IN ONLY

@C — ABORT LEARN MODE

@D — DISARM LEARN MODE

[nn] @E — MIDI TX/MERGE ENABLES
nn : Bits . .
01h=DISABLE SYNC,CONT/STOP,SONG PTR
02h=ENABLE SONG SELECT TRANSMISSION
04h=MERGE REAL TIME BYTES (EXCEPT F8)
08h=MERGE CHANNEL, SYSTEM COMMON DATA
10h=THRU IS OUT2
20h=MIDI TIME CODE ENABLED
40h=PERFORMER DIRECT TIME LOCK ENABLED

[mmnn] X @E — MASKED LOAD - MIDI TX/MERGE ENABLES
mm : Hex mask for bits in nn
nn : Bits to load (see @E)

[nn] @F

LOAD TIMEBASE FREQUENCY CODE
nn : 00 = 96 pulses per beat
01 = 120 pulses per beat
02 = 160 pulses per beat
03 = 240 pulses per beat
04 = 12 pulses per beat
05 = 16 pulses per beat
06 = 20 pulses per beat
07 = 24 pulses per beat
08 = 32 pulses per beat
09 = 40 pulses per beat
0XA = 48 pulses per beat
0XB = 60 pulses per beat
0XC = 80 pulses per beat

[n] @G

LOCK CONFIGURATION BITS
n : 0=No Splice Trap / Slow Relock
1=Enable Splice Trap
2=Enable Slow Lock
3=Enable both

[n] @H

LEARN QUANTIZATION VALUE
n : 0=Minimum, F=Maximum

[nn] @L

LOAD LEARN SOURCE
nn : 00=CAPTURE Tap - override avrg
01=CAPTURE Tap - direct to Map
02=MIDI Clock
03=AUX IN Frequency (Timebase)
04=AUDIO AUX IN ADAPTOR

@P

LOAD DEFAULT CONSTANTS

[n] @N

BANK SWITCH
n = Bank number (0 thru 6)

[n] @R

LOAD FRAMES-PER-BEAT FRAME RATE
n : 0=24
1=25
2=30
3=24/25/24 (System=24/25/30)
4=24/25/30 (Follow System)

[nn] @T

SAVE/LOAD MAP TO TAPE, SAVE/LOAD CANCEL
nn : 00 = Save to Tape
01 = Load from Tape
04 = Cancel Save/Load (Tape or MIDI File Dump)

[aa aa aa aa dd nn] @T	SAVE/LOAD MAP VIA MIDI FILE DUMP aa aa aa aa = File type consisting of four ASCII characters. Temporarily MIDI types only i.e. transmit 'MIDI' or 4XD 49 44 49 dd = Destination/Source device (0-7E, 7F=All/Any) nn : 02 = Transmit (save) file 03 = Receive (load) file
@V	CLEAR OVERRIDES (i.e. Beats per Minute, Frames per Beat, Time Signature, Click length.)
[bbbbnn] @X	EDIT - INSERT BARS AT MARK X. bbbb : HEX Number of bars to insert nn : 00 Note : MIDI sequence must be STOPPED.
[nn] @X	EDIT nn : 01 = DELETE A THRU B 02 = COPY AB to X 03 = Change TEMPO etc. (i.e. adopt all Overrides) Note : MIDI sequence must be STOPPED.
[sssnn] @Z	LOAD SONG NUMBER - DECIMAL sss :Song number, 001 thru 128 nn : 00
[eenn] @Z	SET SONG END TYPE ee : 00 = Stop at end 01 = Continue through end nn : 01
[nn] @Z	INCREMENT/DECREMENT SONG NUMBER nn : 10 = Increment Song number 20 = Decrement Song number
[xxnn] @Z	LOAD SONG NUMBER - HEX xx : Hex song number, 00 thru 7Fh nn : 40
(	SLOW SLEW - DECREMENT OFFSET Sequence **advances** relative to master.
)	SLOW SLEW - INCREMENT OFFSET Sequence **retards** relative to master.
[hhmmssffxx] +	INCREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each '+' command. e.g. 'o0001592399+' Valid registers for the '+' command are: 'o' OFFSET 'q' CUE POINT

[hhmmssffxx] -	DECREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. See comments and register selections for the '+' command.
<	FAST SLEW - DECREMENT OFFSET Sequence **advances** relative to master.
>	FAST SLEW - INCREMENT OFFSET Sequence **retards** relative to master.
A	"AUTO" LOCK MODE Sequencer will initially lock to the Master time position dependent (Lock Mode 'N'), subsequently switching to resolve to the Master frame edge, ignoring Time code addresses (Lock Mode 'W'). Time before switch over is given by Constant 38. (See also commands N,W)
D	DISABLE SEQUENCER CONTROL
E	ENABLE SEQUENCER CONTROL (Causes immediate Song Pointer to be issued.)
G	GIVE UP SLEW MODE Sequencer returns to its original offset, moving at the last slewing rate used.
H	ENTER CHASE MODE
I	INDEPENDANT - EXIT CHASE MODE AND STOP
J	CONDITIONAL INDEPENDANT COMMAND If the sequencer is currently playing, then it will be allowed to continue playing independantly (equivalent to a Play command being issued). If not, then it will be taken out of Chase and stopped.
K	KILL SEQUENCER Always causes Stop command.
[ccvv] K	LOAD CONSTANT cc = constant number vv = new value
[ccvvmm] X K	MASKED CONSTANT LOAD cc = constant number vv = new value mm = binary mask
M	MAINTAIN SLEWED POSITION Temporary offset created by slewing is made permanent, and slewing mode is exited.
[hhmmssffxx] M	LOAD MASTER REFERENCE / LOCATE POINT

N — NORMAL LOCK MODE
Sequencer will lock to the Master time position dependent. (See also commands A,W)

O — CALCULATE OFFSET
New Offset = Cue Point - Master Reference Point

[hhmmssffxx] O — LOAD OFFSET

P — PLAY

Q — CUE TO CUE POINT

[hhmmssffxx] Q — LOAD CUE POINT (and calculate bar/beat register 'r')

[bbbbttssnn] R — LOAD BAR/BEAT CUE POINT (and back calculate 'Q' point)
bbbb : Bar number
tt : Beat number
ss : Sixteenth note
nn : 00

[ppppnn] R — LOAD BCD SONG POINTER
pppp :BCD Song Pointer
nn : 40
This may change !!!

S — STOP

[hhmmssffxx] S — LOAD START TIME REGISTER
(Offset is automatically adjusted)
xx : Subframes

T — TRAP CURRENTLY ATTACHED REGISTER
Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each Trap command.
e.g. 'oT' = Trap offset register
'sT' = Trap start time
Valid registers for the Trap command are:

Ident.	Register	Trap source
'o'	OFFSET	ACTUAL OFFSET
'q'	CUE POINT	SELF TIME
's'	START TIME	MASTER TIME

[hhmmssffxx] T — LOAD SEQUENCE EQUIVALENT TIME CODE
Causes immediate locate to this time.

[bbbbttssnn] U — LOAD MAP BAR/BEAT
bbbb :Bar number
tt : Beat number
ss : Sixteenth note
nn : 00
Note 1. Sequence motion must be STOPPED.
2. Causes an immediate locate.

[inn] U

INCREMENT/DECREMENT MAP POSITION
i : index into Bar/Beat register [bbbbttss].
0 = adjust 1/16 notes
1 = adjust 1/16 notes tens
2 = adjust beat
.
.
7 = adjust bar number 1000's
nn : 10 = Increment
20 = Decrement
Note 1. Sequence motion must be STOPPED.
2. Causes an immediate locate.

[ppppnn] U

LOAD BCD SONG POINTER
pppp :BCD Song pointer
nn : 40
**** THIS MAY CHANGE
Note 1. Sequence motion must be STOPPED.
2. Causes an immediate locate.

[hhhhnn] U

LOAD HEX SONG POINTER
hhhh :Hex song pointer
nn : XF2
Note 1. Sequence motion must be STOPPED.
2. Causes an immediate locate.

[bbbxxnn] V

LOAD BPM OVERRIDE
bbb :Beats per Minute
xx : Fractional beats per min (/100)
nn : 00
Note Sequence must not be Locking.

[inn] V

INCREMENT/DECREMENT BPM OVERRIDE
i : Index into BPM buffer [bbbxx]
0 = bpm /100
.
.
4 = bpm X100
nn : 10 = Increment
20 = Decrement
Note Sequence must not be Locking.

[ffyyynn] V

LOAD FPB OVERRIDE
ff : Frames per Beat
yyy: Fractional frames per beat (1/1000)
nn : 01
Note Sequence must not be Locking.

[inn] V	INCREMENT/DECREMENT FPB OVERRIDE i : Index into FPB buffer [ffyyy] 0 = fpb /1000 . . 4 = fpb X10 nn : 11 = Increment 21 = Decrement Note Sequence must not be Locking.
[ccddnn] V	LOAD TIME SIGNATURE OVERRIDE cc : Numerator dd : Denominator nn : 02 Note Sequence must be STOPPED.
[inn] V	INCREMENT/DECREMENT TIME SIG. OVERRIDE i : Index into Time Signature buffer [ccdd] 0 = denominator units . . 3 = numerator tens nn : 12 = Increment 22 = Decrement Note Sequence must be STOPPED.
[smmnn] V	LOAD CLICK LENGTH OVERRIDE s : 0 = normal 1 = dotted mm : Click length (04 = 1/4 note etc.) nn : 03 Note Sequence must not be Locking.
[nn] V	INCREMENT/DECREMENT CLICK OVERRIDE nn : 13 = Increment 23 = Decrement Note Sequence must not be Locking.
W	FREEWHEEL LOCK MODE Sequencer will now resolve to the Master frame edge. Time code addresses are ignored. (See also commands A,N)
[bbbbttssnr] W	LOAD WORK REGISTER bbbb :Bar number tt : Beat number ss : Sixteenth note n : 0 r : 0 = Mark A 1 = Mark B 2 = Mark X

[inr] W	INCREMENT/DECREMENT WORK REGISTER i : Index into work register buffer [bbbbttss]. n : 1 = Increment 2 = Decrement r : 0 = Mark A 1 = Mark B 2 = Mark X
[ppppnr] W	LOAD HEX SONG POINTER TO WORK REGISTER pppp :Hex Song Pointer n : 4 r : 0 = Mark A 1 = Mark B 2 = Mark X
[nr] W	TRAP WORK REGISTER n : 8 = Trap current Bar/Beat 9 = Trap end of song r : 0 = Mark A 1 = Mark B 2 = Mark X
Z	PAUSE = Stop
[	DECREMENT OFFSET BY 1 FRAME Sequence advances relative to master. (Subframes are made 0)
]	INCREMENT OFFSET BY 1 FRAME Sequence retards relative to master. (Subframes are made 0)
^	HOLD SLEW POSITION

13.11.12 GENERATOR Upper (address K)

[ccvv] K	LOAD CONSTANT cc = constant number vv = new value
[hhmmssffxx] P	LOAD TC PRESET REGISTER
[ccvvmm] X K	MASKED CONSTANT LOAD cc = constant number vv = new value mm = binary mask

13.11.13 EVENTS (address L)

NOTE: Many commands take on different meanings depending upon whether the immediately preceding character was NUMERIC or not. Numeric characters are 0 thru 9, XA thru XF, backspace (@H : 08h), DEL (7Fh), _ (5Fh), and ~ (7Eh).

[v] @A	ARM EVENT v : Event number (0 thru 9)
[v] @D	DISARM EVENT v : Event number (0 thru 9) Disarm all events with v=0Ah
@N	RESET ALL EVENT OUTPUTS
[v] @R	TURN ON CONTINUOUS EVENT OUTPUT v : Event number (0 thru 9)
[v] @T	FIRE CONTINUOUS AND MOMENTARY OUTPUTS v : Event number (0 thru 9)
[nv] @V	LOAD EVENT ENABLE TYPE n: 0 = Normal trigger to selected AUX-OUT 1 = MIDI Note On 2 = MIDI Program Change 3 = Remote Function Key trigger v : Event number (0 thru 9)
[ccpttyynv] @W	LOAD MIDI EVENT PARAMETERS cc = channel (0-0Fh) pp = program change (0-7Fh) tt = note value (0-7Fh) yy = velocity (0-7Fh) n: 0 = Load v: Event number (0 thru 9)
[iaanv] @W	ADJUST MIDI EVENT PARAMETERS i: 0 = channel 1 = program change 2 = note value 3 = velocity aa = increment/decrement amount (hex) n: 1 = increment 2 = decrement v: Event number (0 thru 9)
[v] @X	TURN OFF CONTINUOUS EVENT OUTPUT v : Event number (0 thru 9)

[hhmmssffxx] +	INCREMENT CURRENTLY ATTACHED EVENT by the preceeding 10 digit time value. Although the appropriate event may have already been selected by a previous data read back, it is probably better to re-specify it before each '+' command. e.g. '5v0001592399+' The only valid registers for the '+' command are the Event registers.
[hhmmssffxx] -	DECREMENT CURRENTLY ATTACHED EVENT by the preceeding 10 digit time value.
[ccvv] K	LOAD CONSTANT cc = constant number vv = new value
[ccvvmm] X K	MASKED CONSTANT LOAD cc = constant number vv = new value mm = binary mask
[v] N	FIRE EVENT MIDI NOTE ON/OFF SEQUENCE v: Event number (0 thru 9)
[v] P	FIRE EVENT MIDI PROGRAM CHANGE v: Event number (0 thru 9)
[v] R	PULSE EVENT OUTPUT v : Event number (0 thru 9)
T	TRAP CURRENTLY ATTACHED EVENT Although the appropriate event may have already been selected by a previous data read back, it is probably better to re-specify it before each Trap command. e.g. '2vT' = Trap event 2 The only valid registers for the Trap command are the Event registers.
[hhmmssffqv] V	LOAD EVENT POINT q : Quarter frame division (0-3) v : Event number (0 thru 9)
[hhmmssffxx] W	LOAD EVENT POINT with SUBFRAMES (Event number to be set in advance with 'v' register request.)

13.11.14 SLAVE TRANSPORT (address M)

NOTE: Many commands take on different meanings depending upon whether the immediately preceding character was NUMERIC or not. Numeric characters are 0 thru 9, XA thru XF, backspace (@H : 08h), DEL (7Fh), _ (5Fh), and ~ (7Eh).

[n] @A

ARM RECORD/REHEARSE
n: 0 = Record
1 = Record
2 = Rehearse
Note Both punch in AND punch out are enabled together.

[n] @D

DISARM RECORD/REHEARSE
n: 0 = Record
1 = Record
2 = Rehearse
8 = Everything
Note Both punch in AND punch out are disabled together.

@F

ROLL FORWARD
Cues transport forwards by the time specified in the System Rollback register (default value = 5 sec).

[n] @G

LOCK CONFIGURATION BITS
n : 0=No Splice Trap / Slow Relock
1=Enable Splice Trap
2=Enable Slow Lock
3=Enable both

@R

ROLL BACK
Similar to Roll Forward, but in reverse direction.

[nddd] @X

LOAD/SAVE TRANSPORT CONSTANTS
n: 0 = Load from EPROM
1 = Load from User-Area
2 = Save to User-Area
ddd: Transport/User-Area Ident

User-Area Idents . .
000 thru 009

Transport Ident's . .

000	AEG M-15A (38 cm/s)
001	AEG M-15A (76 cm/s)
002	AEG M-20
003	AKAI MG-1212
004	AKAI MG-1214
005	AKAI MG-14D
006	AMPEX ATR-100/104
007	AMPEX ATR-116/124
008	AMPEX MM-1200
009	AMPEX VPR-2/80
010	AMPEX VPR-6
011	AMPEX VPR-6 (Master only)
012	FOSTEX B16 (DC)
013	FOSTEX B16D (FM)
014	same as 013
015	FOSTEX 20,80 (DC)
016	FOSTEX E-Series
017	JVC CR-5550/6650
018	JVC BR-8600U
019	JVC CR-850
020	not used
021	JVC BR-7700
022	not used
023	not used
024	JVC CR-8250
025	not used
026	JVC BR-6400U
027	not used
028	MCI/SONY JH-16/24/114
029	MCI/SONY JH-110 A/B/C
030	MITSUBISHI X-80/X-80A
031	MITSUBISHI X-800
032	MITSUBISHI X-850
033	MITSUBISHI X-86
034	MITSUBISHI X-400
035	MITSUBISHI X-850 'D'
036	MITSUBISHI X-86 'D'
037	MITSUBISHI X-400 'D'
038	NAGRA T ("TAERP")
039	OTARI MTR-10,12
040	OTARI MTR-10II,12II
041	OTARI MTR-90I
042	OTARI MTR-90II
043	OTARI 5050 (34 pin)
044	OTARI 5050 (16 pin)
045	OTARI MX-7800
046	OTARI DTR-900
047	OTARI MTR-20
048	OTARI MX-70
049	OTARI MX-80
050	OTARI DTR-900 'D'
051	PANASONIC AG-6300 (Zeta cable 024)

052	PANASONIC NV-8500
053	PANASONIC AU-650
054	SONY VO-2850 (Master)
055	SONY VO-5850
056	SONY BVU-800/820/850
057	SONY BVW-10/40
058	SONY PCM-3324 (44.1KHz)
059	SONY APR-5000/5002/5003
060	SONY VO-5600
061	SONY VO-5600 (Search w/Pict)
062	SONY VO-5630 - same as 060
063	SONY VO-5630 (SwP) - same as 061
064	SONY VO-5800
065	SONY PCM-3202 (44.1KHz)
066	SONY SLO-325 (Master)
067	SONY PCM-3202 (48KHz)
068	SOUNDCRAFT 760MK-III/762MK-III
069	SOUNDCRAFT SATURN/ Series 20
070	STUDER A80 (DC)
071	STUDER A80 (FM)
072	STUDER A800 MK-II
073	STUDER A800 MK-III
074	STUDER A810
075	STUDER B67
076	STUDER A80 2" (DC)
077	STUDER A80 2" (FM)
078	STUDER A820
079	TASCAM 40/50 Series
080	TASCAM 85-16B
081	TASCAM 388/STUDIO 8
082	TASCAM 60 Series
083	TASCAM MS-16
084	TASCAM ATR-80
085	3M M-79
086	not used
087	not used
088	not used
089	AMPEX VPR-3
090	not used
091	not used
092	not used
093	not used
094	not used
095	not used
096	not used
097	not used
098	not used
099	not used
100	SONY PCM-3324 (48KHz)
101	SONY VP-7000/7600
102	not used
103	not used
104	not used
105	STELLAVOX TD-9 (w/TC)

106	not used
107	not used
108	not used
109	not used
110	not used
111	AKAI DR-1200
112	LYREC TR-533
113	OTARI MTR-100A
114	OTARI MX-50
115	OTARI MX-55
116	SONY APR-24
117	SONY PCM-3402 (44.1KHz)
118	SONY PCM-3402 (48KHz)
119	SONY VP-9000/VO-9600
120	SONY VO-9800
121	SONY VO-9850
122	not used
123	STUDER A807
124	STUDER A812
125	STUDER A820 2"
126	STUDER C270
127	TASCAM MSR-16
128	TASCAM 60-16
129	FOSTEX D-20 DAT
130	JVC BR-S810U/S610U
131	PANASONIC AG-7500/7300/6500/6300
132	SONY BVU-950 (PARALLEL, MAST ONLY)
133	STUDER C278
134	FOSTEX D-20 DAT 'D'
135	JVC KR-M800U

@Y	CYCLE BETWEEN CUE AND END POINTS
@Z	ZAP CYCLE MODE
(	SLOW SLEW - DECREMENT OFFSET Slave **advances** relative to master.
)	SLOW SLEW - INCREMENT OFFSET Slave **retards** relative to master.

[hhmmssffxx] +	INCREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each '+' command. e.g. '0001592399+' '1v0000000050+' Valid registers for the '+' command are: 'o' OFFSET 'q' CUE POINT '0v' EVENT 0 (= IN POINT) '1v' EVENT 1 (= OUT POINT) 'i' IN POINT 'j' OUT POINT 'l' LOCATE POINT / MASTER REFERENCE 'n' END POINT
[hhmmssffxx] -	DECREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. See comments and register selections for the '+' command.
<	FAST SLEW - DECREMENT OFFSET Slave **advances** relative to master.
>	FAST SLEW - INCREMENT OFFSET Slave **retards** relative to master.
A	"AUTO" LOCK MODE Transport will initially lock to the Master time position dependent (Lock Mode 'N'), subsequently switching to resolve to the Master frame edge, ignoring Time code addresses (Lock Mode 'W'). Time before switch over is given by Constant 38. (See also commands N,W)
D	DISABLE TRANSPORT CONTROL (Also disarms Record/Rehearse)
E	ENABLE TRANSPORT CONTROL
F	FAST FORWARD
G	GIVE UP SLEW MODE Transport returns to its original offset, moving at the last slewing rate used.
H	ENTER CHASE MODE
I	INDEPENDANT - EXIT CHASE MODE AND STOP
[hhmmssffxx] I	LOAD IN POINT xx : Subframes

J	CONDITIONAL INDEPENDANT COMMAND If the transport is currently playing, then it will be allowed to continue playing independantly (equivalent to a Play command being issued). If not, then it will be taken out of Chase and stopped.
[hhmmssffxx] J	LOAD OUT POINT xx : Subframes
K	KILL TRANSPORT Always causes Stop command, even for transports using only Pause.
[ccvv] K	LOAD CONSTANT cc = constant number vv = new value
[ccvvmm] X K	MASKED CONSTANT LOAD cc = constant number vv = new value mm = binary mask
L	LOCATE TO MASTER REFERENCE POINT Transport is parked, ready to lock, as if Master device was in fact parked at the Master Reference Point.
M	MAINTAIN SLEWED POSITION Temporary offset created by slewing is made permanent, and slewing mode is exited.
[hhmmssffxx] M	LOAD MASTER REFERENCE / LOCATE POINT
N	NORMAL LOCK MODE Transport will lock to the Master time, position dependent. (See also commands A,W)
[hhmmssffxx] N	LOAD END POINT
O	CALCULATE OFFSET New Offset = Cue Point - Master Reference Point
[hhmmssffxx] O	LOAD OFFSET
P	PLAY
[ddd] X P	VARIABLE PLAY ddd: Percent of playspeed, (100=play) Limited to 050 thru 150 (+/- 50%), and may be further limited by Constant 44.
Q	CUE TO CUE POINT
[hhmmssffxx] Q	LOAD CUE POINT

R — REWIND

[v] R — PULSE EVENT OUTPUT
v : 0 = Record in
1 = Record out
2 = Rehearse in
3 = Rehearse out

S — STOP

T — TRAP CURRENTLY ATTACHED REGISTER
Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each Trap command.
e.g. 'oT' = Trap offset register
'jT' = Trap punch out point
Valid registers for the Trap command are:

ID.	Register	Trap source
'o'	OFFSET	ACTUAL OFFSET
'q'	CUE POINT	SELF TIME
'0v'	EVENT 0	Trigger* TIME
'1v'	EVENT 1	Trigger* TIME
'i'	IN POINT	Trigger* TIME
'j'	OUT POINT	Trigger* TIME
'l'	LOCATE (M.Ref)	MASTER TIME
'n'	END POINT	SELF TIME

* Trigger time may be either MASTER or SELF TIME, depending on the selection of events trigger source.

[hhmmssffqv] V — LOAD EVENT POINT
q : Quarter frame division (0-3)
v : Event number (0 or 1 only)
Note Event 0 = IN POINT
Event 1 = OUT POINT

W — FREEWHEEL LOCK MODE
Transport will now resolve to the Master frame edge. Time code addresses are ignored. (See also commands A,N)

Z — PAUSE

[— DECREMENT OFFSET BY 1 FRAME
Slave **advances** relative to master. (Subframes are made 0)

] — INCREMENT OFFSET BY 1 FRAME
Slave **retards** relative to master. (Subframes are made 0)

^ — HOLD SLEW POSITION

13.11.15 MASTER TRANSPORT (address N)

NOTE: Many commands take on different meanings depending upon whether the immediately preceding character was NUMERIC or not. Numeric characters are 0 thru 9, XA thru XF, backspace (@H : 08h), DEL (7Fh), _ (5Fh), and ~ (7Eh).

[n] @A

ARM RECORD/REHEARSE
n: 0 = Record
1 = Record
2 = Rehearse
Note Both punch in AND punch out are enabled together.

[n] @D

DISARM RECORD/REHEARSE
n: 0 = Record
1 = Record
2 = Rehearse
8 = Everything
Note Both punch in AND punch out are disabled together.

@F

ROLL FORWARD
Cues transport forwards by the time specified in the System Rollback register (default value = 5 sec).

@R

ROLL BACK
Similar to Roll Forward, but in reverse direction.

[nddd] @X

LOAD/SAVE TRANSPORT CONSTANTS
n: 0 = Load from EPROM
1 = Load from User-Area
2 = Save to User-Area
ddd: Transport/User-Area Ident

User-Area Idents . .
000 thru 009

Transport Ident's (w/cable #'s) . .
see SLAVE TRANSPORT section.

@Y

CYCLE BETWEEN CUE AND END POINTS

@Z

ZAP CYCLE MODE

[hhmmssffxx] +	INCREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each '+' command. e.g. 'i0001592399+' '1v0000000050+' Valid registers for the '+' command are: 'q' CUE POINT '0v' EVENT 0 (= IN POINT) '1v' EVENT 1 (= OUT POINT) 'i' IN POINT 'j' OUT POINT 'n' END POINT
[hhmmssffxx] -	DECREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. See comments and register selections for the '+' command.
D	DISABLE TRANSPORT CONTROL (Also disarms Record/Rehearse)
E	ENABLE TRANSPORT CONTROL
F	FAST FORWARD
I	INDEPENDANT = STOP
[hhmmssffxx] I	LOAD IN POINT xx : Subframes
J	CONDITIONAL INDEPENDANT COMMAND If the transport is not currently playing then it will be stopped.
[hhmmssffxx] J	LOAD OUT POINT xx : Subframes
K	KILL TRANSPORT Always causes Stop command, even for transports using only Pause.
[ccvv] K	LOAD CONSTANT cc = constant number vv = new value
[ccvvmm] X K	MASKED CONSTANT LOAD cc = constant number vv = new value mm = binary mask
[hhmmssffxx] N	LOAD END POINT
P	PLAY

Q	CUE TO CUE POINT
[hhmmssffxx] Q	LOAD CUE POINT
R	REWIND
[v] R	PULSE EVENT OUTPUT v : 0 = Record in 1 = Record out 2 = Rehearse in 3 = Rehearse out
S	STOP
T	TRAP CURRENTLY ATTACHED REGISTER Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each Trap command. e.g. 'qT' = Trap cue point 'jT' = Trap punch out point Valid registers for the Trap command are:

ID.	Register	Trap source
'q'	CUE POINT	SELF TIME
'0v'	EVENT 0	Trigger* TIME
'1v'	EVENT 1	Trigger* TIME
'i'	IN POINT	Trigger* TIME
'j'	OUT POINT	Trigger* TIME
'l'	LOCATE (M.Ref)	MASTER TIME
'n'	END POINT	SELF TIME

* Trigger time may be either MASTER or SELF TIME, depending on the selection of events trigger source.

[hhmmssffqv] V	LOAD EVENT POINT q : Quarter frame division (0-3) v : Event number (0 or 1 only) Note Event 0 = IN POINT Event 1 = OUT POINT
Z	PAUSE

13.11.16 ZETA SYSTEM (address Z)

NOTE:Many commands take on different meanings depending upon whether the immediately preceding character was NUMERIC or not. Numeric characters are 0 thru 9, XA thru XF, backspace (@H : 08h), DEL (7Fh), _ (5Fh), and ~ (7Eh).

[n] @A	ARM EVENT n: Event number (0 thru 9)
[n] @B	SET CURRENT EVENT NUMBER n: Event number (0 thru 9)
@C	CLEAR REGISTERS System GOTO, IN, OUT, END. Slave and MIDI chase OFFSETs. Master and Slave END POINTS ('Limits') System LOOP requests.
[n] @D	DISARM EVENT n: Event number (0 thru 9)
[n] @E	ARM/DISARM CURRENT EVENT n: 0=Disarm 1=Arm
[n] @I	SELECT RECORD/REHEARSE n: 0=Record 1=Rehearse
@K	ZETA POWER UP TYPE RESET
[n] @L	LOCAL/REMOTE ENABLE MODE n: 0 = Local - the ENABLE keys and the Z command work normally. 1 = Remote - enabling a transport with either the ENABLE keys or the Z command will NOT cause the transport to enter the Chase mode.
[n] @N	ASSIGN MASTER n: 4=Generator 7=Master TC Reader
[n] @O	MASTER OUTPUT CONNECTOR FUNCTION n: 0=Transport controls 1=Off 2=Aux outputs 3 thru 10
[n] @V	SET RESOLVE MODE AND REFERENCE n: 2=External input 5=Video 8=Off (System Unresolved) [Other values ignored]

[ff] @W

LOAD COUNT MODE (FRAME RATE)
(Same as Generator 'F' command)

Mode	*ff*
24	24
25	25
30	30
30DF	70 or 69
29.97NON-DROP	29

[n] @X

AUTO-EDIT MODE
n: 0=Off
1=Auto-Edit
2=Auto-Edit plus Talent Cueing

[n] @Y

SET SYSTEM CYCLE MODES
n: 0=Off
1=Cycle
2=Auto Rewind
3=Auto Stop

[tn] @Z

HARDWARE SETUP
(t=TYPE, n=DATA)
tn: 00=Aux Out Tip = AUX 1
01=Aux Out Tip = TIMEBASE
02=Aux Out Tip = OFF
10=Aux Out Ring = AUX 2
11=Aux Out Ring = OFF
[12=Aux Out Ring = CONT/STOP]

[hhmmssffxx] +

INCREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each '+' command.
e.g. 'i0001592399+'
'1v0000000050+'
Valid registers for the '+' command are:
'i' IN POINT
'j' OUT POINT
'q' CUE (GOTO) POINT
'n' END POINT
'u' CURRENT EVENT
'nv' EVENT (n=Event number 0 thru 9)

[hhmmssffxx] -

DECREMENT CURRENTLY ATTACHED REGISTER by the preceeding 10 digit time value. See comments and register selections for the '+' command.

C

SYSTEM CONDITIONAL PLAY
Same as normal SYSTEM PLAY, unless a Cue (GOTO) is in progress, in which case Playing will begin when the Cue Point is reached.

F

SYSTEM FAST FORWARD

[n] H	GENERATOR TIME CODE LINK ON/OFF n: 0=Off 1=On F=reserved
[hhmmssffxx] I	LOAD SYSTEM IN POINT xx: Subframes
[nzhhmmssffxx] XI	LOAD SYSTEM IN POINT - DEVICE SPECIFIC n: 0=Timecode, 8=Bar/Beat z: Command 'Z' format xx: Subframes [If n=8, then hhmm=Bar number ss=Beat number ff=Sixteenth note xx=00] Allows specification of source of time code data - Zeta will adjust to match the current 'system master'. e.g. If the IN point is loaded as 'Slave' data while the current system master is the Master itself, then the requested IN point will be adjusted by the Slave Offset before loading the the System IN point.
[hhmmssffxx] J	LOAD SYSTEM OUT POINT xx: Subframes
[nzhhmmssffxx] XJ	LOAD SYSTEM OUT POINT - DEV. SPECIFIC (See XI command for details)
[hhmmssffxx] L	LOAD SYSTEM ROLLBACK VALUE
[hhmmssffxx] N	LOAD SYSTEM END POINT
[nzhhmmssffxx] XN	LOAD SYSTEM END POINT - DEV. SPECIFIC (See XI command for details)
P	SYSTEM PLAY
[hhmmssffxx] P	LOAD SYSTEM PREROLL VALUE
Q	SYSTEM CUE TO CUE (GOTO) POINT
[hhmmssffxx] Q	LOAD SYSTEM CUE (GOTO) POINT (Also calculates read only bar/beat register 'r')
[nzhhmmssffxx] XQ	LOAD SYSTEM CUE POINT - DEV. SPECIFIC (See XI command for details) (Also calculates read only bar/beat register 'r')
R	SYSTEM REWIND

[n] R	RECORD/REHEARSE IN/OUT n: 0=Punch In 1=Punch Out (Record/Rehearse selected by @I cmd)
S	SYSTEM STOP
T	TRAP CURRENTLY ATTACHED SYSTEM REGISTER Although the appropriate register may have already been selected by a previous data read back, it is probably better to re-specify it before each Trap command. e.g. 'qT' = Trap cue point 'jT' = Trap punch out point Valid registers for the Trap command are: ID. Register 'i' IN POINT 'j' OUT POINT 'q' CUE (GOTO) POINT 'n' END POINT 'u' CURRENT EVENT 'nv' EVENT (n=event # 0-9) The source of data is always the current 'system master' time.
U	SYSTEM PAUSE (Non-pause transports will be issued a STOP command)
[hhmmssffxx] U	LOAD CURRENT SYSTEM EVENT POINT
[nzhhmmssffxx] XU	LOAD CURRENT SYSTEM EVENT - DEVICE SPECIFIC (See XI command for details)
[hhmmssffqv] V	LOAD SYSTEM EVENT POINT q: Quarter frame division (0-3) v: Event number (0 thru 9)
[nzhhmmssffqv] XV	LOAD SYSTEM EVENT POINT - DEV. SPECIFIC (See XI command for details) q: Quarter frame division (0-3) v: Event number (0 thru 9)
[hhmmssffxx] W	LOAD SYSTEM EVENT POINT with SUBFRAMES (Event number to be set in advance with 'v' register request.)
[nzhhmmssffxx] XW	LOAD SYSTEM EVENT POINT with SUBFRAMES - DEVICE SPECIFIC (See XI command for details) (Event number to be set in advance with 'v' register request.)

[z] Y

MASTER/SLAVE RECORD/REHEARSE ENABLE

z	Mast	Slv	
0	-	-	Disable both
2	-	x	Master disable, Slave enable
4	x	-	Master enable, Slave disable
6	x	x	Enable both

Note: For each specified enable, this command will transfer the system in and out points to the local transport registers (at addresses M and/or N), and arm the transport for record or rehearse according to system setting (see the @I SELECT RECORD/REHEARSE command).
If no enable is specified, then a "disarm all" local command will be issued.

[z] Z

ENABLE TRANSPORTS AND SEQUENCER

z	Mast	Slv	Midi	System Master
0	-	-	-	Master
1	-	-	x	Midi
2	-	x	-	Slave
3	-	x	x	Slave
4	x	-	-	Master
5	x	-	x	Master
6	x	x	-	Master
7	x	x	x	Master

Note: This command serves the same function as the front panel ENABLE switches. Under normal conditions, enabling a transport causes it to also enter the Chase mode.

[sz] XZ

ENABLE WITH SOLO CAPABILITY

z: as above
s: 0=No solos
1=MIDI solo
2=Slave solo
4=Master solo
[All other values ignored]

Appendix A
Time Code

In the 1960s, electronic editing became a viable technology for video and audio production. However, the selective recording of program material from one transport to another presented problems. How could two or more tape transports be synchronized to insure frame accurate edits? How could specific points be identified to make dubs, to insert sound effects, or to mark the beginning of a particular sequence? Was the tape in motion and if so, what was its speed?

The answers were provided by the Society of Motion Pictures and Television Engineers (SMPTE) by the specification and approval of a time code format, now industry recognized as "SMPTE" time code.

SMPTE time code is an electronic signal which is recorded on the program tape, typically on a unused audio channel (longitudinal time code, or LTC). The signal is digital in nature, resembling a string of square waves, called "bits", which are encoded with binary "1"s and "0"s.

Because time code was originally developed for the video industry, many of the specifications relate to video technology. For example, time code is specified to have exactly 80 "bits" per video frame. In the U.S., our system of television was originally developed with a rate of 30 frames per second, therefore time code is generated at 2400 bits per second. In European countries that produce video at 25 frames per second, the time code rate calculates to 2000 bits per second.

Each 80 bits of data, numbered from bit "0" to bit "79", is called a time code "word".

In a video application, the time code word must be generated so that bit "0" starts at the beginning of each video frame and bit "79" stops exactly at the end of the frame. In an audio application the relationship between the start of the time code word and the recorded material is arbitrary and not important.

There are various types of information encoded within the 80-bit time code word, the most important being the address data. The time code address is the number used by editing and synchronizing systems to locate a particular point on the tape. The address is an eight digit number in a 24 hour time format of HOURS:MINUTES:SECONDS:FRAMES. Valid time code numbers range between 00:00:00:00 and 23:59:59:29, and occupy 26 of the 80 bits.

Also recorded into the time code word are "USER-BITS". User-Bits occupy 32 bits of data and are available for any end-user application. Typically, user-bits define static data, for example, the date of the recording, reel number, scene number, or any numeric label.

The remaining bits make up the "sync word", which defines the end of the time code word and the direction of the code in playback; and status bits which flag different modes of operation.

The most important mode and the one which causes the most confusion is "DROP-FRAME".

Drop-frame is a mode of operation which compensates for the difference in frame rates between black and white video and color video. Because of technical difficulties in transmitting the color signal on broadcast channels, the frame rate had to be adjusted from 30 frames per second to approximately 29.97 frames per second. In broadcast applications where time code was used as the station clock, the clock would run slow because of the reduced frame rate. The solution to the problem was Drop-Frame.

Drop-Frame compensates for the slower rate by jumping the frame count ahead at periodic intervals, just as you would adjust a slow running clock by pushing the hands forward. The Drop-Frame mode causes the frame count to "drop" the numbers "00" and "01" on the minute so that the count progresses from frame :28 to :29 to frame :02, i.e., pushed ahead by two frames. Because dropping 2 frames every minute would over-compensate the error, the "dropped frames" are added back in on the tenth minute, i.e., not dropped. Pushing the count ahead in this manner effectively causes the time code to increment at the same rate as a clock.

The Drop-Frame mode is much more important to the broadcaster than the post-production facility, so many studios choose the "Non-Drop" mode. The decision to use one mode or the other is up to the user. What is important is to select a mode and use it consistently; that is, not mix Drop-Frame and Non-Drop Frame modes because doing so can cause problems for some editing systems.

FRAME RATE

As mentioned above, there are different time code rates for different video systems. Choosing the correct rate is the single **most important** parameter when generating time code for either audio or video use.

When generating time code for video editing applications, the time code rate must be the same standard

as the video. In the U.S. and other countries that use 60Hz power, that rate is typically 29.97 frames per second. In countries that use 50Hz power, the rate is 25 frames per second.

To insure the correct rate and phasing (where the time code word starts in relation to the video frame), the time code generator MUST be fed a sync reference signal. This signal may be: either the video signal which is to be recorded with time code; or an external reference in the form of composite video, composite sync, color bars or black-burst. When an external reference is used, it must be fed to the video recorder as well as to the time code generator.

For audio applications, where time code is used as the basis for transport synchronization, the correct frame rate is equally as important. SYNCHRONIZERS CAN NOT OPERATE PROPERLY USING TIME CODES WITH DIFFERENT RATES.

To maintain the correct speed, each synchronizer in a system controlling a transport must have time code of the same rate from both the master and <u>all</u> slave transports. The time code addresses can be different, and even the Drop-Frame modes can be different (using Adams-Smith products), but the frame rate must be the same.

The choice of an audio time code rate is determined by the final use of the program material.

If the audio material will ever be used in relation to video, then the audio time code should be recorded at the corresponding video rate. In the U.S., the time code generator would be set to 29.97 frames per second.

If the audio material is for an audio program only, then the choice of frame rate is arbitrary. The only requirement is all of the material must have the same rate time code. In practice, you should use a rate similar to the established norm, i.e., 29.97 or 30.00 frames per second in the U.S., or 25.00 frames per second where appropriate.

RECORDING TIME CODE

Time code is recorded in the same manner as any audio signal with one exception: **the record level is very critical.**

Because time code is in the form of a square wave, the signal becomes very distorted during the record-play process. Time code readers are designed to ignore most of these amplitude problems, but an excessive record level can introduce distortions which will cause decoding errors.

The shape of the time code signal also induces crosstalk problems which are aggrevated by a high record level.

The solution is to record time code at the lowest possible level which provides error-free decoding. Due to variations in types of transports, the best method of choosing a proper level is by trial and error, however experience has shown that the following values will give good performance.

1" vtr's = -5vu to -10vu
3/4" & 1/2" vcr's = -3vu to -5vu
audio recorders = -5vu to -10vu

To prevent crosstalk from entering adjacent pre-amp stages prior to the level controls, the time code signal from the generator should be less than 1 volt peak-to-peak.

Some recorders use special time code channels called "Address-Track" (video) or "Center-Channel Time Code" (audio) which eliminate the need for using an audio channel for the code. These tracks are optimized for time code recording and playback and do not require setting of the input or output signal levels. In these cases follow the directions of the manufacturer.

READING TIME CODE

Playing back time code is as simple as routing the recorder "audio out" with the time code signal, to the input of a reader or synchronizer. If there is an output level control, it should be set to match the input signal, i.e., a -3vu signal in should be -3vu out.

Problems with reading time code relate primarily to the recovery of the signal at other than "play" speed. The shape of the time code waveform was specified so that standard audio amplifiers having a frequency response of 50hz to 15khz could be used to record and reproduce the signal. If the tape speed is increased, even to only twice play speed, the standard audio amplifier will not accurately reproduce the signal due to high frequency bandwidth limiting.

Transports which reproduce time code at speeds greater than play require special amplifiers. At 50 times play speed the time code signal extends close to 1.0Mhz

At speeds slower than play, the greatest problem is waveform distortion. Time code readers can usually decode

the signal accurately down to 1/5th play speed (Adams-Smith time code readers can usually decode the signal accurately down to 1/20th play speed).

Most audio transports are designed to lift the tape off the heads when in high speed shuttle. When this occurs, the time code signal as well as any audio channels are muted.

To compensate for defective or missing code at other than play speed, synchronizers automatically count control track or tach pulses to maintain the address reference.

Sections 4 and 5 of this manual describe in detail the correct procedures for recording, decoding and synchronizing using the ZETA-THREE.

Appendix B
Synchronization

Synchronization is a process which forces two or more audio or video transports to run at a precise speed and position in relation to each other. The device used to perform these functions is a tape synchronizer. In common usage, "synchronization" is considered to be taking place while the synchronizer is in the act of achieving this precise relationship, or "synchronism". When synchronism has been achieved, the transports are said to be running in synchronism, in "sync", synchronized, "locked up", or in "lock".

Running multiple transports "in sync" permits time- and event-related material to be listened to and/or viewed simultaneously in their correct relationship.

For example, when preparing to build an audio track to complement an edited video tape, it is standard practice to first transfer the dialog audio from the video tape to a multi-track audio recorder. The transports carrying the picture and sound, respectively, can then be run in sync while additional dialog, sound effects or music is recorded on the multi-track recorder. This material might come from a tape played on an additional audio transport, which is itself synchronized to one of the other transports.

After the track-building procedure (sometimes called "audio editing") has been completed, the audio material on the multi-track recorder is usually mixed down to make a single or stereo audio track. This can be done with both the multi-track (which is now a "source" transport) and a smaller "record" audio transport synchronized to the same video play-back transport (carrying the same "work-print" of the previously edited video material) a process sometimes referred to as "mix-to-picture". Finally, if the transport carrying the completed audio mix is synchronized to the video transport carrying the video edit master, the mix can be recorded back onto the video tape's audio channels again, in what is often referred to as a "lay-back" session.

Another application of synchronization is to make larger transports out of smaller ones (increase the number of available tracks). For example, two 8-track recorders, when synchronized, perform exactly like a 16-track recorder; three synchronized 24-track recorders operate as a 72-track recorder. There is no limitation to the number of transports which can be made to operate in sync.

In order for a tape synchronizer to do its job it must be able to identify locations on both tapes with a high degree of precision. Longitudinal time code (LTC), a serial

digital code containing signals which can be handled by audio amplifiers at play speed, is now used almost exclusively for this purpose. See Appendix A for an explanation of time code.

Synchronizers use two techniques for reading longitudinal time code from tape.

The first technique requires the use of a wide-speed-range LTC reader to read the code (Adams-Smith wide-speed-range LTC readers will recover LTC at all speeds from about 1/20 play speed to 100 times play speed). Additionally, the tape transport carrying the time-coded tape must be capable of reproducing the time code at those speeds. One-inch C-format VTR's and some VCRs contain wide-band address tracks and heads which can provide continuous code. Most VCRs do not have an address track, and may not even keep the tape against the audio heads when in FF and RW. These types of VCRs must be modified by the user to use this first technique. Almost all audio recorders (the exception being the new center-track time code track transports) must be modified by wide-banding the playback amplifier of the track on which time code is recorded. Additionally, the tape lifters must be defeated to allow the tape to remain against the heads while in Fast-Forward and Rewind.

The second technique used by synchronizers for reading time code requires only a play speed LTC reader. At other speeds, the time code is updated using tach pulses and direction signals from audio recorders, or control track and direction tally signals from VTRs. Since all recorders can reproduce LTC at playspeed, no recorder modifications are required, and the tape lifters of audio recorders can be allowed to operate normally. Since the second technique is less costly, and does not require recorder modification, it is almost always used in audio-for-video post-production even though tape cueing and parking may not be quite as accurate as when reading LTC at all speeds, because of tape slippage and tach errors.

Normally, it is only necessary to read LTC at all times (the first technique) when a tape contains discontinuous code, as it may if it is an original source tape. Time code discontinuities occur in source tapes if the time code generator is kept running continuously, perhaps to record time-of-day time code, instead of being stopped when recording stops and started again when recording starts again.

Before synchronizing can begin, the synchronizer must be told what positional relationship exists between the master and slave time code addresses. This relationship, called the "offset" is calculated by selecting a pair of

corresponding addresses on the two tapes and subtracting the master address from the slave address.

For instance, assume a video transport has been designated as the master in a synchronizing system, and an audio transport has been designated as the slave (a lay-back configuration). If the time code address of a door slamming, as seen on video tape, is one hour and 20 minutes, and the corresponding sound of the door slam on audio tape is one hour, 20 minutes and 15 seconds, then the offset is 15 seconds, obtained by subtracting the master address from the slave address.

In post-production operations, the offset between the master video tape and the slave multi-track audio tape is often deliberately made zero for convenience This is accomplished by simultaneously copying the time code along with the audio program material from the video recorder to the multi-track audio recorder. In situations where the offset between the material on the video and multi-track recorder is not zero, then the offset value must be entered by the user into the synchronizer. Offsets may be known as a result of prior work and entered into the synchronizer, or captured from incoming master and slave time codes while the tapes are running in sync; or an individual sync point on each tape may be captured or entered, with the synchronizer automatically calculating the offset.

Using time-code from each transport as speed and position references, a synchronizer first achieves, and then maintains, a zero offset error.

There are three distinct routines in the synchronizing process. Assume that the tapes have been played, at least momentarily, and the synchronizer knows their locations and status (stopped). When the "synchronize" (or "chase") command is given, the first step begins. The synchronizer compares the MTC and STC addresses, computes the current offset, compares it to the desired offset, and determines the offset error. The synchronizer will then issue motion commands to the slave transport in order to cause the quickest reduction of the offset error. The transport will then be commanded to stop or "park"

When the master tape is put into Play, the second routine begins. In this routine the capstan of the slave transport is controlled by the synchronizer to make the slave tape move faster or slower than play speed, as necessary, to reduce the offset error to zero. Achieving lock from park usually takes no more than a very few seconds.

Once synchronism has been achieved, the synchronizer minutely speeds up or slows down the slave transport to

maintain a zero offset error. The accuracy of synchronism must be on the order of microseconds, equivalent to 1/1000 of a TV frame and stable enough not to add any audible distortions.

Additional information on the process of synchronization is available throughout the ZETA-THREE manual.

Appendix C
Cable and Back Panel Information

This chapter describes cable connections for each of the input/output jacks found on the back panel of the ZETA-THREE, see Figure C.1. Each section gives a brief summary of the purpose of a particular jack and provides information for the type of cables needed for proper connections to the jack.

C.1 POWER MAINS [34]

Mains power is fed to the ZETA THREE through a standard 3-wire EIA power cord, which is supplied with the unit.

C.2 SLAVE TRANSPORT [32]

The SLAVE TRANSPORT connector [32] is responsible for monitoring the state of a transport, i.e. if the machine is in Play Fast forward, or Rewind and also sends commands to control the transport-s actions so that the slave will chase and synchronize to the master. Tach or Control Track pulses are observed through this connector.

1	CAPSTAN Common
2	CAPSTAN Normally Closed
3	STOP
4	FF
5	REW
6	PLAY
7	GP1/PAUSE/LIFTER DEFEAT
8	GP2/LOCK/MUTE
9	GP3/REHEARSE
10	REC
11	REC Tally
12	DIRECTION
13	CTL/TACH
14	GND Sense
15	DC
16	FM
17	+5V DC
18	CAPSTAN Normally Open
19	Command Common
20	GP1 Common
21	GP2 Common
22	GP3 Common
23	SPARE Tally
24	--
25	GND

C.3 SLAVE CODE IN [31]

The SLAVE CODE IN jack [31] accepts longitudinal time code signals. Connections are:

Pin 1: signal ground
Pin 2 and Pin 3: differential input
Shell chassis ground

See Figure C.1 for further detail.

C.4 MASTER TRANSPORT [30]

The Master transport connector [30] is responsible for monitoring the state of a transport, i.e. if the machine is in Play, Fast forward, or Rewind. Tach or Control Track pulses are observed through this connector.

The pin assignments are listed below

1)	
)Pins 1 and 2 tied.	
2)	
3	STOP
4	FF
5	REW
6	PLAY
7	GP1/PAUSE/LIFTER DEFEAT
8	GP2/MUTE
9	GP3/REHEARSE
10	REC
11	REC Tally
12	DIRECTION
13	CTL/TACH
14	GND Sense
15	-
16	
17	+5V DC
18	-
19	Command Common
20	GP1 Common
21	GP2 Common
22	GP3 Common
23	SPARE Tally
24	-
25	GND

C.5 MASTER CODE IN [29]

The MASTER CODE IN jack [29] accepts longitudinal time code signals. Connections are:

Pin 1: signal ground
Pin 2 and Pin 3: differential input
Shell: chassis ground

See Figure C.1 for further detail.

C.6 GEN CODE OUT [28]

Generator time code connections from the ZETA-THREE should be made with shielded, twisted-pair audio cable using type XLR-3 connectors.

Signals are:

Pin 1: signal ground
Pin 2 and Pin 3: balanced output
Shell: Chassis ground

See Figure C.1 for further detail.

C.7 REMOTE [27]

Reserved for Adams-Smith use.

C.8 CONTROL [26]

The CONTROL connector [26] is an RS-422 serial control port. It provides control for transports which require serial commands. Connections are as follows:

1	FRAME GROUND		
		6	RECEIVE Common
2	RECEIVE "A"		
		7	RECEIVE "B"
3	TRANSMIT "B"		
		8	TRANSMIT "A"
4	TRANSMIT Common		
		9	FRAME GROUND
5	$\overline{\text{AUXP}}$		

C.9 COMPUTER [25]

This connector provides serial communications for external computers.

When using RS-422, all connections are as follows:

1	FRAME GROUND		
		6	RECEIVE Common
2	RECEIVE "A"		
		7	RECEIVE "B"
3	TRANSMIT "B"		
		8	TRANSMIT "A"
4	TRANSMIT Common		
		9	FRAME GROUND
5	$\overline{\text{AUXP}}$		

When using RS-232, connections are as follows:

1	-		
		6	DATA SET READY (DSR)
2	RECEIVED DATA (RXD)		
		7	REQUEST TO SEND (RTS)
3	TRANSMIT DATA (TXD)		
		8	CLEAR TO SEND (CTS)
4	DATA TERMINAL READY (DTR)		
		9	-
5	SIGNAL GROUND		

NOTES

1) DATA TERMINAL READY (Pin 4) is always asserted, as long as the ZETA-THREE is powered-up.

2) DATA SET READY (DSR), when asserted, enables the ZETA-THREE receive channel. If DSR is not connected, then it will be asserted internally.

3) CLEAR TO SEND (CTS), when asserted enables the ZETA-THREE to transmit data. If not connected, then it will be asserted internally and the ZETA-THREE will transmit without restriction.

4) REQUEST TO SEND (RTS) should be connected to the external computer CLEAR TO SEND (CTS) input. When asserted, the RTS output indicates that the ZETA-THREE is ready to receive data (i.e. ther is plenty of room in the data buffer of the ZETA-THREE).

If RTS is not asserted, then the external computer should refrain from further transmissions, as the ZETA-THREE buffer is about to overflow.

C.11 AUX IN [18]

The AUX IN is a 1/4 inch stereo jack and requires a 1/4 inch tip sleeve phone plug (tip-signal, sleeve-ground).

The AUX IN jack will accept a square wave or pulse waveform (1 8 volts) as a frame rate when resolving.

C.12 AUX OUT [19]

The AUX OUT jack [19] is a stereo jack and requires a 1/4 inch tip ring sleeve phone plug.

The tip of the plug will normally be responsible for sending a 5V pulse to trigger EVENT 1, OR it can be assigned a drum frequency (MIDI timebase signal).

The ring of the plug will normally be responsible output a 5V pulse to trigger EVENT 2, OR will output a STP/CNT, or FRAME.

G01 PRESET 01000000

Displays the time code address which will appear in the G_TC Selection display when the CAPTURE key is pressed. The number set into Menu number G01 must be a valid time code number.

G02 COPY MODE=JAM
XFER

When set to JAM, the ZETA-THREE's time code generator, when COPYing is taking place, will continue to generate consecutive time code numbers when the copy source time code stops.

When set to TRANSFER, the generator will stop generating the time code sequences as soon as the copy source time code stops.

G03 COPY=TC AND UB'S
TIME CODE
USER BITS
OFF

Defines what the ZETA-THREE generator will generate when COPYing.

When set to TC AND UB'S it will copy both the time code and the user bits from the copy source.

When set to TIME CODE, will copy only time code from the copy source, and will generate whatever is in the G_UB Selection display as user bits.

When set to USER BITS, it will copy only the USER BITS from the copy source, and will generate time code starting from the address in the G_TC Selection display.

When set to OFF, neither time code nor user bits can be copied.

G04 CONSTANTS ->

Time code generator constants. Used for special setups only.

MØ1 MASTER = READER
GENERATOR

Determines where the ZETA-THREE expects to find Master time code.

When set to READER, the Master time code is the time code being fed to the ZETA-THREE via its MASTER CODE IN jack. All enabled slave devices will be synchronized to this time code.

When set to GENERATOR, the Master time code is the time code which is coming from the ZETA-THREE's time code Generator. All enabled slave devices will synchronize to this time code.

MØ2 OUTPUT = TRANSPORT
OFF
EV-3 1Ø

Certain pins in the MASTER TRANSPORT connector (selected through Master Constants) can be used to send signals to control different functions. This menu informs the ZETA THREE of the type of output that will be sent to individual pins.

When set to TRANSPORT, the ZETA-THREE will send transport control signals (such as Stop, Play FF, and RW commands) to pins on the MASTER TRANSPORT connector.

When set to OFF, no signals will be sent to command output pins on the MASTER TRANSPORT connector.

When set to EV-3-1Ø, the ZETA-THREE will expand the Events Selections to include up to 1Ø events. The events will operate in the same manner as events 1 and 2 (which are triggered through the AUX OUT jack) but, in this case, Master Transport connector pins will be used to trigger Events.

MØ3 LIMIT ------ -

Enter the last time code address of the Master tape and the ZETA-THREE will automatically stop the transport when that address is reached This feature prevents the tape from accidentally running to the end of the reel such as when a wrong offset is entered

M04 TRANSPORT ->

CAPTURE-ing a name from the list provided in the higher level menu informs the ZETA-THREE of what make and model of transport is connected to the MASTER TRANSPORT connector.

M05 SAVE TRANSPORT->

Allows modified transport Constants to be saved in one of the 10 files available in this Menu item. When Constant values for a Master transport are changed through Menu M06, the values will be saved when the CAPTURE key is pressed. The saved Constants can then be loaded for controlling a transport by CAPTUREing it as a User-defined transport from the M04 transport list.

M06 CONSTANTS ->

Master Transport constants. Used for special setups only.

S01 LOCK MODE = ADR
FWL
AUTO

Determines how the ZETA-THREE will synchronize all ENABLEd transports.

When set to ADR (Address mode), the ZETA-THREE will synchronize the time code address of the slave tape to the time code address of the master tape (plus or minus the offset).

When set to FWL (Freewheel), the ZETA-THREE will lock the nearest frame edge on the slave tape to the nearest frame edge on the master tape and ignore their respective time code numbers.

When set to AUTO, the ZETA-THREE will lock by reading time code numbers, as it does when set to Address mode, but, as soon as synchronization is achieved, the ZETA-THREE switches to Freewheel and ignores the time code numbers on the tapes. The ZETA-THREE will read time code whenever synchronous running ceases.

S02 SLOW RELOCK = OFF
ON

When set to ON, allows a sub-audible relock to eliminate offset errors of less than one frame, such as might be incurred by editing. This feature allows the ZETA-THREE to "catch-up" to the correct time code offset without re-cuing the transport.

S03 SPLICE TRAP = OFF
ON

The Splice-Trap routine should be considered a means of "rescuing" a play-through of tapes which have been physically spliced due to tape breakage or damage, where the splice occurs in a section which contains critical material.

When set to ON, the ZETA-THREE will recognize the time code discontinuity and immediately recapture the offset using the next available valid time code address.

S04 LIMIT --------

Enter the last time code address of the Slave tape and the ZETA-THREE will automatically stop the transport when that address is reached. This feature prevents the tape from accidentally running to the end of the reel such as when a wrong offset is entered.

S05 TRANSPORT ->

CAPTURE-ing a name from the list provided in the higher level menu informs the ZETA-THREE of what make and model of transport is connected to the SLAVE TRANSPORT connector.

S06 SAVE TRANSPORT->

Allows modified transport Constants to be saved in one of the 10 files available in this Menu item. When Constant values for a Master transport are changed through Menu M06, the values will be saved when the CAPTURE key is pressed. The saved Constants can then be loaded for controlling a transport by CAPTUREing it as a User-defined transport from the M04 transport list.

S07 CONSTANTS ->

Slave transport Constants. Used for advanced special setups only.

Z01 IN/OUT = RECORD
REHEARSE

When set to RECORD, the ZETA-THREE will send a Record command to the Master and/or Slave transport when the left-most ENABLEd transport plays the Z_IN time code address, and a Stop Record command when the Z_OUT address is played.

When set to REHEARSE, the ZETA-THREE send a Rehearse command to the Master and/or Slave transport when the left-most ENABLEd transport plays the Z_IN time code address, and a Stop Rehearse command when the Z_OUT address is played. Only transports which are equipped with Rehearse capabilities can take advantage of Z01 IN/OUT=REHEARSE.

Z02 PREROLL = 00 00

Preroll is a number entered (in seconds and frames) which the ZETA-THREE uses when using the GOTO function. When the GOTO command is issued, the ZETA-THREE will move the tape to the number in the Z_GO Selection display minus the value in the Z01 display.

Z03 LOOP = OFF
CYCLE
AUTO REWIND
AUTO STOP

LOOP allows a certain section of tape to be played over and over again.

When set to CYCLE, the left-most ENABLEd transport, upon reaching an End point, will GO TO the cue point. The transport will park, and then, after any ENABLEd slaves have also cued and parked, will return to Play. The transport will play until it reaches the End point and repeat the sequence.

When set to AUTO REW, the left-most ENABLEd transport, upon reaching an End point, will GO TO the cue point and park. Press STOP/CONT to repeat the sequence.

When set to AUTO STOP, the left-most ENABLEd transport, upon reaching an End point will stop.

Z04 FRAMES = 30.00
24
25
29.97
29.97 DF

Selects the frame rate which the ZETA-THREE will use as the system standard.

Z05 SYSTEM = UNRESOLVED
RESOLVED

When set to "UNRESOLVED", the ZETA-THREE will run referenced to its own internal clock.

When set to "RESOLVED", the ZETA-THREE will run resolved to an external reference.

Z06 RESOLVE = VIDEO
MAINS
AUX IN

In effect only when menu Z05 is set to "RESOLVE". Informs the ZETA-THREE of the type of external sync it will resolve to.

When set to VIDEO, the system will resolve to the signal being sent into the VIDEO IN jack [20 or 21].

When set to MAINS, the ZETA THREE will use the frequency set by the power mains to resolve the system.

When set to AUX IN, the system will be resolved to the frequency received through the AUX IN jack.

Z07 COMPUTER PORT ->

Press MENU key again to enter the list of possible serial formats for the Computer port. Press CAPTURE when the desired protocol type appears on the display.

Z08 CONTROL PORT ->

The Control Port [26] is used for transports which require serial control. Press MENU key again to enter the list of possible transports. Press CAPTURE when the desired transport appears on the display.

Z09 SYS ADDRESS = 0000

This provides the ZETA-THREE with a unique address when used for serial communication.

Z10 XOUT TIP = EVENT 1
TIMBASE

When set to EVENT 1 the ZETA-THREE will trigger the tip of the AUX OUT jack when time code from the left-most ENABLEd transport reaches the address in the E_01 Selection display.

When set to TIMBASE a square wave signal will be sent out through the tip of the AUX OUT jack at a frequency locked to MIDI sync pulses.

Z11 XOUT RING = EVENT 2
STP/CNT
FRAME

When set to EVENT 2, the ZETA-THREE will trigger the ring of the AUX OUT jack when time code from the left most ENABLEd transport reaches the address in the E_02 Selection display.

When set to STP/CNT... (MIDI function)

When set to FRAME... (MIDI function)

Z12 DATA TRANSFER ->

Allows BEATMAP and MIDI information to be saved onto a tape, and allows that same information to be re-loaded from the tape back into the ZETA-THREE.

Z13 CLEAR SELECTIONS

Pressing the CAPTURE key while this menu is on the display will cause the ZETA-THREE to CLEAR all System (Z) and Event (E) Selections, and the Slave and MIDI Offset

Selections. This feature is convenient when switching work tapes, allowing easier entry of new information.

EØ1 EV-Ø1 = DISARMED
ARMED

When ARMED, the pulse which triggers Event 1, will occur when the Event point is reached.

When DISARMED, the event will not occur.

EV-Ø2 = DISARMED
ARMED

When ARMED, the pulse which triggers Event 2, will occur when the Event point is reached.

When DISARMED the event will not occur.

EØ2 EVENT CONSTS ->

Event Contants. Allows modification of Event signal.

Appendix E

ADAMS-SMITH ZETA THREE SLAVE TRANSPORT CONSTANTS

(Master Transport uses a subset of these)

Software Rev 3.60 (standard) and 3E60 (emulator)
Document revised Oct 29, 1990

Constant Number		Typ. Val.	Description
01	lsd	x6	Number (0 thru F) of consecutive, contiguous frames during which offset errors greater than the "wide lock window" width (Constant 06 msd) must occur before re-synchronizing will begin, after lock has been achieved. Setting 01 lsd too low will cause unnecessary re-synchronizing when occasional bursts of corrupted time code occur (assuming this time code has passed all other tests).
01	msd	6x	Capstan FM frequency final divisor code. Used in conjunction with constants 03 (msd,lsd) and 04 (lsd only) to establish the capstan FM servo output center frequency. (See Constant 03 for formula). One special case (0x) specifies Constants 45 and 46 as the final divisor.

The following table shows the Final Divisor for each setting of this constant, as well as the final capstan frequencies ASSUMING THAT CONSTANTS 03 AND 04 CONTAIN x823:

Val.	Divisor	Frequency (hz)
0x	[Constants 45,46]	[38400/(Constants 45,46)]
1x	512	75
2x	32	1200
3x	16	2400
4x	16	2400
5x	8	4800
6x	4	9600
7x	2	19200

02	lsd	x5	Lock servo coefficient K2. Weights the running sum of offset errors to optimize damping while the slave is synchronizing. Set to the smallest value (0 thru F) which does not cause the slave transport to oscillate during final synchronization.
02	msd	3x	Lock servo coefficient K1. Weights present positional error of offset to optimize final synchronizing time. Set to the smallest value (0 thru F) which does not cause the slave transport to overshoot or oscillate about its lock position.
03	both	23	Used in conjunction with Constant 04 lsd to form the Main Divisor for the capstan center frequency. Typically, if this constant contains 23, and Constant 04 lsd contains x8, then the divisor value is 823 hexidecimal. The formula for deriving the output capstan frequency (in hertz) is: F = 80000000 / Main Divisor / Final Divisor where the Final Divisor is established by Constant 01 msd.
04	lsd	x8	Part of the Main Divisor for the capstan center frequency (see Constant 03)..
04	msd	0x	Capstan relay switching mode: 0x Capstan Relay is ON (with the Zeta providing the Capstan frequency or voltage) only while the slave is enabled. This is the NORMAL capstan relay mode. 1x ALWAYS ON - Otari MX-5050 mode 2x OFF if not PLAYing or not LOCKing 4x OFF after lock achieved - VCR mode [6x Sony PCM-3324 mode. This is the combination of modes 2x and 4x.] See also constant 38 MSD.
05	lsd	x2	Limit for error contribution to lock speed change. Set to the highest value which adequately accomodates speed changes (varispeed) without incurring unacceptable overshoot.
05	msd	4x	Capstan speed control range. i.e. The percentage limit on capstan speed relative to the center frequency. The range is specified in 3% increments, so that, for example, 4x specifies +/-12%, Fx (hexadecimal for 15) specifies +/-45%, etc.
06	lsd	x3	"Narrow lock window" width. (Unit = .001 frame)
06	msd	1x	"Wide lock window" width. (Unit = .05 frame)

07	lsd	x1	x0 to x7	Number of retries that will be attempted if Play-to-cue-point should fail.
			x8 to xF	Same as x0 to x7, but velocity projections will not be used in Play-to-cue stopping-window calculations.
07	msd	0x	Play-to-cue-point. (Unit = 1 sec.)	
			0x	No play-to-cue.
			1x to 7x	Play-to-cue enabled. When cueing, the transport will cue to a point which is the number of seconds specified here (0 thru 7) before the target cue point, stop, play up to the cue point, and stop. Useful if the transport tach or control track is unreliable.
			8x	Same as 0x
			9x to Fx	Same as 1x to 7x, but a simpler stopping algorithm is used.
08	lsd	x2	Cue oscillation limit.	
			x0 to x7	The number (0 thru 7) of oscillations around the cue point which are permitted while the synchronizer is attempting to stop or park at the cue point. If and when this count expires, the synchronizer "gives up" and issues a Stop command. Note: Movements across the cue point in the forward direction only are counted.
			x8 to 0F	Same as x0 to x7, but the synchronizer will never repeat a Fast Forward or Rewind command (good for transports which enter spooling mode when these commands are issued twice in a row).
08	msd	0x	Pause/Stop/Play command configurations for VCR's	
			0x	Normal non-VCR.
			1x	Pause:Pause instead of Stop:Play.
			2x	Pause:Play instead of Stop:Play.
			4x	Never repeat a Stop command (probably wired to the VCR's Pause input). This may be combined with 0x,1x,2x,9x,Ax making 4x,5x,6x,Dx,Ex respectively.
			9x or Ax	Same as 0x or 1x, but Stop command will not be issued before entering wind modes (avoids unlacing).
09	lsd	x0	Motion sense etc. (bits may be combined)	
			x1	Wait for stop motion between commands
			x2	Do not wait for stop motion before issuing Play
			x4	Defeat time code reading over 2x play speed
			x8	Do not use the presence of time code to indicate that a VCR is laced.
09	msd	0x	Cueing type (for cueing parameters see Constants 11-18)	
			0x	Normal fast forward / rewind style cueing.
			Cx	Serial cueing (via Control Port).
			Ex	Special JVC DS-DT900 R-DAT cueing.
			Fx	Special Fostex D-20 R-DAT cueing.

10 both 03 Cueing constant.

00 to 0F Simple K factor for 'KV squared' cueing algorithm.
The value here weights the influence of tape velocity on determination of the point at which the opposite direction command (i.e. Fast Forward or Rewind) is issued during cueing and parking.
Set to the largest value which produces acceptably low transport overshoot.
(Typical values are x1 thru x5).
Note: Values xD, xE, xF are special "negative" range values, and can be thought of as -3, -2, and -1.

80 to FF Higher resolution K factor which may be used when the simple K factor is between F (-1) and 4. Correspondance between the two K scales is as follows:

Simple K	Hi-res K
4	81
3	82
2	84
1	88
0	90
F	9F

11-18 both ?? Specific cueing parameters. Definitions will change according to the setting of constant 09 msd (Cueing type).

Normal cueing [Constant 09 = 0x]:

11 both 00 Optional separate window specification for Play-to-cue (fast wind stopping window before entering play).
If non-zero, and constant 12 is zero, then stopping window will be +/- this value (normally window is +/- the play-to-cue value).
msd = seconds; lsd = seconds/16

12 both 00 Optional separate negative window specification for Play-to-cue.
If non-zero, and if constant 11 is also non-zero, then the Play-to-cue stopping window will be +(constant 11) / -(constant 12).
msd = seconds; lsd = seconds/16

Serial cueing (via Control Port) [Constant 09 = Cx]:

Note: Serial cueing will only be used following "specific" requests to cue the transport unless either/both of the following are set:

11 lsd x0 Set to "C" if serial cueing is to be used during "locate to master position" operations in addition to "specific" cueing requests.

11 msd 0x Set to "C" if serial cueing is to be used during "chase" cueing operations in addition to "specific" cueing requests.

Special JVC DS-DT900 R-DAT cueing [Constant 09 = Ex]:
See Fostex R-DAT constants 11 thru 14.

Special Fostex D-20 R-DAT cueing [Constant 09 = Fx]:

11	both	10	Stopping allowance - ahead of cue point msd = seconds; lsd = seconds/16
12	both	20	Stopping allowance - behind cue point msd = seconds; lsd = seconds/16
13	both	15	Distance from cue point threshold for transition from slow to fast wind msd = minutes; lsd = minutes/16
14	both	0E	Distance from cue point threshold for transition from fast to slow wind msd = minutes; lsd = minutes/16
15	both	09	Delay before accepting new FF/REW/PLAY tallies (unit=seconds/120, range=00-FF)

19	both	00	Jog/Shuttle type: 00 None supported 01 FF/Rew toggling 02 FM frequency speed control 03 DC speed control 0A Special Panasonic VCR speeds 0C Serially controlled (via Control Port) F0 FF/Rew toggle Test mode
20-32	both	??	Multipurpose Jog/Shuttle parameters. Definitions will change according to the setting of constant 19 (Jog/Shuttle type). For setup procedures, refer to Technical Bulletin #10, "Optimizing Jog/Shuttle Constants".

FF/Rew Toggling [Constant 19 = 01]:

20	both	20	Period of FF/Rew oscillation. (Hexadecimal, unit=frames/16)
21	both	00	not used
22	both	01	Minimum duty cycle; i.e. minimum time spent in either FF or Rewind. (Hexadecimal, unit=frames/16)
23	both	00	not used
24	msd	0x	not used
24	lsd	x6	Acceleration error clamp; lower values produce more clamping; more clamping may produce smoother but slower reaction to changes in acceleration.
25	msd	Cx	Parameter "M"; final multiplier/divisor for duty cycle adjustment (0=multiply by 1; 1 up thru 7 produce increasingly smaller adjustments; F down thru 8 produce increasingly larger adjustments).
25	lsd	x6	Parameter "K"; larger values increasingly damp the acceleration error contribution.
26	both	38	Maximum velocity of transport at full wind. (Hexadecimal, where 03h=playspeed)
27	both	0A	Start up boost value.

FM Frequency Speed Control [Constant 19 = 02]:

20	lsd	x0	Reverse flag:
		x1	FM jog/shuttle not available in reverse.
20	msd	6x	FM jog/shuttle operating flags; combine hexadecimal values as required:
		2x	Hold capstan relay on during FM jog/shuttle.
		4x	Switch to normal FF/Rew commands if requested jog/shuttle velocity is greater than the maximum available velocity (as specified in constants 27 and 28).
21,22	both	00,00	Minimum available jog/shuttle velocity low byte, high byte. (Hexadecimal, where 0300h=playspeed) Example: values 23,01 would produce a combined hexadecimal value of 0123h.
23,24	both	6A,08	Main center frequency divisor low byte, high byte (similar to constants 03 and 04).
25,26	both	06,00	Playspeed final hardware divisor low byte, high byte (similar to constants 45 and 46).
27,28	both	00,0C	Maximum available jog/shuttle velocity low byte, high byte. (Hexadecimal, where 0300h=playspeed)

DC Speed Control [Constant 19 = 03]:

20	lsd	x0	DC jog/shuttle Test Modes:
			x0 — Test Mode off
			x1 — Set Span (constants 23,24) to 0FFFh and adjust Offset (constants 21,22) for zero velocity.
			x2 — Adjust Span (constants 23,24) to just attain maximum velocity; read and later set maximum velocity in constants 27 and 28.
			x3 — Adjust constants 25,26 for playspeed jog/shuttle.
20	msd	Cx	DC jog/shuttle operating flags; combine hexadecimal values as required:
			1x — Use bipolar DC range.
			2x — Hold capstan relay on during DC jog/shuttle.
			4x — Switch to normal FF/Rew commands if requested jog/shuttle velocity is greater than the maximum available velocity (as specified in constants 27 and 28).
			8x — Invert DC polarity.
21,22	both	4A,0A	DC Offset value for zero velocity - low byte, high byte. (Hexadecimal, range = 0000h-0FFFh).
23,24	both	C0,09	DC Span low byte, high byte (similar to constants 98 and 99).
25,26	both	FB,02	DC Playspeed value low byte, high byte.
27,28	both	00,0F	Maximum available jog/shuttle velocity low byte, high byte. (Hexadecimal, where 0300h=playspeed)

Special Panasonic VCR Speeds [Constant 19 = 0A]:

20	both	24	Interface cable number

Serially Controlled (via Control Port) [Constant 19 = 0C]:

No parameters used.

FF/Rew Toggle Test Mode [Constant 19 = F0]:

20	both	20	Period of FF/Rew oscillation. (Hexadecimal, unit=frames/16)
21	both	10	Test duty cycle (portion of period spent on FF).
22	both	01	Minimum duty cycle; i.e. minimum time spent in either FF or Rewind. (Hexadecimal, unit=frames/16)

33	lsd	x0	[not used]
33	msd	0x	Subframe splice trap request 0x — Normal Fx — Subframe splice trap. In this mode, the synchronizer will allow the slave transport to run locked even when there is a "subframe" error (i.e. an error of less than a frame). Mostly used for video decks whose time code sync word position on tape will typically be offset by a fraction of a frame from the reference video.
34	both	29	Lock routine exit error. Determines the size of the lock error (unit = frames) which must occur before the synchronizer will switch from capstan speed control to Fast Forward / Rewind cueing in order to try to correct the error.
35	lsd	x6	Transport command pulse output duration. (Unit = 8.3 msec)
35	msd	6x	Transport command repeat count. i.e. The number of times that the synchronizer is prepared to repeat a transport command until the desired action is achieved. If the transport has not responded after the last repeat, a Stop command is issued (except if 0x is entered).
36	lsd	x0	Determines percentage of playspeed above which changes in direction will not be recognized. Typically used for direction sensing when a Rewind tally is used as a direction tally. x0 — No special direction sensing threshold x1 — 62% x2 — 50% x3 — 40% x4 — 32% x5 — 25% x6 — 20% x7 — 16%
36	msd	0x	Stopping allowance for Play-to-cue (unit=frames, range=0-F). Not used during chase or locate cueing.

37	lsd	x0	Smart-start defeat flags x0 Smart-Start routine enabled. (Normal) x1 Defeat Smart-start. x2 Use fixed Smart-start offset. x4 Smart-start may be adjusted to Master Tach or Control Track data i.e. not necessary to wait for time code. x8 Smart-start adjust to Slave Tach / Control Track is ok.
37	msd	2x	Park-ahead. i.e. The number of seconds (0 thru 7) that the slave will park ahead of a playing master when cueing up in order to synchronize to the already playing master.
38	lsd	x0	AUTO lock switchover delay (unit = secs). When LOCK MODE=AUTO, the synchronizer will initially lock the slave in ADDRESS mode, switching to FREEWHEEL mode the number of seconds specified here after lock occurs. Note: Special value xF means "switch immediately after lock is achieved".
38	msd	0x	Special AUTO override modes (bits may be combined) . . 8x Override the LOCK MODE menu setting and force AUTO mode. 4x The same as the 8x setting, but the CAPSTAN RELAY WILL BE DISABLED AFTER THE AUTO SWITCHOVER. Actually a variation of the Constant 04 msd capstan relay switching functions. Useful for digital dubbing where two digital recorders are "word-clock-locked" with the possiblity that their time code tracks will "drift" relative to each other. This results of course from incorrect time code striping. To avoid such situations, both digital tapes must be striped with the same time code WHILE THEY ARE WORD-CLOCK-LOCKED. 1x Inhibit slave resolve. When in resolve mode, the Zeta will normally try to resolve the slave transport whenever it is placed into play mode by itself (master disabled). This is unnecessary for a video slave which is referenced to the same video sync as the Zeta.
39	lsd	x5	Failure Sense play mode timeout (unit=frames). If the transport drops out of play mode for longer than this time period then the transport is judged to have failed. Failure Sense mode must first be invoked.
39	msd	0x	Fast mode flag: 1x Use only Fast Forward and Rewind pins when executing FF/Rew commands. In normal operation, Forward Hold, Reverse Hold and Search pins are also asserted.

41	both	29	Park mode exit criterion (unit = frames). If a slave transport fails to park within this programmed window after chasing a master, it will re-cue and attempt to re-park within the window, until the number of oscillations around the cue point (set by Constant 08 lsd) have occured.
42,43		[not used]	
44	both	50	Varispeed limit (unit = percent of playspeed)
45	both	00	Used in conjunction with Constant 46 lsd to form an alternative Final Divisor for the capstan center frequency. (Constants 45 and 46 are used ONLY IF Constant 01 msd = 0. See Constant 01.) If this constant contains 23, and Constant 04 lsd contains x1, then the divisor value is 123 hexidecimal.
46	lsd	x0	Part of the alternative Final Divisor for the capstan center frequency (see Constant 45).
46	msd	0x	[not used]

NOTE:
Constants 47 thru 81 contain hex values which map various functions to the command output pins of the Master/Slave transport connectors (plus the Slave capstan relay, and occasionally Aux Out tip/ring).

The connector pins are defined as follows . .

Map Value	Pin Name	Pin Number	Common Pin Number
01	Stop	3	19
02	FF	4	19
04	Rew	5	19
08	Play	6	19
10	GP1	7	20
20	GP2	8	21
40	GP3	9	22
80	Rec	10	19
F1	Event 1	(Aux out tip)	
F2	Event 2	(Aux out ring)	
F4	Capstan relay		

47	both	00	Connector command pins causing VCR to unlace
48	both	00	Connector command pins causing VCR to lace and move
49	both	00	Connector command pins causing VCR to lace and pause

50	both	01	Stop function connector pin map
51	both	00	Forward continuous (search only)
52	both	00	Reverse continuous (search only)
53	both	02	Fast forward
54	both	04	Rewind
55	both	08	Play
56	both	00	Pause
57	both	00	Search momentary
58	both	00	Search Continuous
59	both	00	Locked
60	both	00	Lock momentary
61	both	F4	Capstan
62	both	00	Bump up
63	both	00	Bump down
64	both	20	Mute
65	both	10	Lifter defeat
66	both	00	Record out overlap
67	both	08	Record out
68	both	00	Record continuous
69	both	88	Record in
70	both	00	Rehearse out overlap
71	both	00	Rehearse out
72	both	40	Rehearse continuous
73	both	00	Rehearse in
74	both	00	Record armed
75	both	00	Rehearse armed
76			[not used]
77	both	00	Alternate capstan
78	both	00	Parked (chase mode only)
79	both	00	Chase enabled
80	both	00	Capstan momentary
81	both	00	Standby (ready) command
82	both	00	Transport connector invert mask - pins selected here will become active high, normally low
83	lsd	x0	Invert mask for Capstan, Aux Out tip/ring x1 Invert Aux out tip x2 Invert Aux out ring x4 Invert Capstan relay sense
83	msd	4x	Time-Code-Only stop delay (unit = .133 sec) (i.e. when no tach is connected) 0x No special time-code-only delay

84	lsd	x0	Audio mute disables x0 Mute enabled x1 Disable mute during initial lock x2 Disable mute during high speed winding x3 Disable all muting
84	msd	0x	Delay from lifter release to un-mute (unit = .133 sec)
85	lsd	x0	Tally flags x1 Use Record tally x2 Invert sense of Spare tally (i.e. active high) x4 Invert sense of Record tally x8 Punch Out only if Record Tally is active; Punch In only if Record Tally is not active.
85	msd	0x	Use of Spare Tally 0x Disabled 1x Play tally 2x Pause tally 3x Eject tally 4x Standby (ready) tally Fx Special Fostex D-20 R-DAT tallies, where Zeta spare tally = Fostex play tally Zeta record tally = Fostex fast forward tally Zeta direction input = Fostex rewind tally.
86	lsd	x0	Standby (ready) command type (Constant 81) x0 Pulse on only (no action for "off") x1 Pulse toggles standby on/off. Availability of a Standby tally (see Constant 85 msd) will guarantee correct operation. x2 Continuous (output asserted when Standby = On).
86	msd	0x	Auto disarm / Rehearse flag 1x Auto disarm Record/rehearse after punch out 2x Rehearsal requires Transport to be playing
87	both	00	Record in advance (0-99, unit = .25 frame)
88	both	00	Record out advance
89	both	00	Rehearse in advance
90	both	00	Rehearse out advance
91	both	00	Tach rate - least significant hex byte
92	both	00	Tach rate - most significant byte (for values greater than 255)
93	both	00	Tach rate - fraction byte (xx/256)

94	lsd	x1	Lifter operation and Control-Track/Tach overrides x0 Lifters always defeated by Zeta x1 Allow normal Zeta lifter defeat operation x2 Zeta does not defeat lifters x3 As x1, but also sample high speed code x8 Control track rate override = 1 pulse per frame xC Control track rate override = 1 pulse per field These last two are normally combined with x2: xA No lifters, frame rate control track (tach) xE No lifters, field rate control track (tach)
94	msd	2x	Reader flags 1x Defeat time code 'Plus One' tests 2x Learn tach and direction 4x Use serial (Control Port) time code if available -- MASTER ONLY 8x Invert sense of tach direction tally
95	lsd	x0	Varispeed controls x0 None x1 Inhibit capstan during varispeed x2 Inhibit "bump" outputs during varispeed x3 Both x1 and x2
95	msd	0x	Slave Capstan DC polarity 0x Normal 8x Invert polarity of capstan DC output
96	both		Slave Capstan DC 'Offset' (Least significant digits)
97	lsd		Slave Capstan DC 'Offset' (Most significant digit)
97	msd		[not used]
98	both		Slave Capstan DC 'Span' (Least significant digits)
99	lsd		Slave Capstan DC 'Span' (Most significant digit)
99	msd		[not used]

Additional Constants for VTR Emulation (Zeta-Three^{em} only):

Constant Number	Typ. Val.	Description

CONSTANTS LOADED BY TRANSPORT SELECTION:

D8	lsd	x0	Four audio channels at a time may be assigned to the Track Select (Master transport) connector:
		x0	Analog channels 1 thru 4
		x1	Digital channels 1 thru 4 [SONY EMULATION ONLY]
		x2	Digital channels 5 thru 8 [SONY EMULATION ONLY]
		x8 thru xA	Same as x0 thru x2, but upper four outputs of Track Select connector are converted to individual track EE commands.
D8	msd	0x	Record Strobe and Ready Arming configurations:
		1x	Record strobe required when adding new record tracks while recording
		2x	Rehearse strobe required when adding new rehearse tracks while rehearsing
		4x	Track record ready signals are used also as EE (input) commands [AMPEX EMULATION ONLY]
		8x	Transport requires advanced rehearse arming
		Note:	Bits may be combined e.g. 5x = Record strobe required plus use record ready's as EE commands.
D9	lsd	x0	Jog/shuttle quality flags:
		x1	Transport's jog/shuttle performance is not adequate for cueing (used in conjunction with constant E5 msd 4x bit).
		x2	Lack of lifter defeat option makes low speed jog/shuttle inadequate for audition.
D9	msd	2x	Play-to-cue-point override. (Unit = 1 sec.) This constant is almost identical to constant 07 msd, and is used only during emulation mode cueing.
		0x to 7x	Play-to-cue override. The actual value used will be the larger of this and constant 07 msd.

CONSTANTS LOADED BY EDITOR / VTR SELECTION:

E0	both	??	VTR device type (low byte).
E1	both	??	VTR device type (high byte) Note: During Sony emulation, if the Zeta is set up for 25 frame code time code, this byte will be modified to reflect PAL operation before being transmitted back to the Editor.

E2	lsd	x2	Number of analog tracks supported by target transport.
E2	msd	0x	Number of "digital" tracks supported by target transport. If the Editor is capable of specifying digital channels, then it may be useful to assign all of the transport's tracks as "digital". When operating such an Editor in a mode where the Zeta's transport "chases" the Record VTR, analog channel inserts will then select VTR tracks, while digital channel inserts will select tracks on the Zeta's transport. [SONY EMULATION ONLY]
E3	lsd	x6	Edit command delay (unit=frames): Editing commands are typically operated with a fixed delay between receipt of the command and its execution. Commands affected are . . Sony: Select EE On, EDIT Off, EDIT On, EDIT PRESET Ampex: Entry, Exit.
E3	msd	0x	Field 2 and other bits: 1x Execute all edit commands on Field 2 [SONY EMULATION ONLY]. 4x Automatically issue a "Standby (ready) On" command whenever the Zeta's transport becomes enabled. 8x "Zero delay timeline run correction" [AMPEX EMULATION ONLY].
E4	both	0B	Capstan scale factor. When the Editor requests varispeed play, this constant will cause the deviation from playspeed to be either decreased or increased. msd: 0=decrease (damp) 1=increase lsd: amount of adjustment (unit=deviation/16) For example, a setting of 01 will result in a 1/16th reduction in the varispeed deviation from playspeed. SONY EMULATION ONLY: Varispeed play is typically used during the synchronization process. Both the value of Constant E4 and the setting of the "Address Resolve" bit in Constant E5 will be critical in achieving synchronization within a reasonable preroll time.

E5 lsd x0 Emulation controls (may be combined to form hexadecimal values 0-F):

x1 "Address Resolve" bit:
At the conclusion of the synchronization process, when the Editor switches the Zeta from varispeed play to standard play, setting this bit guarantees that the transport will resolve at exactly the same frame offset that existed when the standard play command was given.
If this bit is not set, then normal resolving will take place, with the possiblity that the transport's inertia will carry it to another frame. In this case, the Editor must account for this drift. Generally, an Editor which is trying to be more "intelligent" during synchronization may run better if this bit is not set.

x2 Should the Editor request a re-cue after the transport has already cued, then the re-cueing action will actually take place. Normally, the Zeta will not bother to re-cue an already cued transport, and this may be confusing to some Editors.

x4 Inhibit smart start allowances during cueing [AMPEX EMULATION ONLY].

x8 Target Cue position bit:
At the end of a Target Cue operation, the Zeta will normally inform the Editor that it has parked the transport precisely at the requested cue point. Due the the difficulty of cueing an audio transport to an exact time code frame, this may not entirely be the case! If the transport's actual position were to be reported, the danger is that the Editor may then request an unnecessary and probably futile re-cue.
Entering an 8 in this constant will force the Zeta to "tell the truth". Assuming that the Editor does not subsequently re-cue the transport, then this setting may in fact lead to shorter synchronization times.

E5 msd 0x Emulation controls (may be combined to form hexadecimal values 0-F):

1x Force lifter defeat at all times (some Editors require a separate, continuous time code feed).

2x Inhibit Play-to-cue action during emulation. Not recommended while editing, but may be useful when simple machine control only is required. Will not inhibit the effect of constant 7 msd however.

4x Only "manual quality" jog/shuttle is required. If not set, and transport constant D9 lsd has its "x1" bit set, then jog/shuttle will be replaced by fast forward or rewind commands. This ensures that a poor quality jog/shuttle will not be invoked when an Editor uses these modes for cueing purposes.

8x Allow "direct" fast forward and rewind commands from the Editor to operate normally, even if this will cause an attached VCR to unlace. If not set, and if constant 47 specifies that unlacing will take place, then "direct" ff/rew commands will be converted to maximum speed shuttle.
[SONY EMULATION ONLY, Ampex has no "direct" ff/rew commands]

E6 lsd x1

Global conversions to varispeed play (may be combined to form hexadecimal values 0-7):
VTR protocols typically support three different variable speed commands - Variable-play, Jog and Shuttle. An audio transport, on the other hand, may reliably implement only varispeed play, and that only within a narrow range.

- x1 — Convert all forward direction VTR Variable-play commands to the transport's varispeed play mode (this bit is normally set, as Variable-play is most often used during synchronization).
- x2 — Convert all forward direction VTR Jog commands to the transport's varispeed play mode.
- x4 — Convert all forward direction VTR Shuttle commands to the transport's varispeed play mode.

E6 msd 0x

Specific VTR Variable-play controls (may be combined to form hexadecimal values 0-F):
<u>These bit definitions apply also to Constants E7 lsd (Jog) and E7 msd (Shuttle).</u>
The tape speed requested by the editor is classified as being in a Low, Mid or High speed range (see constants E8 thru F1). Unless overridden globally in constant E6 lsd, or "specifically" in this constant, the emulator section will always attempt to use whatever jog/shuttle functionality that the transport is capable of.
Many Editors use VTR variable speed commands (play/jog/shuttle) during the cueing process. The purpose of these controls (constants E6 msd, E7 lsd and msd), together with the threshold constants E8 thru F1, is primarily to intercept speed changes during the cue, and synthesize a play-to-cue-point at the end of the cue. For example, an Editor will typically request a fairly "Low" speed when approaching its cue point, and if variable play is enabled for the appropriate Low speed range (forward direction only), then a play-to-cue will be achieved.
These controls may secondarily be used to control manual jog/shuttle action, especially in the High speed range, where the choice is to hold the transport at its maximum jog/shuttle speed, or allow it to switch to full fast forward or rewind.

- 1x — Force variable play in the Low speed range.
- 2x — Allow "switch-back" from Mid and High speeds to variable play in the Low speed range (requires 1x also). If not set, then once the transport has switched to a higher speed mode (ff/rew/shuttle etc.), it will never switch back to variable play, even if the requested speed enters the Low range. Typically, this bit will be set if optimizing for cueing operations, and not set if optimizing for manual shuttling, where jumping in and out of play mode would be undesirable.
- 4x — Enable transport fast forward and/or rewind action when requested speed is in the High range. Otherwise, use jog/shuttle mode.
- 8x — Further allow fast forward and rewind commands in the High range, even if this will cause an attached VCR to unlace (requires 4x to be set). Setting the 4x bit and NOT setting this bit says "use ff/rew, but not if the deck will unlace".

E7	lsd	x0	Specific VTR Jog controls (may be combined to form hexadecimal values 0-F): Same bit functions as constant E6 msd.
E7	msd	0x	Specific VTR Shuttle controls (may be combined to form hexadecimal values 0-F): Same bit functions as constant E6 msd.
E8	both	05	Requested-tape-speed threshold, "Low" to "Mid" speed upward transition (hexadecimal, 03=playspeed, 05 represents 1.67 x play).
E9	both	03	Requested-tape-speed threshold, "Mid" to "Low" speed downward transition (hexadecimal, 03=playspeed).
F0	both	1E	Requested-tape-speed threshold, "Mid" to "High" speed upward transition (hexadecimal, 03=playspeed, 1E represents 10 x play).
F1	both	12	Requested-tape-speed threshold, "High" to "Mid" speed downward transition (hexadecimal, 03=playspeed, 12 represents 6 x play).
F2	lsd	x3	Delay between receipt of a ROLL command and actual execution. 0 thru 7 Delay in frames. 8 thru F Negative delay (advance) of -8 thru -1 frames. [AMPEX EMULATION ONLY]
F2	msd	0x	Delay between receipt of a TIMELINE RUN and actual execution. 0 thru 7 Delay in frames. 8 thru F Negative delay (advance) of -8 thru -1 frames. [AMPEX EMULATION ONLY]
F3	lsd	x0	Delay between receipt of a deferred edit command (Entry/Exit) and actual execution. (Rarely used.) 0 thru F Delay in frames. [AMPEX EMULATION ONLY]
F3	msd	0x	Various Ampex bits [AMPEX EMULATION ONLY]: 1x Ignore parity errors on serial receive line. 4x Inhibit drop frame compensation of serial input data. 8x Return only limited VPR-3 record/rehearse mode status.

F4	both	00	EDIT CHANNEL MASK.

When the slave transport is chasing a Record VTR, then it will normally punch in and out of record at the same time as the VTR. However, it will often be desirable to limit the transport's record activity such that, for example, it will only go into record when the edit calls for audio channels. This Constant specifies the channel or combination of channels which must be enabled before recording will take place:

		00	No channel restrictions for record (or rehearse)
lsd	x0	x1	Record/rehearse if Audio 1 enabled
		x2	Record/rehearse if Audio 2 enabled
		x4	Record/rehearse if Audio 3 enabled
		x8	Record/rehearse if Audio 4 enabled
msd	0x	1x	Record/rehearse if Video enabled
		2x	Record/rehearse if Digital Audio 1 or 2 enabled
		4x	Record/rehearse if Digital Audio 3 or 4 enabled
		8x	Record/rehearse if Digital Audio 5,6,7 or 8 enabled

Notes:

1. Bits may be combined. For example, a value of "0F" specifies "any analog audio channel".
2. Digital audio channels may apply during Sony emulation only, and then only if supported by the editor in use.
3. Use of the EDIT CHANNEL MASK provides an alternative to the necessity of wiring the slave transport to the Zeta's track select connector. In most cases, the operator will manually select which channels are to be recorded on the slave deck, and the EDIT CHANNEL MASK will determine when those tracks are to be recorded.
4. Correct operation will be maintained during split edits. For example, if the EDIT CHANNEL MASK specifies audio channels only, and if a split video/audio edit calls for audio to enter record a certain time after the beginning of the edit, then the slave deck will not begin recording at all until the audio entry time is reached.

Appendix F

ADAMS-SMITH ZETA THREE MIDI SECTION CONSTANTS

Software Rev 3.60 (standard) and 3E60 (emulator)
Document revised Oct 29, 1990

Constant Number		Typ. Val.	Description
01	lsd	x6	Lock errors required before re-syncronization via cueing routines (Constant 34 defines 'lock error' size)
01	msd	0x	[not used]
05	lsd	x2	Limit for error contribution to lock speed change
05	msd	4x	Limit for lock speed adjust steps (x 3% center freq)
06	lsd	x3	Narrow lock window (unit = .001 frame)
06	msd	1x	Wide lock window (unit = .05 frame)
12	both	0C	Number of frames of MIDI Time Code jam errors required before re-jam takes place. Also applies to Direct Time Lock.
13	both	08	Number of frames of identical time code required before MIDI Time Code section recognizes a "stopped" condition. Also applies to Direct Time Lock.
15	lsd	x0	MIDI Time Code User Bits control x1 — Transmit User Bits Full Message at the onset of each new MIDI Time Code stream (typically whenever the transport starts to move).
15	msd	1x	MIDI Time Code transmit control 0x — Transmit 1/4 frames messages continuously 1x — Transmit 1/4 frame messages only when new time data is available (i.e. when transport moving) 2x — Transmit 1/4 frame messages when transport is in Play mode, and Full Messages (Time Code) at all other times 3x — Same as 2x, but cease all transmissions when no new data is available (transport stopped).
16	both	01	Minimum spacing between transmissions of MIDI Time Code Full Messages (unit = 0-99 1/2 seconds).

34	both	29	Maximum allowable lock error (unit = frames) before possible re-cueing (see also Constant 01 lsd)
37	lsd	x0	Smart-start defeat flags x0 Normal x2 Use fixed Smart-start offset
37	msd	0x	[not used]
38	lsd	x0	AUTO lock switchover delay (unit = secs)
38	msd	0x	Special AUTO override modes (bits may be combined) . . 8x Override the LOCK MODE menu setting and force AUTO mode. 1x Inhibit resolve. When in system resolve mode, the Zeta will normally try to resolve the tempo map whenever it is placed into play mode by itself (master and slave disabled).
39	lsd	x3	Failure Sense MIDI Input timeout (unit=frames). If activity at the MIDI input lapses for longer than this time period, then the MIDI sending device is judged to have failed. Failure Sense mode for MIDI IN must first be invoked.
39	msd	0x	not used
45	both	00	Time that the Zeta must WAIT after transmitting a MIDI Song Pointer command and before transmitting a Continue command. (Unit = 0-99 1/2 seconds)
46	both	00	SPACING time that the Zeta must wait after transmitting a MIDI Song Pointer command and before transmitting another Song Pointer command (unit = 0-99 1/2 seconds).
47	both	7F	Destination MIDI device ident for MIDI File Dump i.e. tempo map save to Sysex. Value 7F is an "allcall" address, meaning that the map file is transmitted to any and all MIDI devices that are attached.
48	both	7F	MIDI device ident to which a "File Dump Request" message will be sent when a tempo map load is initiated. File Dump Requests must be enabed in Constant 50.

49	lsd	x0	MIDI File Drop Frame adjustment:	
			x0	No drop frame time correction.
			x1	If a received MIDI File specifies that timings are expressed frames and subframes (normally they will be in "ticks per quarter note"), and if the File also specifies the 30 Drop Frame rate, then the Zeta will treat all frame counts as measurements of absolute time, making appropriate adjustments for the slightly slower frame rate.
49	msd	0x	[not used]	
50	lsd	x0	Automatic MIDI File Save flag:	
			x0	No automatic save
			x1	Transmit Tempo Map contents in MIDI File format through the MIDI Out port whenever requested to do so by a "File Dump Request" message which is received at the MIDI In port (no operator intervention required).
50	msd	0x	Automatic MIDI File Load flags:	
			0x	No automatic load actions
			1x	Accept any MIDI File which is received at MIDI In port without the operator first having to initiate such action through the ".4 LOAD MFILE SYSEX" menu.
			2x	Transmit a "File Dump Request" message whenever the operator requests a MIDI File Load. This may be used to prompt another device to automatically download a MIDI File to the Zeta.
			3x	Combination of 1x and 2x.
51	lsd	x0	[not used]	
51	msd	0x	Special Song Pointer / Sync modes :	
			1x	Issue a single Song Pointer only, immediately prior to Continue command.
52	lsd	x1	x0	Frames-per-Beat display fully decimal.
			x1	Frames-per-Beat increments by 1/8ths (.125) when the cursor is in the ".1" column.
52	msd	0x	0x	Display all messages during Map editing.
			1x	Supress confirmation messages during Map edits.
53	both	02	Number of bars that the Remote Control beeper is to count in before the start of a Song when enabled (0-99).	
54	both	10	MIDI Metronome channel number.	
55	both	37	MIDI Metronome note number.	

Appendix G

ADAMS-SMITH ZETA THREE GENERATOR CONSTANTS

Software Rev 3.60 (standard) and 3E60 (emulator)
Document revised Oct 29, 1990

Constant Number		Typ. Val.	Description
03	lsd	x0	User Bits Copy type (User bits may be copied from Time Codes, User Bits containing time code, or from User Bits containing static numbers or other data. When copying from any type of time code, it is important that the various status bits are masked correctly, and that the time code number is incremented by one frame to allow for decoding delays.) x0 Automatic detection of Time Code / non-Time-Code sources (see also Constant 12) x2 Force non-Time-Code mode x3 Force Time Code mode
03	msd	0x	Time Code Copy type 0x Jam (i.e. Generator code will continue to run when source code stops or disappears). 1x Transfer (i.e. Generator code will stop when the source code is detected to have stopped). Note: THIS CONSTANT DUPLICATES MENU ITEM G02
05	lsd	x0	Output options while generator stopped: x0 Continuously repeat current frame number x1 Output fixed 2400Hz
05	msd	0x	not used
10	lsd	x0	User Bit Group Flags to be inserted in the output Time Code
10	msd	1x	Remote Beeper enable 0x Beeper off 1x Beeper will sound at Generator "REJAM"
11	lsd	x0	User Bits fast mode copy x0 Copy only Play speed, non-tach-updated source code x1 Attempt to copy any source code, regardless of code velocity or tach update.

11	msd	1x	Time Code fast mode copy 0x Copy only Play speed, non-tach-updated source code 1x Attempt to copy any source code, regardless of code velocity or tach update. 2x Same as 1x, but if source is tach updating at play speed the generator will run without copying, regardless of the Jam/Transfer setting. (Good for covering time code "holes".)
12	both	0C	Time Code error hysteresis (hexadecimal). i.e. the number of contiguous frames of copying errors that must be encountered before a Re-jam will occur.
13	both	03	Time Code "stopped" hysteresis (hexadecimal). i.e. the number of contiguous frames for which no motion is detected in the source time code before the Generator will stop while copying in the "Transfer" mode.
12	both	03	User Bits "counting" hysteresis (hexadecimal). Used in "automatic" copy mode (Constant 03 msd = 0) for detection of time-code / non-time-code sources. i.e. the number of contiguous frames of correctly incrementing source numbers that must be detected before being accepted as Time Code.

Appendix H

ADAMS-SMITH ZETA THREE EVENTS CONSTANTS

Software Rev 3.60 (standard) and 3E60 (emulator)
Document revised Oct 29, 1990

Constant Number		Typ. Val.	Description
01	both	00	Event #1 advance (unit = 0-99 1/4 frames)
02	both	00	Event #2 advance
03	both	00	Event #3 advance
04	both	00	Event #4 advance
05	both	00	Event #5 advance
06	both	00	Event #6 advance
07	both	00	Event #7 advance
08	both	00	Event #8 advance
09	both	00	Event #9 advance
10	both	00	Event #10 advance
11	both	00	Event #1 x100 advance (unit = 0-99 hundreds of 1/4 frames)
12	both	00	Event #2 x100 advance
13	both	00	Event #3 x100 advance
14	both	00	Event #4 x100 advance
15	both	00	Event #5 x100 advance
16	both	00	Event #6 x100 advance
17	both	00	Event #7 x100 advance
18	both	00	Event #8 x100 advance
19	both	00	Event #9 x100 advance
20	both	00	Event #10 x100 advance
21	lsd	x0	Event #1 pairing flags
			x1 Next event = 'punch out' for this one
21	msd	0x	Event #1 controls
			1x Automatic disarm after firing
			8x Invert output state while disarmed
22	both	00	Event #2 pairing and controls
23	both	00	Event #3 pairing and controls
24	both	00	Event #4 pairing and controls
25	both	00	Event #5 pairing and controls
26	both	00	Event #6 pairing and controls
27	both	00	Event #7 pairing and controls
28	both	00	Event #8 pairing and controls
29	both	00	Event #9 pairing and controls
30	both	00	Event #10 pairing and controls

35	lsd	x6	Momentary event output duration (unit = 8.3 msec)
35	msd	3x	Number of talent cueing beeps to be inserted at the request of the Zeta system (1-7). Events will be assigned in reverse order starting at event #10.
36	both	25	Number of frames separating talent cueing beeps.

NOTE:
Constants 51 thru 70 contain hex values which allow each event, momentary or continuous, to be mapped to the Aux Out tip/ring connector, as well as to the Master transport connector if desired. (The latter requires that the Master menu item "OUTPUT=AUX3-10" also be selected.)
The connector pins are defined as follows . .

Map Value	Pin Name	Pin Number	Common Pin Number
01	Aux 3 (Stop)	3	19
02	Aux 4 (FF)	4	19
04	Aux 5 (Rew)	5	19
08	Aux 6 (Play)	6	19
10	Aux 7 (GP1)	7	20
20	Aux 8 (GP2)	8	21
40	Aux 9 (GP3)	9	22
80	Aux 10 (Rec)	10	19
F1	Aux 1 (Tip)		
F2	Aux 2 (Ring)		
F4	Slave capstan relay		

51	both	F1	Event #1 momentary output
52	both	F1	Event #2 momentary output
53	both	F1	Event #3 momentary output
54	both	F1	Event #4 momentary output
55	both	F1	Event #5 momentary output
56	both	F2	Event #6 momentary output
57	both	F2	Event #7 momentary output
58	both	F2	Event #8 momentary output
59	both	F2	Event #9 momentary output
60	both	F2	Event #10 momentary output

61	both	00	Event #1 continuous output
62	both	00	Event #2 continuous output
63	both	00	Event #3 continuous output
64	both	00	Event #4 continuous output
65	both	00	Event #5 continuous output
66	both	00	Event #6 continuous output
67	both	00	Event #7 continuous output
68	both	00	Event #8 continuous output
69	both	00	Event #9 continuous output
70	both	00	Event #10 continuous output
82	both	00	Aux 3-10 (Master transport connector) invert mask - pins selected here will become active high, normally low
83	lsd	x0	Aux 1,2 (and capstan) invert mask
		x1	Invert Aux 1 (Tip)
		x2	Invert Aux 2 (Ring)
		x4	Invert Capstan relay sense
83	msd	0x	[not used]

Appendix I

ADAMS-SMITH ZETA THREE ERROR MESSAGES AND ERROR CODES

Software Rev 3.60 (standard) and 3E60 (emulator)
Document released Oct 29, 1990

Dec	Hex	Displayed Message
0	00	not used
1	01	G_TC ** REJAM **
2	02	G_UB ** REJAM **
3	03	*01* NO MORE MENUS
4	04	*02* AT LOWEST MENU
5	05	*03* NO ENABLES
6	06	*04* NO GOTO POINT
7	07	*05* NO FUNCTION
8	08	*06* NOT LOCKED
9	09	-- CONSTS LOADED ---
10	0A	*07* USER AREA EMPTY
11	0B	--- CONSTS SAVED ---
12	0C	S_SLEW NEW OFS 00.00
13	0D	-- COPY HALTED --
14	0E	M_TC ** XX FRAME **
15	0F	S_TC ** XX FRAME **
16	10	-- REG'S CLEARED --
17	11	REMOTE TEST - WAIT
18	12	REMOTE TEST - RUN
19	13	REMOTE XX FAILURES
20	14	*08* NO LOCAL CTRL
21	15	*09* NO MAST OUTPUTS
22	16	*10* COMMAND TIMEOUT
23	17	--- VERSION XXXX ---
24	18	*11* SONG PTR OVFLW
25	19	*12* MAP RUNNING
26	1A	*13* MAP CHASING
27	1B	*14* FATAL MAP ERROR
28	1C	------ MARK B ------
29	1D	---- CHANGES OK ----
30	1E	-- CANCEL CHANGES --
31	1F	- CANCEL OVERRIDES -
32	20	-- BARS INSERTED ---
33	21	--- BARS DELETED ---
34	22	--- BARS COPIED ----
35	23	*15* MARKER OVER END
36	24	*16* MAP OVERFLOW
37	25	*17* CLICK/SIG CLASH

Dec	Hex	Displayed Message
38	26	*18* TOO MANY BARS
39	27	*19* BAR OVERFLOW
40	28	*20* NO CHANGES
41	29	*21* CLICK VARIATION
42	2A	*22* SIG VARIATION
43	2B	*23* BEAT OVERFLOW
44	2C	*24* MULTIPLE SONGS
45	2D	*25* MARKS COINCIDE
46	2E	*26* MARKER MISSING
47	2F	*27* NOT AT BARLINE
48	30	*28* SAVE/LOAD
49	31	not used
50	32	not used
51	33	not used
52	34	*29* GEN NOT READY
53	35	not used
54	36	---- SAVING MAP ----
55	37	--- MAP SAVED OK ---
56	38	----- WAITING ------
57	39	--- LOADING MAP ----
58	3A	-- MAP LOADED OK ---
59	3B	*30* CRC ERROR
60	3C	*31* LOAD TIMEOUT
61	3D	*32* TIMING ERROR
62	3E	not used
63	3F	FRAME RATE MISMATCH
64	40	- MAP SAVE ABORTED -
65	41	- MAP LOAD ABORTED -
66	42	- RECALL OVERRIDES -
67	43	*33* FILE BUSY
68	44	not used
69	45	not used
70	46	not used
71	47	*34* BAD BAR COUNT
72	48	*35* BAD FILE DATA
73	49	*36* DATA OVERRUN
74	4A	*37* MAP DESTROYED
75	4B	*** SYSTEM RESET ***
76	4C	------ MARK A ------
77	4D	----- RESTART ------
78	4E	*38* MAP LEARN ERROR
79	4F	*39* NOT IN SONG
80	50	*40* NO REFERENCE
81	51	*41* TEMPO TOO FAST
82	52	*42* TEMPO TOO SLOW
83	53	*43* LEARNING TEMPO
84	54	-- LEARN ABORTED ---
85	55	*44* BEAT CLIPPING
86	56	*45* MIDI DATA LOSS
87	57	*46* MERGE DATA LOSS
88	58	*47* MIDI DATA ERROR
89	59	*48* MIDI DATA OVFLW

Dec	Hex	Displayed Message
90	5A	*49* NOT ENABLED
91	5B	*50* NO MASTER SOLO
92	5C	*51* COMP RX ERR XX
93	5D	*52* CTRL RX ERR XX
94	5E	*53* TAPE I/0 BUSY
95	5F	*54* FILES BUSY
96	60	*55* FILE ERROR
97	61	*56* FILE TYPE ERROR
98	62	*57* NO AUTO FILES
99	63	-- FILE TRANSMIT ---
100	64	---FILE TXMIT OK ---
101	65	---- FILE WAIT -----
102	66	- TXMIT CANCELLED --
103	67	- FILE TXMIT ABORT -
104	68	*58* WAIT TIMEOUT
105	69	*59* FILE NAK ERROR
106	6A	--- FILE RECEIVE----
107	6B	-- FILE RECVD OK ---
108	6C	-- RECV CANCELLED --
109	6D	- FILE RECV ABORT --
110	6E	-- PREMATURE END ---
111	6F	*60* FILE TIMEOUT
112	70	*61* BANK RESTART
113	71	*62* ZETA TC LINK ON
114	72	-- MAP TRUNCATION --
115	73	**SERIAL TRANSLATE**
116	74	*63* TIME SIG ERROR
117	75	*64* QUEUE OVERFLOW
118	76	--- ALL DISARMED ---
119	77	*65* NO LOCATE POINT
120	78	-- SERIAL RESTART --
121	79	*66* EDIT DELAY OVFL
122	7A	D** MIDI TC *REJAM*