13 COMPUTER PORT [Adams-Smith Protocols]

[Software Revision 3.60 (standard) and 3E60 (emulation)]
[Document revised Oct 29, 1990]

13.1 INTRODUCTION

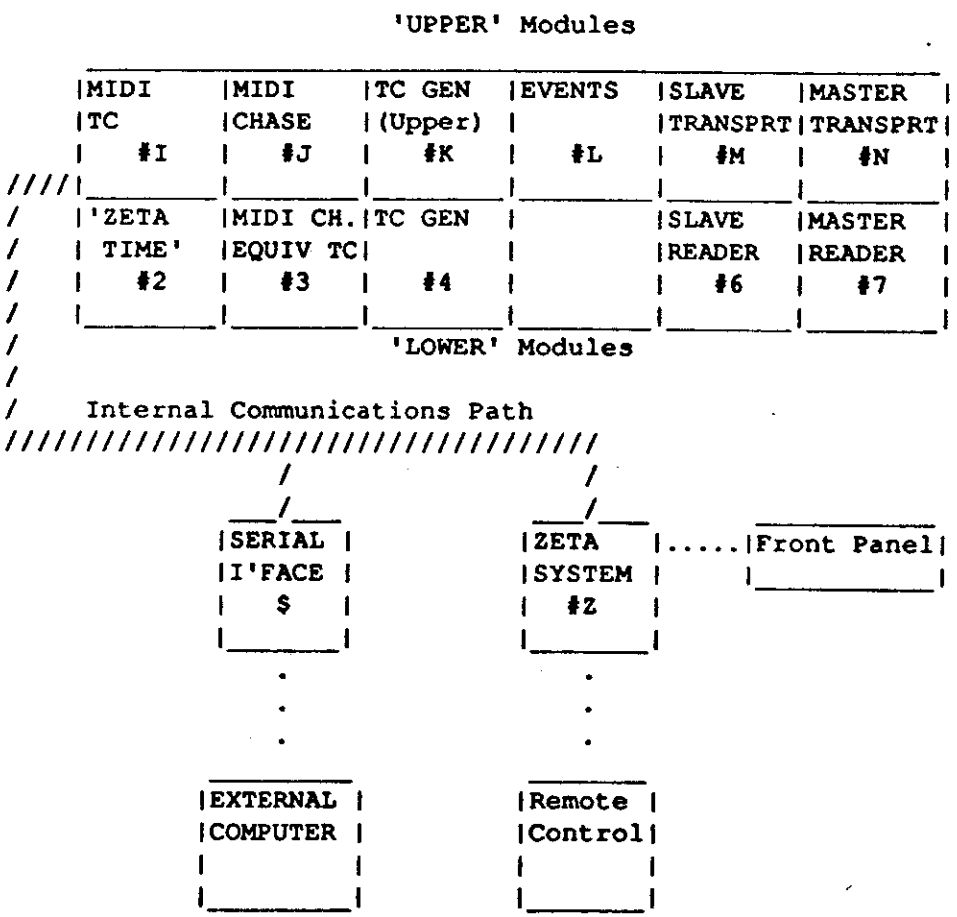
Commands sent to the Zeta III consist entirely of ASCII characters with no special formatting.

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 preceded 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 address '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 ('attached 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
'j'	EVENTS TRIGGER SOURCE TIME ('j' = 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 compatibility 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.

13.1.9 ZETA MENUS FOR SERIAL OPERATION

The COMPUTER PORT protocols are enabled and controlled by Zeta system menus Z08, Z09 and Z10.

Menu Z08 selects Adams-Smith Protocols over either an RS-232, RS-422, or MIDI connection:

```
Z08 COMPUTER PORT ->
                    .1 OFF
                    .2 ADAMSSMITH 232
                    .3 ADAMSSMITH 422
                    .4 ADAMSSMITH *MIDI*
```

Menu Z09 selects the serial baud rate for an RS-232 connection only. The RS-422 connection runs at a fixed baud rate of 38.4Khz.

```
Z09 RS-232 BAUD   ->
                    .1 150
                    .2 300
                    .3 600
                    .4 1200
                    .5 2400
                    .6 4800
                    .7 9600
                    .8 19200
                    .9 38400
```

RESPONSES (from Zeta to Controller):

F0 17 <chan> 02 <Immediate chars . . > <Responses . . > F7

17h	Adams-Smith sysex id
<chan>	**Source** Zeta system address
02	Response sub-ID#1
<Immediate chars>	X-ON, X-OFF etc (if any)
<Responses>	Adams-Smith responses (if any)

NOTES

1. *<Immediate Characters>* in a Command message consist of X-ON and X-OFF, as defined in section 13.2.1, and should be placed in front of any other commands.

2. *<Immediate Characters>* in a Response will always be placed at the beginning of the message. These consist of the Serial Control characters X-ON and X-OFF as well as the Special characters NAK and <FRAME> as defined in sections 13.3.1 and 13.3.2 respectively. Serial Metronome characters (13.3.2) will also be placed here.

3. The BREAK message will produce exactly the same results as a serial break. Execution is immediate.

4. The Zeta system address is set up using the "Z12 SYS ADDR=0/8282" menu. Only the first address digit (before the "/") is relevant.

13.1.11 MONITORING SERIAL RECEPTION

Activity on the serial receive line may be monitored by observing the Zeta system menu "Z08 COMPUTER PORT ->" (software version 3.60 or later). The presence of an asterisk in this menu indicates that incoming characters are being detected by the Zeta's serial routines.

"Z08*COMPUTER PORT ->"

Jump to Page 71

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.

[tn] @T

TRANSPORT OVERRIDES:
t=0 LIFTER DEFEAT:
n : 0 = Zeta controls LIFTER DEFEAT (default)
1 = Force LIFTER DEFEAT
2 = Force LIFTER UNDEFEAT
3 = Force SOFT LIFTER DEFEAT
(mute follows lifters; not available during lock)
t=1 MUTE:
n : 0 = Zeta controls MUTE (default)
1 = Force MUTED
2 = Force UNMUTED
t=2 PLAY TO CUE:
n : Overrides Constant 7 MSD, unless the latter contains a larger playup value.
[Range = 0-7 seconds]
t=3 WIDE LOCK WINDOW:
n : Overrides Constant 6 MSD.

t=4 STANDBY (READY) OFF/ON:
n : 0 = Output OFF
1 = Output ON

[vvmmtn] @T

t=F MASKED TRANSPORT OVERRIDES:
mm=bit mask for active bits in vv
n=0:
vv: bits 1-7:undefined
bit 0: Override Constant 38 msd bit 1, (resolve inhibit), forcing resolve mode

[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

Jump to Page 75

140	not used
141	not used
142	not used
143	not used
144	not used
145	not used
146	not used
147	not used
148	not used
149	not used
150	FOSTEX R8
151	STELLAVOX TD-9 PERFO (w/TC)
152	not used
153	STUDER D-820 (44.1KHz)
154	STUDER D-820 (48KHz)
155	JVC DS-DT900 (DAT)
156	PANASONIC AU-660
157	not used
158	not used
159	not used
160	not used
161	not used
162	not used
163	not used
164	SONY BVH-2000 (Master only)
165	TASCAM BR-20T

@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. 'o0001592399+'
'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
't'	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.

[t dd] G JOG/SHUTTLE

t=0 : JOG mode:
- dd : 00h = 0.033 x Playspeed
  - 01h = 0.05
  - 02h = 0.08
  - 03h = 0.12
  - 04h = 0.20
  - 05h = 0.32
  - 06h = 0.50
  - 07h = 1.00
  - 80h-87h = same speeds, reverse direction

t=1 : SHUTTLE mode:
- dd : 00h = 0.033 x Playspeed
  - 01h = 0.10
  - 02h = 0.20
  - 03h = 0.50
  - 04h = 1.00
  - 05h = 2.00
  - 06h = 5.00
  - 07h = maximum
  - 80h-87h = same speeds, reverse direction

t=3 : STILL
- dd : ignored

[fff t dd] G

JOG/SHUTTLE (Alternate Format)
t=4 : HI-RES JOG/SHUTTLE (auto lifters):
fff : 12-bit playspeed fraction (msd first)
dd : Hex integer playspeed multiplier (01h=play)
Range :00h-7Fh in forward direction
80h-FFh in reverse direction
Notes : 1. fff=dd=0 produces STILL mode
2. Speed specifications are identical in both forward and reverse modes. Direction is simply given by the 80h bit in dd.

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 (00-99, A0-A9, B0-B9 . . F0-F9)
vv = new value

[ccvvmm] X K

MASKED CONSTANT LOAD
cc = constant number (00-99, A0-A9, B0-B9 . . F0-F9)
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 or [xv] R	PULSE EVENT OUTPUT The first four variations of this command (v=0-3) simply cause record or rehearse punch in/out without affecting internal status (i.e. bits "tttt" in the STATUS response - see section 13.10.14) x : Ignored - need not be sent. v : 0 = Record in 1 = Record out 2 = Rehearse in 3 = Rehearse out The next four variations (v=4-7) have the same outward effect, but do affect internal record/rehearse status. The internal "edit" flag, described below, is always reset. x : Ignored - need not be sent. v : 4 = Record in 5 = Record out 6 = Rehearse in 7 = Rehearse out Variations v=8-B are the same as 4-7, but with the possibility of inhibiting the hardware outputs as well as manipulating the internal "edit" flag. This "edit" flag allows punch in/out sequences to occur without recording or rehearsing actually taking place. This emulates a mode found on many transports, where the deck may be in record (record button lit), but no recording takes place until record tracks are selected.

x : bit 8 : 1=inhibit hardware output
bit 4 : not used
bit 2 : 1=don't change "edit" flag (ignore bit 1)
bit 1 : 1=set "edit" flag; 0=reset "edit" flag (if bit 2 = 0)
v : 8 = Record in
9 = Record out
A = Rehearse in
B = Rehearse out

Variations v=C,D simply set or reset the internal "edit" flag. Internal record/rehearse status is not affected.
v : C = "Edit" flag on
D = "Edit" flag off

Finally, variation v=E allows record/rehearse arming outputs to be directly controlled, independent of actual arming status.
x : bit 8 : 1=Adjust record arming output via to bit 4
bit 4 : 0=reset record arming output; 1=set (if bit 8 = 1)
bit 2 : 1=Adjust rehearse arming output via to bit 1
bit 1 : 0=reset rehearse arming output; 1=set (bit 2 = 1)
v : E = Adjust arming outputs.

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
't'	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.

V RESOLVE
Causes the transport to begin playing and to resolve to the current system master reference. That reference will typically be provided via the system resolve mode assignment, but may also be derived from the current system master time code when the system is unresolved. Results may be unpredictable if no reference is available.

XV

"ADDRESS" RESOLVE
This command is identical to the 'V' command except when the transport is already in varispeed play mode. Under that condition, this routine will capture the frame offset relationship that exists between the slave and the master frame reference at the time that the 'XV' command is received, and maintain that offset while resolve lock is being achieved. When locked, it will switch back to a standard resolve mode.
(This mode is provided mainly for the ZETA-THREE^{em}, which allows a controlling editor to achieve synchronization by direct control of the transport's varispeed).

[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