

Options:_____

Date Received:_____

Software Revision:_____
(manual not accurate for prior to "C1")

AEC-BOX-25

INSTRUCTION MANUAL

ADRIENNE ELECTRONICS
CORPORATION

(VITC-to-LTC Translator with RS232/RS422 Interface)

The serial interface is programmed as follows (change as needed):

Protocol:_____ Op Mode:_____ Address:_____

Baud Rate:_____ # Data Bits:_____ Parity:_____
(protocol related)

1 2 3 4 5 6 7 8
Switches = _ _ _ _ _

Third Edition

January 1997

Copyright (C) 1997 by Adrienne Electronics Corporation
7225 Bermuda Road, Unit G * Las Vegas, NV 89119 * U.S.A.
tel:+1-702-896-1858 fax:+1-702-896-3034 www.adrielec.com
Printed in U.S.A. All rights reserved.

*** FCC NOTICE ***

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, could cause harmful interference to radio communications. Operation of this equipment in a residential area may cause harmful interference, in which case the user will be required to correct the interference at his or her own expense.

*** COPYRIGHT NOTICE ***

You are authorized to copy and make changes to the firmware for this product only as described in this manual. Any other copies or changes constitute an infringement of our copyright.

*** TRADEMARK NOTICES ***

Adrienne, AEC, AEC-BOX, PC-207M, PC-LTC, PC-VITC, and PC-VLTC are trademarks of Adrienne Electronics Corporation.

SMPTTE is a registered trademark of
The Society of Motion Picture and Television Engineers, Inc.

TABLE OF CONTENTS

Title Page.....	1
FCC, Copyright, and Trademark Notices.....	2
Table of Contents.....	3
Introduction.....	4
Getting Started Quickly.....	4
Specifications.....	5
Hardware Description.....	6
External Cabling Installation.....	7
AC Input Voltage Range Selection.....	8
DIP Switch Programming.....	10
VITC Reader Operations.....	12
LTC Generator Operations.....	14
Comparator and Relay Operations.....	16
EPROM Modifications.....	18
Binary Broadcast Mode.....	20
Binary Polled Mode.....	21
ASCII Broadcast Mode.....	30
ASCII Polled Mode.....	31
Break Character Detection.....	32
Serial Interface Standards.....	33
LED Operations.....	34
Troubleshooting Guide.....	35
Warranty Registration, Warranty Service, etc.....	37
ASCII Code Chart.....	38
Number System Conversion Table.....	39
Where to Order Copies of Standards.....	40
User Feedback Request.....	back
Ordering Guide.....	back
Schematics.....	back

INTRODUCTION

The AEC-BOX-25 is a play speed VITC-to-LTC translator with RS232 and RS422 interfaces. The LTC output is synchronized to the video input, and will work even if the video/VITC input disappears or is otherwise of poor quality. Through the serial interface, the LTC generator can be monitored and controlled using a variety of personal computers and other devices. The AEC-BOX-25 works equally well with SMPTE (30fps) and EBU (25fps) time codes, in both the forward and reverse tape directions.

Longitudinal Time Code (LTC) is a specialized audio signal which contains digital time-of-day and frame count information relating to an accompanying video signal. As an audio signal, it can be recorded and played back by video and audio tape machines. LTC is used for audio/video editing, tape logging, and automation.

Vertical Interval Time Code (VITC) is typically present on two video lines in each vertical interval of a video signal. It contains the same information that LTC signals have. VITC cannot be read at high tape speeds (like LTC can), but it can be read at very slow (and even still) speeds, where LTC usually drops out.

Both time code standards have been around for many years, and are sponsored by both the SMPTE (for NTSC) and the EBU (for PAL).

Related products include our PC-LTC, PC-VITC, and PC-VLTC families of LTC and VITC readers and generators for IBM PC personal computers, along with other members of the AEC-BOX family of standalone VITC, LTC, video, RS232, and RS422 interface boxes. Call us if you would like further information.

GETTING STARTED QUICKLY

If you want to use your AEC-BOX-25 right away, without reading the whole manual, just do the following:

- 1) Plug the AC power cord into a suitable voltage AC outlet.
- 2) Connect the video/VITC input signal to the "VITC IN" BNC's.
- 3) Connect the "LTC OUT" RCA jack to wherever it is going.
- 4) (optional) Use a serial data cable to connect the 9-pin "D" connector (on the box) to your computer or other device.

The box should now be generating LTC, using EPROM default values or the VITC input data as a starting point. It will also be spewing out serial data every LTC frame, describing the present time, user, and embedded bits. The software protocol is described elsewhere in this manual. If any video or VITC errors are detected, or if either one is missing, the power LED will blink off, then back on. If something doesn't work, you will have to carefully read the "INSTALLATION" section of this manual.

AEC-BOX-25 SPECIFICATIONS

VITC/VIDEO INPUT:

Impedance	6kohms typical (Hi-Z)
DC on Input	+1.0V maximum
Input Level	0.8Vpp to 2.2Vpp (1.0Vpp nominal)
Looping Response	+0.1db maximum, 0-5MHz
Video Frequency	Must be within 2% of nominal frequency.

VITC READER:

Tape Speed Range (3)	-1x to STILL to +3x (w.r.t. play speed)
Tape Direction	Forward or Reverse
Bits Read	ALL time, user, and embedded bits.
Time Code Standards	SMPTE and EBU (automatically selected).

LTC OUTPUT (2):

Impedance	100ohms typical
Level	1Vpp typical
DC on Output	+50mV maximum
Freerun Error	+0.2% maximum
Phasing Error	+20us maximum w.r.t. NTSC/PAL specs

VITC-to-LTC TRANSLATOR:

Tape Direction	Forward or Reverse
Bits Read	ALL time, user, and embedded bits.
Time Code Standards	SMPTE and EBU (automatically selected).

MISCELLANEOUS:

Box Dimensions (4)	16cm wide x 5cm high x 21cm long
Box Weight	1.2kg
Power Consumption	7W
Temperature Range	0 to 50 degrees Centigrade
Relative Humidity	Up to 95%, noncondensing

Notes:

- (1) All specifications are subject to change without notice.
- (2) The LTC output frequency is always at "play" speed (+1x).
The time bits count may be STILL or changing at other rates.
- (3) Highly VTR dependent. Some are better, some are worse.
- (4) Allow at least 6cm in rear for cables and connectors.

AEC-BOX-25 HARDWARE DESCRIPTION

Throughout the following discussion you may want to refer to the AEC-BOX-25 schematics which are in the back of this manual. If your box has been customized in any way for your application, then the descriptions below may not be entirely accurate.

The power supply primary side comprises power transformer T1 and thermal "fuse" device F1. Of special note is the fact that this "fuse" does not burn out. If it trips, turn the power off for one minute to let it cool/reset, then turn the power back on.

The power supply secondary side starts with full wave rectifiers DB1 and DB2, plus large filter capacitors C72, C73, C82, and C83. The resulting unregulated DC supplies are then passed through voltage regulators U70(+5V), U72(+12V), and U73(-12V).

The video/VITC input signal first passes through 3-pole low pass filter R17,C44,L1,C45, then is buffered by U8A. Transistor Q5 is turned on by each sync tip, and thus generates horizontal sync pulses which are fed to U4 pin 1. Low pass filter RN10B,C40 only exceeds the threshold of comparator U6B during vertical reset pulses. The vertical sync output of U6B then goes to U4 pin 2.

OTA (operational transconductance amplifier) U11A is strobed on during the back porch of each video line, and forces the voltage on C42 to track the blanking level of the incoming video signal. OTA U11C is strobed on during each sync tip, and together with the blanking voltage from U11A generates a +40IRE voltage on C41. Comparator U6A then compares the VITC input pulses with the +40IRE level, and its digital logic output then goes to proprietary VITC reader chip U3.

The LTC output pulses are created directly by microcomputer U4. OTA U11C switches current into and out of C21, with the amplitude limited by zener diodes D8 and D9. This circuit forms the proper LTC waveform, with flat tops and sloped edges, per SMPTE/EBU specifications. Buffer amplifier U10A then drives RCA jack J2 through R18 and R19 to create the proper output level and impedance.

Microcomputer U4, together with address latch U20 and EPROM U1, form a completely self contained (but miniature) computer system. DIP switch SW1 allows easy modification of box operating modes, baud rates, and other features. "Watch dog" timer U5 resets the microcomputer chip, and thus the entire box, if the supply voltage drops too low or if the software crashes for some reason.

Serial data from UART U52 is translated to RS232 levels by U7, and is translated to RS422 levels by U14. U14 also translates received RS232 and RS422 data for use by UART U52. Nine pin "D" connector J3 contains the RS232 and RS422 data lines.

AEC-BOX-25 EXTERNAL CABLING INSTALLATION

VITC INPUT CONNECTION:

The two "VITC IN" BNC connectors are wired together, so you may loop your video/VITC signal through this box if desired. The high impedance input circuitry will load the video signal very little (see specifications). The nominal video input level is 1Vpp, but the input amplifiers will adjust themselves to other input levels, including the unterminated input condition.

LTC OUTPUT CONNECTION:

Simply connect the "LTC OUT" RCA jack (J2) to wherever you want it to go. Nominal signal level is 1Vpp.

SERIAL INPUT/OUTPUT CONNECTIONS:

If you ordered a serial data cable with your AEC-BOX-25, just plug it in to the 9-pin "D" connector (which has socket contacts) on the back of the unit. Otherwise, you may use the kit of mating connector parts to wire up to the box as follows:

Pin #	Function
1	
2	TX422-
3	RX422+
4	
5	TX232
6	
7	TX422+
8	RX422- and RX232
9	Chassis GND

Notes:

- 1) Tiny pin numbers are molded into the connector face. Be careful not to be "off by one".
- 2) For RS422, note that the pinout is that of an Eibus Tributary.
- 3) Limit RS232 cables to 30 meters maximum.
- 4) Limit RS422 cables to 1200 meters maximum.

AC INPUT VOLTAGE RANGE SELECTION

Normally all AEC-BOX's are shipped with the transformer primary wired for 100-130VAC. Your box will bear a special marking if it has been wired for 200-260VAC instead. For your own safety, PLEASE do not proceed unless the line cord has been unplugged! Just turning off a power switch somewhere is not sufficient!

*** WARNING ***

NEVER OPEN UP THE BOX unless the line cord has been unplugged from its AC power source! To do otherwise risks damage to your AEC-BOX, and could even KILL you! We cannot assume responsibility for such careless behavior.

Box Cover Removal:

First you must UNPLUG the AC power cord, remove all other cables, then remove the bottom cover as follows:

- 1) Use a small (#1) Phillips screwdriver to remove the two small black screws which are on each side of the box.
- 2) Slide off the front and back black plastic bezels.
- 3) Turn the box over, then lift off the BOTTOM cover.

Note that you are now exposing yourself to a severe (FATAL) shock hazard if the box is still plugged in to an AC power source!

Voltage Strap Modification for 200-260VAC:

In the area underneath power transformer T1, you will find four large holes in a row, with "115V" and "230V" markings adjacent. Using sharp nosed cutters, or some other appropriate tool, cut out at least 2mm of the narrow trace next to each of the "115V" markings. Then solder a short wire between the two holes closest to the "230V" marking, being careful not to poke the ends of the wire too far into the holes (could damage power transformer T1). Also make sure that the wire you added is flush with the bottom of the board, and will not even come close to touching to bottom of the box.

Voltage Strap Modification for 100-130VAC:

In the area underneath power transformer T1, you will find four large holes in a row, with "115V" and "230V" markings adjacent. Remove the wire between the two holes closest to the "230V" marking. Then solder a short wire between each pair of holes closest to the "115V" markings, being careful not to poke the ends of the wire too far into the holes (could damage power transformer T1). Also make sure that the two wires you added are flush with the bottom of the board, and will not even come close to touching to bottom of the box.

AC POWER INPUT VOLTAGE RANGE SELECTION
(continued)

Box Cover Replacement:

Basically, just follow the earlier instructions in reverse order (power to the box must be OFF):

- 1) Put the bottom cover back in place.
- 2) Slide a black plastic bezel onto each end of the unit. The box looks better if the two small molding marks are facing towards the bottom of the unit.
- 3) Reattach the bezels to the chassis with the four small black screws you removed earlier. Be careful not to strip the threads in the aluminum side extrusions!

Label The Line Cord:

Attach a small label to the plug end of the line cord, so that the next person to use this AEC-BOX will know what AC power input voltage range it expects to see.

Test Your Work:

BEFORE connecting any cables to the box, plug it in to the appropriate AC power source and make sure it works (no smoke).

INSTALLING YOUR OWN AC POWER LINE PLUG

If the plug on the end of the AC line cord is not suitable, you can cut it off and put on your own. Where possible, please wire the new plug as follows:

- 1) Green = Ground (Chassis)
- 2) Blue = Neutral
- 3) Brown = Hot

In no case should the green wire be connected to anything but ground! Use a continuity tester to verify that the ground lug on your new power cord is connected directly to the AEC-BOX chassis.

DIP SWITCH PROGRAMMING

Box Cover Removal:

First you must UNPLUG the AC power cord, remove all other cables, then remove the top cover as follows:

- 1) Use a small (#1) Phillips screwdriver to remove the two small black screws which are on each side of the box.
- 2) Slide off the front and back black plastic bezels.
- 3) Lift off the top cover.

Note that the bottom cover will fall off easily at this point, exposing you to a severe (FATAL) shock hazard if the box is still plugged in to an AC power source!

*** WARNING ***

NEVER OPEN UP THE BOX unless the line cord has been unplugged from its AC power source! To do otherwise risks damage to your AEC-BOX, and could even KILL you! We cannot assume responsibility for such careless behavior.

Changing DIP Switch (SW1) Settings:

Note that the switches are numbered 1 through 8. Also note the small "1" and "0" numbers down on the PCB next to both ends of SW1. To set a switch to be a "1", simply press down on the "1" (OPEN) end of that switch. Conversely, to set a switch to be a "0", simply press down on the "0" end of that switch. All done!

DIP Switch Functionality:

Switch	Function
1	Baud Code #1
2	Baud Code #2
3	Odd(1) or Even(0) Parity
4	Parity Enabled(1) or Disabled(0)
5	Seven(1) or Eight(0) Data Bits, & Protocol Control
6	Box Address Bit #2
7	Box Address Bit #1
8	Broadcast(1) or Polled(0) Operating Mode

The two "Baud Code" bits function as follows:

1	2	
1	1	=> 38400 baud
1	0	=> 19200 baud
0	1	=> 9600 baud
0	0	=> 1200 baud

Note that at 1200 baud, it takes longer than one LTC frame to transmit a message, so some LTC frame numbers will be skipped.

DIP SWITCH PROGRAMMING (continued)

Factory Default Setting:

Unless you requested otherwise, the factory default setting is BROADCAST mode, 9600 baud, BINARY message protocol (8 bits), and ODD parity, so SW1 will normally be 01110001 for switches 1-8, respectively.

Note that 1 stop bit is used for all serial operations, in all modes, and is not programmable to other values.

Message Protocol Notes:

It does not make sense to send 8-bit time and user bits data in binary form over a 7-bit data link, since the top bit will be lost. Thus you will find that DIP switch #5 controls the message protocol as well as the number of data bits being sent per serial character.

If you select 8 data bits, one of the BINARY serial interface modes will be used. See pages 20 and 21 for details.

If you select 7 data bits, one of the ASCII serial interface modes will be used. See pages 30 and 31 for details.

Box Cover Replacement:

Basically, just follow the earlier instructions in reverse order (power to the box must be OFF):

- 1) Put the top cover back in place.
- 2) Slide a black plastic bezel onto each end of the unit. The box looks better if the two small molding marks are facing towards the bottom of the unit.
- 3) Reattach the bezels to the chassis with the four small black screws you removed earlier. Be careful not to strip the threads in the aluminum side extrusions!

VITC READER OPERATIONS

The AEC-BOX's VITC reader uses a proprietary chip to read VITC from the video signal present at the "VITC IN" BNC's. It also filters out noise, and includes automatic level sensing circuits to compensate for input levels other than the nominal 1Vpp.

The VITC signal is usually present on two nonadjacent lines (for redundancy) in each vertical interval. Lines 10-20 are normally used with NTSC, and lines 6-22 are normally used with PAL. The box needs to be in the SMPTE/NTSC mode to read VITC from NTSC video signals, and needs to be in the EBU/PAL mode to read VITC from PAL video signals. Effective with software revision "D1" in October 1993, the box automatically detects whether NTSC/SMPTE or PAL/EBU type VITC is present. The box mode may also be changed via BINARY POLLED mode serial commands (see page 28).

The VITC reader needs to know which lines have VITC on them. When the power is first turned on (or if the box resets itself) the box reads default line numbers from the EPROM. You can change these EPROM default numbers by modifying the EPROM (see page 18). You can also change the line numbers via the POLLED BINARY mode (see page 28).

The factory default setting of "0" for both lines tells the VITC reader to read the first two lines which it thinks have VITC on them. If other signals are present in the vertical interval, such as teletext or a second set of VITC lines, it may get confused. Alternatively, you can specify which line numbers to use, as mentioned above. For NTSC, lines 10-25 are accepted. For PAL, lines 6-25 are accepted. The second line number should always be greater than or equal to the first line number. The box will force the numbers to be valid if you choose something which it can't understand.

If a nonstandard vertical sync pulse is detected, which is often the case with non-broadcast quality VTR's at anything other than play speed, the VITC reader is unable to count video lines. Bit 7 of time code status byte #1 will go high, and the VITC reader will read the first two lines it finds (if any) which have VITC on them, regardless of your EPROM and software settings.

It is also important to note that those VTR's which generate nonstandard vertical sync pulses often obliterate whatever is on NTSC lines 10-12. PAL machines undoubtedly do similar things. Thus if you are using these type VTR's, and if you have a choice, it is best to place VITC on some of the higher numbered lines.

(continued)

VITC READER OPERATIONS (continued)

It has been our experience that VITC is being used more and more with VHS and other "consumer" quality tape formats. These VTR's often do not have very good frequency and phase response. Since VITC requires accurate reproduction of the first 2MHz of the video signal, use of VITC on these machines is often marginal. Sometimes they will work pretty well at play speed, but not much else. Poor quality VITC waveforms simply cannot be read.

The VITC reader always uses the data read from the first VITC line, if it is OK. Otherwise it will use the data read from the second VITC line, if that is OK. If both lines are bad, the "VITC OK" time code status bit will be set to 0, and the LTC generator will create the next LTC data frame by itself.

Read errors are a fact of life, even with good tape machines. The CRC byte in each VITC word provides fairly good error detection capability, but does not eliminate 100% of bad data. We have included a "nonsequential VITC data filter" which ignores a VITC data word if it is more than 10 seconds off from the last valid VITC reader data word. However, if the very next VITC data word is also outside this 10 second window, it will be accepted so that the box will work properly when VITC data first appears, or when it jumps to a new set of values. This VITC data filter can be disabled by bit 4 of time code mode byte #2 (see page 27), which may be changed via EPROM modification or the binary polled mode.

Even better rejection of VITC read errors can be obtained by comparing the two lines in each vertical interval with each other, and/or field 1 vs. field 2 data. These two special filters can in some cases, however, make the VITC data virtually unreadable (rejects some good data). These filters are enabled by bits 6 and 5 of time code mode byte #2 (see page 27), which may be changed via EPROM modification or the binary polled mode.

If you desire to read VITC reader data instead of LTC generator data via the serial port, simply change bit 7 of time code mode byte #2 (see page 27) to a "1". Be sure not to accidentally change any other bits in this byte. Setting of the generator time and user bits is not affected by this bit.

LTC GENERATOR OPERATIONS

The LTC output is generated directly by the microcomputer chip. The box must be in the SMPTE/NTSC mode in order to sync to an NTSC input and generate SMPTE (30fps) LTC, and must be in the EBU/PAL mode in order to sync to a PAL input and generate EBU (25fps) LTC. Effective with software revision "D1" in October 1993, the box automatically detects whether an NTSC/SMPTE or PAL/EBU type video input is present. The box mode may also be changed via BINARY POLLED mode serial commands (see page 28).

The starting time, user, and embedded bits are taken from the EPROM (see page 18). You may change the generator time, user, and/or embedded bits, every frame if needed, by sending the appropriate serial commands (see pages 24-26). Since the generator data is updated shortly after the midpoint of field 2, you must load your new data into the generator before this time if you want it to appear in the very next LTC frame. Otherwise it will first appear in the following frame. Changing the generator data during the last half of field 2 will not hurt anything, but you won't know exactly when the new data will first be used.

For VITC-to-LTC translator operation, the first requirement is that the video input be present and valid. If present, the LTC generator will smoothly align itself with the video input, per the SMPTE/EBU standards, usually well within 1 second.

The second requirement is that the VITC data be present and readable. If the VITC data is OK, it will be automatically transferred to the LTC generator for use in the very next LTC frame. If desired, you can disable the jam (transfer) of time, user, and/or embedded bits from the VITC reader to the LTC generator. This is done by setting one or more of bits 5-7 to "0" in time code mode byte #1.

By disabling the jamming of user bits, you can perform the fairly common task of inserting your own user bits while transferring the time data. By modifying the EPROM, fixed user bits can automatically be inserted without computer control.

When transferring VITC reader data to the LTC generator, there is a one frame delay. In order to have the LTC output time bits agree exactly with the VITC input time bits, bit 3 of time code mode byte #1 can be set to a "1" (this is the default). The VITC time is then incremented once before being loaded into the LTC generator. If this bit is set to a "0", there will be the usual one frame delay. If you set bit 4 to a "1", the VITC time count will be incremented twice, so the LTC time output will actually be one frame ahead of the VITC! Note that these adjustments only apply to the time bits. The user and embedded bits will always have a 1 frame delay, since this is a causal system (we can't predict these bits ahead of time).

LTC GENERATOR OPERATIONS (continued)

If the video or VITC input disappears, or if any errors are detected, the LTC generator will "coast" as long as necessary, automatically incrementing the time count every frame. When (and if) the video input reappears, the LTC generator will smoothly realign itself with the video input within 1 second. When (and if) the VITC data reappears, the LTC generator will once again jam the VITC data into the LTC output data (if enabled) as usual.

Because the LTC generator output is always at "play speed", things get a little tricky when the VITC input is not at play speed. For instance, with helical scan VTR's, the tape can be STILL or moving backwards, yet the video signal coming into the AEC-BOX-25 still looks fairly normal. If time bit jams are enabled, and the box detects that VITC is valid, yet the VITC counts are not incrementing at the normal play speed rate, it will suspend its "auto-increment" and frame count adjustment functions. Instead, it will use the VITC time data which comes in, even if the VITC drops out for up to 1 second. If VITC disappears for more than 1 second, the box will revert to its normal "auto-increment" mode.

While you probably purchased the AEC-BOX-25 to translate VITC to LTC, you may want to use it as a freerunning LTC generator. If no video input is present, this will be the default. If there is a video input present, you must set bit 2 to "0" in time code mode byte #1, either via EPROM modifications (see page 18) or via serial commands (see page 27).

Time code mode byte #2 can be altered to control the LTC output in three ways. The LTC generator's time count will "freeze" (not change) if you set bit 1 to a "1", although the LTC output will still be enabled. The LTC output can be disabled at any time by setting bit 2 to a "1" (allow 200ms for the output amplifier to respond). Finally, the LTC output will be disabled automatically when the VITC count stops changing (or disappears for 1 second) if you set bit 3 to a "1". This latter feature is sometimes necessary to emulate what happens when a tape machine pauses or stops. When altering this time code mode byte, make sure you don't accidentally change any other bits.

The AEC-BOX-25 does not have the ability to detect color subcarrier phase, so the LTC generator cannot properly color frame or color lock the LTC output. Of course, if the VITC input is properly color framed, the LTC output will be also.

COMPARATOR AND RELAY OVERVIEW

The AEC-BOX-25 includes comparator software, which allows you to have the box transmit a short message whenever the LTC generator time bits (or user bits) reach a predetermined value. This frees up your system for other tasks. An optional relay may also be installed, which can be controlled at will, and can be turned on automatically whenever a comparator match is found.

COMPARATOR OPERATIONS

At the start (during bit 0) of every LTC frame, the LTC generator data is first compared, then masked, with values which you have set up ahead of time. If a match is detected (after masking), and if enabled, an ASCII "=" character (3Dh) will be transmitted, and/or relay #1 will be closed. Use of one of the polled modes eliminates transmission of other data which could obscure the "=" character.

If bit 7 of the comparator mode control byte is a 1, a "comparison" will be indicated both when the match first occurs, and again when the match disappears. It detects the "edges", and avoids generating lots of comparison indications when successive frames have the same data (after masking).

If you are using SMPTE drop frame counting, be sure not to set up a comparison time, such as 00:01:00:00, which doesn't exist.

The mask and data bytes may be changed at any time. See the software protocol section for details.

HOW TO USE COMPARATOR MASKS

There are many cases where you don't care what the value of one or more bits is, but you still want to use the comparator. To do this, just set the corresponding mask bit(s) to zero. For instance, if you want to generate a relay pulse every ten seconds, set the seconds mask byte to 0Fh, and set the minutes and hours mask bytes to 00h.

The comparator first does a logical XOR (comparison) of the current time (or user) data with the data you have supplied ahead of time. It then does a logical AND of the results with the mask data you have supplied. In positions where the mask data is 0, a "match" is always indicated. In positions where the mask data is 1, an exact match is required.

The default masks of 3Fh and 7Fh are used to filter out six special bits, called embedded bits, which are usually mixed in with the time bits. Always mask them out when not being used!

RELAY OPERATIONS

Relay #1 is installed if you ordered the "RELAY" option. This relay may be closed automatically by the comparator whenever a match is found. It can also be opened automatically after 50ms if desired. Relay #2 is available on a custom basis, and may only be controlled "manually". These relays open and close within 2ms of receiving the appropriate command. At all times, your software can control and sense the status of both relays. They are intended for switching control signals, not high currents or voltages.

HOW TO MODIFY COMPARATOR AND RELAY OPERATIONS

Comparator and relay operations are controlled by a single byte, which may be changed by EPROM modifications (see page 18) or by serial commands (see page 29). The structure of this byte is as follows:

Bit 7 = 1 if want to enable "edge" comparisons (see page 16)
Bit 6 = 1 if want to transmit "=" when comparator triggers
Bit 5 = turn optional relay #2 on(1) or off(0)
Bit 4 = turn optional relay #1 on(1) or off(0)
Bit 3 = 1 if want to turn off relay #1 automatically after 50ms
Bit 2 = 1 if want to turn on relay #1 when comparator triggers
Bit 1 = 1 if want to compare to user bits instead of time bits
Bit 0 = 1 if want to disable all comparator functions

As there is no way to read back the status of these bits, your software needs to keep track of what it has written to the box. The power-up default settings are comparator disabled, relays off (data = 01h).

HOW TO GET COMPARATOR AND RELAY STATUS

Comparator and relay status may be sensed by BINARY POLLED mode command 73h (see page 29). The status byte which is returned has the following structure:

Bit 7 = reserved
Bit 6 = reserved
Bit 5 = indicates relay #2 on(1) or off(0)
Bit 4 = indicates relay #1 on(1) or off(0)
Bit 3 = reserved
Bit 2 = reserved
Bit 1 = 1 if the comparator currently has a match
Bit 0 = 1 if the comparator has been triggered

Note that bit 0 gets cleared whenever you read this register.

AEC-BOX-25 EPROM MODIFICATIONS

In a standalone or transmit-only environment, most box operations can be tailored to your specific application by modifying the EPROM in the box. Whatever is stored in the EPROM becomes the new power-up default. All box operations can still be modified via serial commands as described elsewhere.

Please note that should you make these EPROM modifications, you are largely on your own. We cannot analyze your EPROM's for wrong data, wrong addresses, wrong checksum, creating weird combinations of features, and things of that sort.

The proper procedure for making EPROM modifications is:

- 1) In the charts which follow, indicate the changes to be made. This is necessary for debugging and technical support reasons.
- 2) Obtain a blank 2764A EPROM (with 12.5V programming voltage).
- 3) Copy the entire AEC-BOX-25 EPROM into your EPROM programmer. Save the original EPROM in case your application changes, or in case your EPROM modifications don't work.
- 4) Using the EPROM programmer, make changes as desired. Make sure the data in the byte you are changing matches the default values shown in the table below.
- 5) Be sure to calculate and save the new checksum byte(!).
- 6) Program your new AEC-BOX-25 EPROM.
- 7) Install the new EPROM, then turn on the box. If the LED fails to come on, you probably have a checksum error. You may also have modified something you shouldn't have.

*** IMPORTANT ***

You are authorized to copy and make changes to the firmware (EPROM) for this product only for the memory locations described below. Any other copies or changes constitute an infringement of our copyright (and probably won't work either).

Address	Description	Default Value	New Value
0060h	VITC Reader 1st Line Number	00h	_____
0061h	VITC Reader 2nd Line Number (see page 12)	00h	_____
0064h	Time Code Mode Control Byte #1	ECh	_____
0065h	Time Code Mode Control Byte #2 (see page 27)	00h	_____

AEC-BOX-25 EPROM MODIFICATIONS
(continued)

Address	Description	Default Value	New Value
0067h	LTC Generator Embedded Bits	00h	_____
0068h	LTC Generator Time Bit Frames	00h	_____
0069h	LTC Generator Time Bit Seconds	00h	_____
006Ah	LTC Generator Time Bit Minutes	00h	_____
006Bh	LTC Generator Time Bit Hours	00h	_____
006Ch	LTC Generator User Bit Frames	00h	_____
006Dh	LTC Generator User Bit Seconds	00h	_____
006Eh	LTC Generator User Bit Minutes	00h	_____
006Fh	LTC Generator User Bit Hours	00h	_____
0070h	Comparator Frames Data	00h	_____
0071h	Comparator Seconds Data	00h	_____
0072h	Comparator Minutes Data	00h	_____
0073h	Comparator Hours Data	00h	_____
0074h	Comparator Frames Mask	3Fh	_____
0075h	Comparator Seconds Mask	7Fh	_____
0076h	Comparator Minutes Mask	7Fh	_____
0077h	Comparator Hours Mask	3Fh	_____
0078h	Comparator Operating Mode (see page 17)	01h	_____
0080h	ASCII Mode Upper Address Byte	30h["0"]	_____
0081h	ASCII Mode Lower Address Byte	30h["0"]	_____
0082h	Binary Mode Upper Address Byte	AEh	_____
0083h	Binary Mode Upper Address Byte	C0h	_____
0090h	Box Operating Mode (SMPTE/NTSC)	06h	_____
0090h	Box Operating Mode (EBU/PAL) (see page 28)	16h	_____
1FFFh	Checksum Byte (3)	??h	_____

Notes:

- (1) Time and user bit data and mask bytes are all packed BCD.
- (2) Unless noted otherwise, the format for these bytes is the same as that described in their respective sections in this manual.
- (3) The checksum is chosen so that the single byte sum of ALL of the bytes in the EPROM is 00h.

BINARY BROADCAST MODE

At the start of every LTC generator frame, the AEC-BOX-25 will use 8 data bits per character to transmit the following message, which indicates the current LTC generator status:

Byte 0 = BREAK (20 bits low, then 2 bits high)
Byte 1 = Xh Status and length byte:
Bit 7 = 1
Bit 6 = 1 if the VITC input is OK
Bit 5 = 1 if the video input is OK
Bits 3-0 = # of bytes which follow (including checksum)

Byte 2 = 00h-29h Time bits frames (packed BCD) (EBU=24h max)
Byte 3 = 00h-59h Time bits seconds (packed BCD)
Byte 4 = 00h-59h Time bits minutes (packed BCD)
Byte 5 = 00h-23h Time bits hours (packed BCD)

Byte 6 = 00h-FFh User bits frames
Byte 7 = 00h-FFh User bits seconds
Byte 8 = 00h-FFh User bits minutes
Byte 9 = 00h-FFh User bits hours

Byte 10 = Xh Embedded bits (plus the EBU flag):

	SMPTE	EBU
Bit 7 = reserved	-----	-----
Bit 6 = EBU flag	0	1
Bit 5 = LTC bit 59	bin flag	phase bit
Bit 4 = LTC bit 58	reserved	reserved
Bit 3 = LTC bit 43	bin flag	bin flag
Bit 2 = LTC bit 27	phase bit	bin flag
Bit 1 = LTC bit 11	color frame flag	color lock flag
Bit 0 = LTC bit 10	drop frame flag	reserved

Byte 11 = Xh Status byte:
Bit 6 = 1 if VITC input indicates FORWARD tape direction
Bit 5 = 1 if VITC input is PLAYING (+0.8x to +1.2x)
Bit 4 = 1 if VITC input is FAST (greater than 1.2x)
Bit 3 = 1 if VITC input is SLOW (0.1x to 0.8x)
Bit 2 = 1 if VITC input is STOPPED (0.1x or less)

Byte 12 = Xh Checksum (sum of bytes 1-12 should be 0)

Notes:

- 1) All embedded bits have been removed from the time bits data, and appear instead in byte 10.
- 2) If a nonstandard video vertical sync pulse is detected, the "video OK" bit will be a "0", but the "VITC OK" bit will usually still be a "1" (if the data is OK).
- 3) In the reverse tape direction, the PLAY flag is not used. Instead, the SLOW flag will cover the 0.1x to 1.2x range.
- 4) If all of the speed indicator flags are "0", the box is unable to detect the speed, usually because of poor signal quality.

BINARY POLLED MODE

The box does not send any data until requested to do so. The protocol conforms closely with the ESBUS machine control standard (when 38400 baud and EVEN parity is selected) which is widely used in the television equipment industry. We highly recommend that you obtain a copy of the RP113 standard from SMPTE (see page 40). If you are not familiar with this standard, the state diagram (for AEC-BOX's) on page 23 will be very helpful. Eight data bits are used for all messages in this mode.

Controllers and Tributaries:

In the ESBUS standard, there is one "controller" (typically your computer) which controls one or more "tributaries". In this case, each AEC-BOX is a "tributary", and must have its own unique SELECT address and its own unique (but related) POLL address.

SELECT Address:

The default SELECT address of AEC0h is stored in the EPROM, and may be changed if needed (see page 18). DIP switches 5 through 8 form an address offset of 0(0000), 2(0010), 4(0100), or 6(0110) (switches 5 and 8 are always zero). This DIP switch address offset is then added to the address stored in the EPROM to form the SELECT address for this particular box. For instance, if switch 6 is a "1", and switch 7 is a "0", the box SELECT address will become AEC4h. This allows you to have up to four boxes on the same data line, each with a different address, without having to do any EPROM modifications. The MSB of each SELECT address byte must be "1", and the LSB of the lower SELECT address byte must be "0".

POLL Address:

The POLL address is equal to the SELECT address plus 1.

Inter-Character Timeouts:

In all cases where two or more bytes are being sent in a message, including two-byte addresses, if the space between any two of these bytes exceeds six character periods, a timeout error will be indicated. The box will set its internal NAK flag and revert to the IDLE state if it detects any such errors.

Response Timeouts:

Whenever the controller expects a reply from the AEC-BOX, the first character in the reply message must arrive within six character periods of the end of the previous message. Otherwise the controller can assume that the box is not responding.

BINARY POLLED MODE
(continued)

IDLE State:

The IDLE state is entered:

- 1) When power is first applied or the box resets itself.
- 2) Whenever any protocol errors are detected.
- 3) Whenever any parity or framing errors are detected.
- 4) Whenever a timeout error is detected.

The IDLE state is exited whenever a BREAK character is received.

ACTIVE State:

The ACTIVE state is entered whenever a BREAK character is received, no matter what else the box was doing. The box immediately ceases all transmissions, then waits for a two byte address (with inter-character timeout):

- 1) If the box's POLL address is received, it immediately jumps to the POLL state.
- 2) If some other POLL address is received, it reverts to the ACTIVE state.
- 3) If the box's SELECT address is received, it immediately jumps to the SELECT state.
- 4) If some other SELECT address is received, it goes back to the IDLE state and waits for a BREAK character.

POLL State:

This state is used to quickly determine which tributary devices (such as our box) have information ready for the controller. The AEC-BOX immediately transmits a single byte response, then reverts to the ACTIVE state. The POLL response byte may be:

- 1) 07h (RST) - Indicates that the box has powered up or reset.
- 2) 05h (NAK) - Indicates that a communications or protocol error of some kind has been detected.
- 3) 08h (SVC) - Indicates that the box needs to be serviced.
For the AEC-BOX-25, this means that a new frame of LTC generator data started, or that the comparator found a match.
- 4) 04h (ACK) - Indicates that nothing has changed.

The above responses are listed in order of priority (RST highest).

A typical POLL sequence from the controller might be:

Byte 0 = BREAK (20 bits low, then 2 bits high)
Byte 1 = AEh Upper POLL address byte.
Byte 2 = C1h Lower POLL address byte (LSB=1 !).

The POLL response byte from the AEC-BOX might then be:

Byte 1 = 04h ACK response byte indicating all is OK.

BINARY POLLED MODE
(continued)

SELECT State:

This state is used for sending messages back and forth between the AEC-BOX and the controller. It includes the ability to place the box in the BINARY BROADCAST mode. If any errors are detected, the box sets the NAK flag and jumps to the IDLE state.

A typical SELECT sequence from the controller might be:

Byte 0 = BREAK (20 bits low, then 2 bits high)
Byte 1 = AEh Upper SELECT address byte.
Byte 2 = C0h Lower SELECT address byte (LSB=0 !).

Once in the SELECT state, the box will remain there unless errors are detected or a BREAK character is received. The following commands may be sent to the AEC-BOX-25 whenever it is in the SELECT state:

- 1) GET BOX ID command format (to AEC-BOX) is:
Byte 1 = 3Fh Question mark ("?")

The response to this command (from AEC-BOX) is:

Byte 1 = 25d AEC-BOX Number (25 in this case)
Byte 2 = 41h-5Ah Software Revision Letter (major) (A-Z)
Byte 3 = 31h-39h Software Revision Number (minor) (1-9)
Byte 4 = XXh Checksum (sum of bytes 1-4 should be 0)

BINARY POLLED MODE
(continued)

- 2) GO TO BROADCAST MODE command format (to AEC-BOX) is:
Byte 1 = 1Bh(ESC) Escape to BINARY BROADCAST mode.

The response to this command (from AEC-BOX) is:
Byte 1 = 04h(ACK) Acknowledges receipt of this command.

The box will now be in the BINARY BROADCAST mode. You can force it back to the ACTIVE state with a BREAK character.

- 3) READ FULL MESSAGE command format (to AEC-BOX) is:
Byte 1 = 09h(TEN) Transmit Enable

The response to this command is a single message identical to what you would expect from the BINARY BROADCAST mode.

- 4) READ TIME BITS command format (to AEC-BOX) is:
Byte 1 = 40h Requests time bits (with no embedded bits).

The response to this command (from AEC-BOX) is:
Byte 1 = 00h-29h Time bits frames (packed BCD) (EBU=24h max)
Byte 2 = 00h-59h Time bits seconds (packed BCD)
Byte 3 = 00h-59h Time bits minutes (packed BCD)
Byte 4 = 00h-23h Time bits hours (packed BCD)
Byte 5 = XXh Checksum (sum of bytes 1-5 should be 0)

If you are reading VITC reader data instead of the usual LTC generator data, and if the VITC data is no good, the sum of bytes 1-5 will be 5Ah.

- 5) SET TIME BITS command format (to AEC-BOX) is:
Byte 1 = 41h Sets generator time bits.
Byte 2 = 00h-29h Time bits frames (packed BCD) (EBU=24h max)
Byte 3 = 00h-59h Time bits seconds (packed BCD)
Byte 4 = 00h-59h Time bits minutes (packed BCD)
Byte 5 = 00h-23h Time bits hours (packed BCD)
Byte 6 = XXh Checksum (sum of bytes 1-6 should be 0)

The response to this command (from AEC-BOX) is:
Byte 1 = 04h(ACK) Acknowledges receipt of this command.

This command must be received before the midpoint of video field 2 in order for this data to be used in the next frame. Otherwise it will be used in the following frame.

BINARY POLLED MODE
(continued)

6) READ USER BITS command format (to AEC-BOX) is:

Byte 1 = 42h Requests user bits.

The response to this command (from AEC-BOX) is:

Byte 1 = 00h-FFh User bits frames

Byte 2 = 00h-FFh User bits seconds

Byte 3 = 00h-FFh User bits minutes

Byte 4 = 00h-FFh User bits hours

Byte 5 = XXh Checksum (sum of bytes 1-5 should be 0)

If you are reading VITC reader data instead of the usual LTC generator data, and if the VITC data is no good, the sum of bytes 1-5 will be 5Ah.

7) SET USER BITS command format (to AEC-BOX) is:

Byte 1 = 43h Sets generator user bits.

Byte 2 = 00h-FFh User bits frames

Byte 3 = 00h-FFh User bits seconds

Byte 4 = 00h-FFh User bits minutes

Byte 5 = 00h-FFh User bits hours

Byte 6 = XXh Checksum (sum of bytes 1-6 should be 0)

The response to this command (from AEC-BOX) is:

Byte 1 = 04h(ACK) Acknowledges receipt of this command.

This command must be received before the midpoint of video field 2 in order for this data to be used in the next frame.

BINARY POLLED MODE
(continued)

8) READ EMBEDDED BITS command format (to AEC-BOX) is:

Byte 1 = 44h Command Code

The response to this command (from AEC-BOX) is:

Byte 1 = XXh Embedded bits:

	SMPTE	EBU
Bit 5 = LTC bit 59	bin flag	phase bit
Bit 4 = LTC bit 58	reserved	reserved
Bit 3 = LTC bit 43	bin flag	bin flag
Bit 2 = LTC bit 27	phase bit	bin flag
Bit 1 = LTC bit 11	color frame flag	color lock flag
Bit 0 = LTC bit 10	drop frame flag	reserved

Byte 2 = XXh Checksum (sum of bytes 1-2 should be 0)

If you are reading VITC reader data instead of the usual LTC generator data, and if the VITC data is no good, the sum of bytes 1-2 will be 5Ah.

9) SET EMBEDDED BITS command format (to AEC-BOX) is:

Byte 1 = 45h Command Code
Byte 2 = XXh Has same format as returned by command 44h.
Byte 3 = XXh Checksum (sum of bytes 1-3 should be 0)

The response to this command (from AEC-BOX) is:

Byte 1 = 04h(ACK) Acknowledges receipt of this command.

If you are using SMPTE time code, the DROP FRAME flag (bit 0) can be used to force generator drop frame counting.

While the COLOR FRAME and COLOR LOCK flags can be set as you wish, our AEC-BOX's cannot properly color align the LTC output with the video input. To us, fields 1,3(,5,7) are identical.

The phase bit information you supply is ignored, as the phase bit is calculated internally each frame by the generator.

This command must be received before the midpoint of video field 2 in order for this data to be used in the next frame. Otherwise it will be used in the following frame.

BINARY POLLED MODE
(continued)

- 10) READ TIME CODE STATUS BITS command format (to AEC-BOX) is:
Byte 1 = 46h Command Code

The response to this command (from AEC-BOX) is:

Byte 1 = XXh Time code STATUS byte #1:
 Bit 7 = 1 if a nonstandard vertical sync pulse is detected
 Bit 6 = 1 if the VITC input is OK
 Bit 5 = 1 if the video input is OK (normal)
 Bit 4 = 1 if the LTC generator is locked to the video input
 Bit 3 = reserved
 Bit 2 = 1 if the LTC generator output is disabled
 Bit 1 = reserved
 Bit 0 = reserved
Byte 2 = XXh Time code STATUS byte #2:
 Bit 7 = reserved
 Bit 6 = 1 if VITC input indicates FORWARD tape direction
 Bit 5 = 1 if VITC input is PLAYING (+0.8x to +1.2x)
 Bit 4 = 1 if VITC input is FAST (greater than 1.2x)
 Bit 3 = 1 if VITC input is SLOW (0.1x to 0.8x)
 Bit 2 = 1 if VITC input is STOPPED (0.1x or less)
 Bit 1 = reserved
 Bit 0 = reserved
Byte 3 = XXh Checksum (sum of bytes 1-3 should be 0)

- 11) SET TIME CODE MODE BITS command format (to AEC-BOX) is:

Byte 1 = 47h Command Code
Byte 2 = XXh Time code MODE byte #1 (default = ECh):
 Bit 7 = 1 if want to jam VITC time bits to the LTC output
 Bit 6 = 1 if want to jam VITC user bits to the LTC output
 Bit 5 = 1 if want to jam VITC embedded bits to the output
 Bit 4 = 1 if want to add two counts to the VITC time bits
 Bit 3 = 1 if want to add one count to the VITC time bits
 Bit 2 = 1 if want to sync the LTC output to the video input
 Bit 1 =
 Bit 0 =
Byte 3 = XXh Time code MODE byte #2 (default = 00h):
 Bit 7 = 1 if want VITC reader data, not LTC generator data
 Bit 6 = 1 if want to compare VITC line 1 with line 2
 Bit 5 = 1 if want to compare VITC field 1 with field 2
 Bit 4 = 1 if want to disable the 10 second VITC data filter
 Bit 3 = 1 if want LTC output to drop out when the tape stops
 Bit 2 = 1 if want to disable the LTC generator output
 Bit 1 = 1 if want to freeze the LTC generator count
 Bit 0 =
Byte 4 = XXh Checksum (sum of bytes 1-4 should be 0)

The response to this command (from AEC-BOX) is:

Byte 1 = 04h(ACK) Acknowledges receipt of this command.

These bits can also be modified via the EPROM (see page 18).

BINARY POLLED MODE
(continued)

12) READ VITC LINE NUMBERS command format (to AEC-BOX) is:

Byte 1 = 4Ch Command Code

The response to this command (from AEC-BOX) is:

Byte 1 = 00d-28d First video line number being used.

Byte 2 = 00d-30d Second video line number being used.

Byte 3 = XXh Checksum (sum of bytes 1-3 should be 0)

Note that the VITC line numbers returned by this command may be different from the ones you specified. The program forces the numbers to be valid if it found anything wrong with the data you supplied (or programmed in the EPROM).

13) SET VITC LINE NUMBERS command format (to AEC-BOX) is:

Byte 1 = 4Dh Command Code

Byte 2 = 00d-28d First video line number to be used.

Byte 3 = 00d-30d Second video line number to be used.

Byte 4 = XXh Checksum (sum of bytes 1-4 should be 0)

The response to this command (from AEC-BOX) is:

Byte 1 = 04h(ACK) Acknowledges receipt of this command.

See the VITC READER OPERATIONS section of this manual (on page 12) for a discussion of which line numbers to use.

14) SET BOX MODE command format (to AEC-BOX) is:

Byte 1 = 63h Command Code

Byte 2 = 06h/16h Box mode control bits:

 Bit 7 = 1 disables auto NTSC/PAL switching (default = 0)

 Bit 6 = 0

 Bit 5 = 0

 Bit 4 = 1 if want to use EBU/PAL inputs and outputs

 Bit 3 = 0

 Bit 2 = 1

 Bit 1 = 1

 Bit 0 = 0

Byte 3 = XXh Checksum (sum of bytes 1-3 should be 0)

The response to this command (from AEC-BOX) is:

Byte 1 = 04h(ACK) Acknowledges receipt of this command.

The EBU flag (bit 4) determines whether the box will operate with EBU/PAL(1) or SMPTE/NTSC(0) inputs and outputs. Effective with software revision "D1" in October 1993, the box software will automatically detect and adapt to whatever video input is present. This automatic switching capability can be disabled by setting mode bit 7 to "1".

BINARY POLLED MODE
(continued)

- 15) SET COMPARATOR DATA command format (to AEC-BOX) is:
Byte 1 = 70h Command Code
Byte 2 = 00h-FFh Comparator frames DATA
Byte 3 = 00h-FFh Comparator seconds DATA
Byte 4 = 00h-FFh Comparator minutes DATA
Byte 5 = 00h-FFh Comparator hours DATA
Byte 6 = XXh Checksum (sum of bytes 1-6 should be 0)

The response to this command (from AEC-BOX) is:
Byte 1 = 04h(ACK) Acknowledges receipt of this command.

- 16) SET COMPARATOR MASK command format (to AEC-BOX) is:
Byte 1 = 71h Command Code
Byte 2 = 00h-FFh Comparator frames MASK
Byte 3 = 00h-FFh Comparator seconds MASK
Byte 4 = 00h-FFh Comparator minutes MASK
Byte 5 = 00h-FFh Comparator hours MASK
Byte 6 = XXh Checksum (sum of bytes 1-6 should be 0)

The response to this command (from AEC-BOX) is:
Byte 1 = 04h(ACK) Acknowledges receipt of this command.

- 17) SET COMPARATOR/RELAY MODE command format (to AEC-BOX) is:
Byte 1 = 72h Command Code
Byte 2 = XXh Comparator MODE byte (see page 17).
Byte 3 = XXh Checksum (sum of bytes 1-3 should be 0)

The response to this command (from AEC-BOX) is:
Byte 1 = 04h(ACK) Acknowledges receipt of this command.

- 18) READ COMPARATOR/RELAY STATUS command format (to AEC-BOX) is:
Byte 1 = 73h Command Code

The response to this command (from AEC-BOX) is:
Byte 1 = XXh Comparator STATUS byte (see page 17).
Byte 2 = XXh Checksum (sum of bytes 1-2 should be 0)

ASCII BROADCAST MODE

This protocol is much simpler than the BINARY BROADCAST protocol, but it is also much more limited in its capabilities. At the start of every LTC generator frame, the AEC-BOX-25 will use 7 data bits per character to transmit the following message:

Byte 1	=	30h-32h	0-2	Time bits hours, tens digit.
Byte 2	=	30h-39h	0-9	Time bits hours, units digit.
Byte 3	=	3Ah	:	Separator
Byte 4	=	30h-35h	0-5	Time bits minutes, tens digit.
Byte 5	=	30h-39h	0-9	Time bits minutes, units digit.
Byte 6	=	3Ah	:	Separator
Byte 7	=	30h-35h	0-5	Time bits seconds, tens digit.
Byte 8	=	30h-39h	0-9	Time bits seconds, units digit.
Byte 9	=	3Ah/2Eh	:/.	Separator (see note 1 below)
Byte 10	=	30h-32h	0-2	Time bits frames, tens digit.
Byte 11	=	30h-39h	0-9	Time bits frames, units digit.
Byte 12	=	20h/3Fh	/?	Status character (see note 2 below)
Byte 13	=	0Dh	CR	Carriage Return

Notes:

- 1) Byte 9 will be an ASCII period (2Eh) if SMPTE drop frame counting is being used. Otherwise it will be a normal ":" separator (3Ah).
- 2) Byte 12 will be an ASCII question mark (3Fh) if any VITC read errors are found or if the video input is missing or bad. Otherwise byte 12 will be an ASCII space (20h).

ASCII POLLED MODE

This protocol is much simpler than the BINARY POLLED protocol, but it is also much more limited in its capabilities. The box does not send any data until requested to do so. Seven data bits are used for all messages in this mode.

- 1) TIME REQUEST command format (to AEC-BOX) is:
Byte 1 = 02h STX Start Transmission
Byte 2 = 30h-39h 0-9 Upper Address Byte (from EPROM)
Byte 3 = 30h-39h 0-9 Lower Address Byte (from EPROM) (3)
Byte 4 = 05h ENQ Enquiry
- 2) TIME REQUEST response format (from AEC-BOX) is identical to that listed for the ASCII BROADCAST mode on the previous page.
- 3) BOX ID REQUEST command format (to AEC-BOX) is:
Byte 1 = 02h STX Start Transmission
Byte 2 = 30h-39h 0-9 Upper Address Byte (from EPROM)
Byte 3 = 30h-39h 0-9 Lower Address Byte (from EPROM) (3)
Byte 4 = 3Fh ? Question Mark
- 4) BOX ID REQUEST response format (from AEC-BOX) is:
Byte 1 = 25d --- AEC-BOX Number (25 in this case)
Byte 2 = 41h-5Ah A-Z Software Revision Letter (major)
Byte 3 = 31h-39h 1-9 Software Revision Number (minor)
Byte 4 = 0Dh CR Carriage Return
- 5) BROADCAST MODE command format (to AEC-BOX) is:
Byte 1 = 02h STX Start Transmission
Byte 2 = 30h-39h 0-9 Upper Address Byte (from EPROM)
Byte 3 = 30h-39h 0-9 Lower Address Byte (from EPROM) (3)
Byte 4 = 1Bh ESC Escape
At this point, the box will enter the ASCII BROADCAST mode.
It will exit this mode upon receipt of an STX (02h) byte.

Notes:

- 1) If any transmission or protocol errors are detected, the LED on the front panel will blink off for about 500ms, and the software will wait for the next 02h STX byte.
- 2) There are no timeout periods between bytes.
- 3) The lower address byte is actually the sum of the EPROM value (normally ASCII "0") plus the two address bits on the DIP switch, which give address offsets 0, 2, 4, and 6. Thus if address bit 2 is a "1", and bit 1 is a "0", the default box address is "04". As shipped from the factory, the address will normally be "00".

BREAK CHARACTER DETECTION

The transmit data line on the microcomputer chip is normally high (inactive). When a normal serial data character is transmitted, the transmit pin first goes low for 1 bit period (the START bit), followed by 7 or 8 DATA bits, then a PARITY bit (if enabled), then finally goes high for 1 bit period (the STOP bit). The START bit for the next serial data character may start immediately thereafter.

Break characters are very different. A "break" character is defined as a special pulse which goes low for 20 bit periods, then goes back high for at least 2 bit periods.

The break character guarantees that the receiving UART will be properly locked to the serial data stream, even under worst case conditions. Since the binary mode message string includes user bits data, which can assume any value from 00h to FFh, the break character is also necessary to unambiguously define the start of a message string.

The break character can be detected in several ways:

- a) Some UART's have a break character flag and/or interrupt, which makes your job real easy.
- b) A break character will be received as 00h data together with a framing error.
- c) If odd parity is being used, a break character will cause reception of 00h data with a parity error.

RS232 Note:

The RS232 output signal on the 9-pin "D" connector has a polarity opposite from that described above. The TX232 line, which is normally low (at -6V), pulses high (to +6V) for 20 bit periods when the break character is transmitted, then goes back low.

SERIAL INTERFACE STANDARDS

For more information on any of these standards, contact the appropriate agency as indicated on page 40.

RS232 Standard:

Interface signals are inverted versions of the UART (TXD & RXD) signals. A valid "1" is -5V to -15V. A valid "0" is +5V to +15V. Since RS232 drivers are always on, you can't bus them together, which makes RS232 strictly a point-to-point communication link. RS232 is THE most commonly used interface in the computer industry, and is usually seen as a 25pin "D" connector on modems, terminals, serial ports (like IBM PC COM1 and COM2), and just about any type of peripheral you can think of. Cables should be limited to 30 meters max, and the data rate should be limited to 19200 baud max, in accordance with the RS232 standard. This AEC-BOX does not have (and does not need) any of the handshake lines which are used by many RS232 devices.

RS422 Standard:

The RS422 transmission standard allows for cables up to 1200 meters long, and data rates up to 10Megabaud. It uses differential (2 complementary line) transmitters and receivers, which greatly reduces sensitivity to common mode noise. In addition, the transmitters can be set to a high impedance (Hi-Z) state, which allows several transmitters to share a pair of data lines. Thus many pieces of equipment can share the same data bus. For the "+" output, typical output low voltages are about 0V, and typical output high voltages are about +4V. For the "-" output, the signal polarity is reversed.

In BROADCAST modes, the RS422 transmitter is always enabled. In POLLED modes, the RS422 transmitter is enabled only while messages are being sent. This enables multiple AEC-BOX's (and other devices) to be controlled from one RS422 port.

ESbus Standard:

The ESbus (EBU/SMPTE Machine Control Bus) is used in the television industry to control VTR's, routers, switchers, mixers, and other equipment. The pinout of the AEC-BOX's 9-pin "D" connector is that of an ESbus tributary. Experienced users will also note that the BINARY POLLED mode message protocol conforms with ESbus standards for a tributary device.

AEC-BOX-25 LED OPERATIONS

The so called "POWER" LED on the front of the box behaves in a variety of ways so that you can have some clues as to what is (or is not) going on inside the box.

When power is first turned on, a hardware reset circuit forces the LED to blink ON for a short (barely noticeable) time. If this fails to happen, there is something seriously wrong with the power supply or LED. Check the TROUBLESHOOTING section on page 35 for details.

If the LED blinks on initially, but then fails to come on any more, there must be some kind of hardware/software problem. This condition usually indicates an EPROM checksum error caused by improper user modifications. See the EPROM MODIFICATIONS section on page 18 for details.

If the "POWER" LED blinks OFF occasionally (or constantly), the box is working OK, but it is indicating that there is something unusual with the time code and/or serial data signals which are coming into the box.

The most serious problems, which blink the LED off for about 500ms per occurrence, are any serial data reception errors. These may include setup errors (wrong baud rate, parity, etc.), random communication errors (like parity or framing errors), or protocol errors. If there is no connection to the serial interface, it cannot make the LED blink, and you should read further.

The second problem, which blinks the LED off for about 250ms per occurrence, is any kind of video sync error. This may occur if the automatic NTSC/PAL mode switching has been disabled, if there are any tape dropouts or distortions, or if no video is present. If you are just using the box as a freerunning LTC generator, with no video input, this blinking will just be an annoyance. It can be disabled via EPROM modifications or serial commands.

The third problem, which blinks the LED off for about 50ms per occurrence, is the lack of a readable VITC input (after filtering) for one full second. This may be due to no VITC input, abnormal signal distortions, wrong VITC line numbers, or other problems. If you are just using the box as a video synchronized LTC generator, with no VITC input, this blinking will just be an annoyance. It can be disabled via EPROM modifications or serial commands.

Note that no matter how many errors are detected, the LED will always come on at least once per second. This way you will know that the power supply is OK.

If the "POWER" LED stays on all the time, everything must be running perfectly, and you can go read something else.

AEC-BOX-25 TROUBLESHOOTING GUIDE

This guide lists anticipated problems and their solutions. If you really get stuck, call our Service Department:

Problem #1: Power LED does not blink ON when power is applied:
Solutions : a) Check for presence of external AC power source.
b) Make sure AC voltage agrees with box wiring.
c) Leave off for 1 minute, then turn back on.
Thermal "fuse" will then be cooled and reset.
d) Fix broken LED wiring.
e) Return AEC-BOX for power supply repairs.

Problem #2: Power LED blinks ON initially, then stays off:
Solutions : a) May be an EPROM checksum error. If you have modified the EPROM, the checksum byte may be wrong. See the EPROM MODIFICATIONS section.
b) Return AEC-BOX for repairs.

Problem #3: Power LED blinks OFF during operations:
Solutions : a) See the LED OPERATIONS section of this manual.
Something is wrong with your signal source(s).

Problem #4: Box is not transmitting anything:
Solutions : a) Make sure switch 8 is a "1" (BROADCAST mode).
b) Make sure other DIP switches are set properly.
c) Use oscilloscope to check the transmit lines.
d) Remove external serial cabling - works now?
e) In POLLED mode, will not respond to incorrect box address, data errors, protocol errors, etc.
f) Make sure each box driving the same data line(s) has a different address.

Problem #5: Box transmits, but I can't receive anything:
Solutions : a) Check cables for shorts, opens, crossed wires.
b) Make sure UART parameters (baud rate, etc.) are the same for both the box and your serial device.
c) If using RS232, limit cable length to 30 meters.

Problem #6: Bytes are missing in the received data stream:
Solutions : a) The receiving device must be able to receive ALL bytes in a message without any OVERRUN errors, even if interrupts occur during reception.
Change interrupt priorities, disable some, etc.
b) Use a lower baud rate and see if the problem goes away (gives more time between characters).

Problem #7: Some time code messages and/or counts are missing:
Solutions : a) Use a higher baud rate so that the entire message can be sent before the next LTC frame arrives.
b) SMPTE drop frame counting eliminates some counts.
c) The VITC input may be counting faster than play speed, but the LTC generator is always at play speed, so has to skip some counts to keep up.

This is (was) a blank page.

WARRANTY REGISTRATION

We no longer have a formal warranty registration procedure, but do like to keep in touch with our end users. If you did not purchase this product directly from us, please copy the User Feedback Request form in the back of this manual, fill it out, then fax or mail it back to us. This way we will know who and where you are and be able to provide you with the following:

- 1) product upgrade and and bug reports,
- 2) manual updates and application notes,
- 3) safety/recall notices, and
- 4) better service in many other ways.

OUR WARRANTY

For the first two years following the shipment of an AEC product, we will repair or replace, at our option, any such product which is found to be inoperative due to defects in materials or workmanship. Not covered is damage due to unusual electrical and/or physical abuse. Altered hardware, software, labels, or other identifying marks may also void the warranty.

GENERAL GUIDELINES

Before sending a product back to us for service, please do the following (we've found over 90% of returned items work fine):

- 1) Check the "Troubleshooting Guide" in this manual.
- 2) Call our Service Department for assistance if needed.
- 3) Obtain our current return address, and possibly an RMA number, before shipping anything back to us.
- 4) Package the unit carefully before shipping it (it's yours).

WARRANTY SERVICE PROCEDURES

All you have to do is call our Service Department and describe the nature of the problem. We will attempt to fix it over the phone, but if that doesn't work we will give you an RMA number and you can ship the defective product back to us. We will repair or replace the product and return it to you as soon as possible.

OUT-OF-WARRANTY SERVICE PROCEDURES

If the two year warranty period has expired, or if the product has been altered or damaged, we will repair the product for a charge to be agreed upon before the repairs are begun. Call our Service Department for assistance. We have the test equipment, parts, and experience to quickly find and fix any problems.

7-BIT ASCII CODE CHART

Dec	Hex	Key	Char	Description	Dec	Hex	Char	Dec	Hex	Char
0	00h	^2	NUL	null char.	43	2B	+	86	56	V
1	01h	^A	SOH	start heading	44	2C	,	87	57	W
2	02h	^B	STX	start of text	45	2D	-	88	58	X
3	03h	^C	ETX	end of text	46	2E	.	89	59	Y
4	04h	^D	EOT	end of trans.	47	2F	/	90	5A	Z
5	05h	^E	ENQ	enquiry	48	30	0	91	5B	[
6	06h	^F	ACK	acknowledge	49	31	1	92	5C	\
7	07h	^G	BEL	ring bell	50	32	2	93	5D]
8	08h	^H	BS	backspace	51	33	3	94	5E	^
9	09h	^I	HT	horiz. tab	52	34	4	95	5F	~
10	0Ah	^J	LF	line feed	53	35	5	96	60	`
11	0Bh	^K	VT	vertical tab	54	36	6	97	61	a
12	0Ch	^L	FF	form feed	55	37	7	98	62	b
13	0Dh	^M	CR	carriage ret.	56	38	8	99	63	c
14	0Eh	^N	SO	shift out	57	39	9	100	64	d
15	0Fh	^O	SI	shift in	58	3A	:	101	65	e
16	10h	^P	DLE	data link esc	59	3B	;	102	66	f
17	11h	^Q	DC1	device ctrl 1	60	3C	<	103	67	g
18	12h	^R	DC2	device ctrl 2	61	3D	=	104	68	h
19	13h	^S	DC3	device ctrl 3	62	3E	>	105	69	i
20	14h	^T	DC4	device ctrl 4	63	3F	?	106	6A	j
21	15h	^U	NAK	no acknowldge	64	40	@	107	6B	k
22	16h	^V	SYN	synch. idle	65	41	A	108	6C	l
23	17h	^W	ETB	end TX block	66	42	B	109	6D	m
24	18h	^X	CAN	cancel	67	43	C	110	6E	n
25	19h	^Y	EM	end of medium	68	44	D	111	6F	o
26	1Ah	^Z	SUB	substitute	69	45	E	112	70	p
27	1Bh	^[ESC	escape	70	46	F	113	71	q
28	1Ch	^\	FS	file sepratr.	71	47	G	114	72	r
29	1Dh	^]	GS	group sep.	72	48	H	115	73	s
30	1Eh	^6	RS	record sep.	73	49	I	116	74	t
31	1Fh	^-	US	unit sepratr.	74	4A	J	117	75	u
32	20h	SPC		space	75	4B	K	118	76	v
33	21h	!	!	exclamation	76	4C	L	119	77	w
34	22h	"	"	double quote	77	4D	M	120	78	x
35	23h	#	#	number sign	78	4E	N	121	79	y
36	24h	\$	\$	dollar sign	79	4F	O	122	7A	z
37	25h	%	%	percent sign	80	50	P	123	7B	{
38	26h	&	&	ampersand	81	51	Q	124	7C	
39	27h	'	'	apostrophe	82	52	R	125	7D	}
40	28h	((left parenth	83	53	S	126	7E	~
41	29h))	right parenth	84	54	T	127	7F	DEL
42	2Ah	*	*	asterisk	85	55	U			

ASCII is an abbreviation for
 "the American Standard Code for Information Interchange".

NOTE: The keystrokes indicated above are for IBM PC's,
 and may be slightly different for your equipment.

NUMBER SYSTEM CONVERSION TABLE

This chart will help you make conversions between the various numbering systems which are used in this manual.

Hexadecimal	(MSB) Binary (LSB)	Decimal	BCD
0	0 0 0 0	0	0
1	0 0 0 1	1	1
2	0 0 1 0	2	2
3	0 0 1 1	3	3
4	0 1 0 0	4	4
5	0 1 0 1	5	5
6	0 1 1 0	6	6
7	0 1 1 1	7	7
8	1 0 0 0	8	8
9	1 0 0 1	9	9
A	1 0 1 0	10	invalid
B	1 0 1 1	11	invalid
C	1 1 0 0	12	invalid
D	1 1 0 1	13	invalid
E	1 1 1 0	14	invalid
F	1 1 1 1	15	invalid

(base 16)

(base 2)

(base 10)

(BCD is an abbreviation for "Binary Coded Decimal")

PACKED BCD NUMBERS

A "packed BCD" byte contains two BCD digits in an 8-bit byte. Bits 7-4 (upper nibble) contain the upper BCD digit, and bits 3-0 (lower nibble) contain the lower BCD digit.

For example, incrementing BINARY 09h leaves you with 0Ah, but incrementing PACKED BCD 09h leaves you with 10h. A packed BCD number such as 0Ah would be invalid, because "A" is not a valid BCD digit.

Here is one more example, showing the packed BCD format as used for time bits I/O. The 30 second (half minute) mark would be read (or written) as a 30h byte, even though 30 decimal is the same as binary 1Eh.

WHERE/HOW TO ORDER COPIES OF STANDARDS

We suggest that you fax, call, or write the organizations below for current prices and ordering/payment procedures. Due to copyright restrictions, we cannot provide standards copies for you. It takes some of these organizations up to 2 months to respond, so plan ahead. Also, please let us know if you find anything on this page which needs updating. Thanks.

SMPTE Engineering Standards Service
595 West Hartsdale Avenue
White Plains, NY 10607
U.S.A.

Tel: +1-914-761-1100 Fax: +1-914-761-3115

1) SMPTE 207M-1992	ESbus Electrical/Mechanical	\$16.00
2) SMPTE RP113-1992	ESbus Supervisory Protocol	\$16.00
3) SMPTE RP138-1992	ESbus Control Message Architecture	\$13.00
4) SMPTE RP139-1992	ESbus Tributary Interconnection	\$16.00
5) SMPTE RP163-1992	ESbus System Service Messages	\$16.00
6) SMPTE RP170-1993	ESbus VTR-Specific Messages	\$24.00
7) SMPTE RP172-1993	ESbus Common Messages	\$18.00
8) SMPTE 12M-1986	Time and Control Code for Television	\$16.00
9) SMPTE 262M	Data Storage & Trans. - Binary Groups	\$13.00
A) SMPTE RP159-1991	VITC and LTC Relationship	\$10.00
B) SMPTE RP164-1992	Location of VITC	\$10.00

European Broadcasting Union (EBU)
Technical Department
Case Postale 67
CH-1218 Grand-Saconnex/Geneve
SWITZERLAND

Tel: +41-22-717.21.11 Fax: +41-22-717.24.81
You must order the "Annual Volume of EBU Official Technical Texts",

which (for 250 Swiss Francs) includes the following:

- 1) N12 Time & Control Codes for Television
- 2) N18 Relationship Between Time Code and PAL 8-Field Sequence
- 3) I29 Recording of Information in User Bits

Telecommunications Industry Association (TIA)
(formerly part of Electronic Industries Association - EIA)
2500 Wilson Blvd.
Arlington, VA 22201
U.S.A.

Tel: +1-703-907-7700 Fax: +1-703-907-7727
These standards are sold only by Global Engineering Documents:
Tel: 1-800-854-7179 Tel: +1-303-397-2573 Fax: +1-303-397-2740

1) TIA/EIA Standard RS232-E	\$49.00
2) TIA/EIA Standard RS422-A	\$51.00
3) TIA/EIA Standard RS485	\$60.00