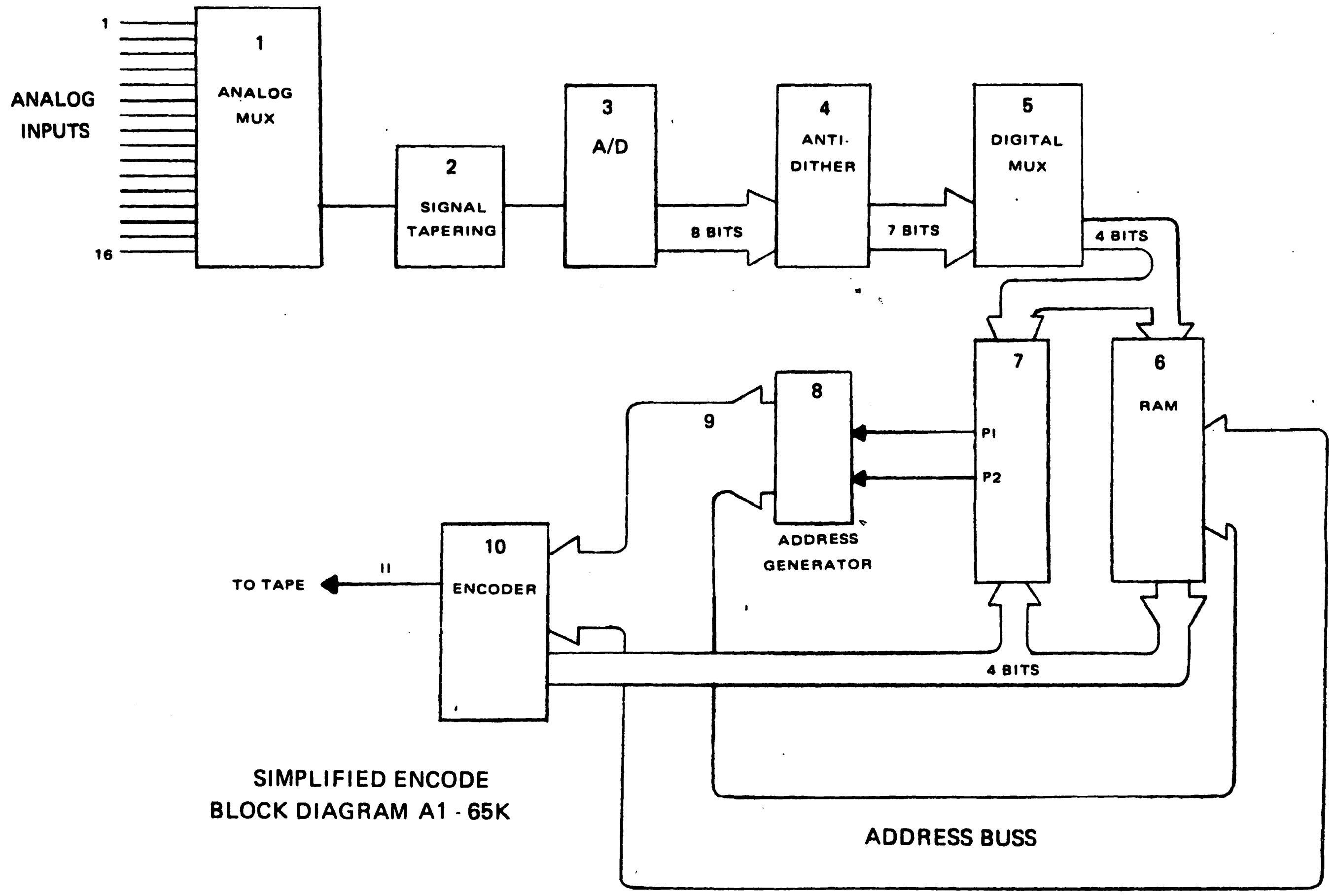


SIMPLIFIED ENCODE
BLOCK DIAGRAM A1 - 65K

Figure 2-1

CHAPTER TWO

OPERATIONAL OVERVIEW

Specifications and Capacities

The A1-65K-XX Programmer was designed to be the fastest, most reliable and most easily expanded automation available today. Its capabilities exceed amost every conceivable application. We will in this section outline just what it can do and cannot do, along with the procedure for expansion.

The A1-65K-XX Programmer can support up to 4096 analog control inputs. Each analog control voltage is — .6v to 5.6v. The encode inputs clamp their voltages to this range when an overvoltage is applied, so as not to create errors in the programmer's internal workings. Each of these analog control voltages is converted to an 8-bit digital word which represents 256 steps. The first 0-2.5v digital word is scaled such that their resolution is ½dB steps (25mv) and the last 3.lv are 1dB steps.

The programmer is organized as 64 I/O (input-output) ports of 64 channels each, giving it a capacity of 4096 analog control functions, as stated earlier. Each I/O port operates independently of any other I/O port in the system. Most 65K systems contain only one I/O port, since very few consoles contain more than 64 control channels. Because of this fact, the programmer was designed to be expanded in segments of 16 channels. In the programmer architecture, each I/O port consists of what we term an I/O pair (because 2 circuit boards are involved) and at least one multiplex card (MUX card). Each MUX card handles 16 analog control functions; therefore, the minimum size analog programmer is 16-channel. To expand the I/O port to its full capacity we would add 3 more MUX cards, giving us a full 64-channel programmer. If we wish to expand beyond this figure, we must add at least one more I/O pair. So, to sum up, each I/O pair can support 4 MUX cards, (64 channels), and the programmer architecture can support up to 64 I/O pairs. Expansion is always just a matter of plugging the proper combination of cards into the programmer itself or, if necessary, an expander package. (The A1-65K programmer only holds 1 I/O pair and 4 MUX cards.)

The Signal Path

The programmer contains two completely independent, simultaneously operating processes. It performs the encoding process, (the taking of a number of analog data (control) inputs and their conversion to a form suitable for storage on a master tape), while at the same time it can be decoding data (returning encoded data to its original form and its original location). Since these processes operate separately, they will be covered one at a time, starting with the encode signal path.

Encode Path

The block diagram in ***Figure 2-1*** gives a simplified view of the path (processes) that an analog input travels as it goes to be recorded on tape. The block diagram is drawn in function blocks so that a basic understanding may be achieved before plunging into the actual electronic details contained in the next chapter.

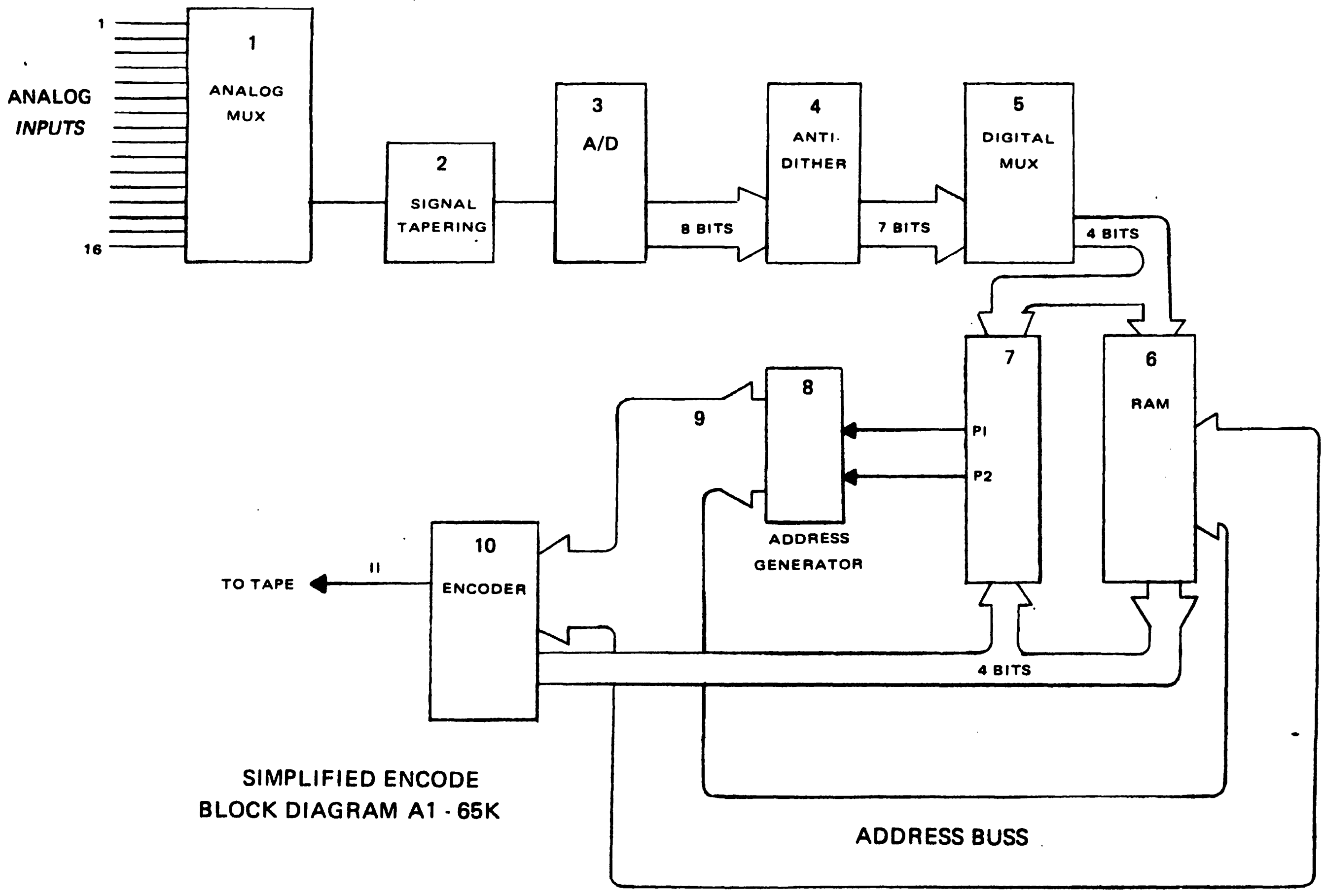
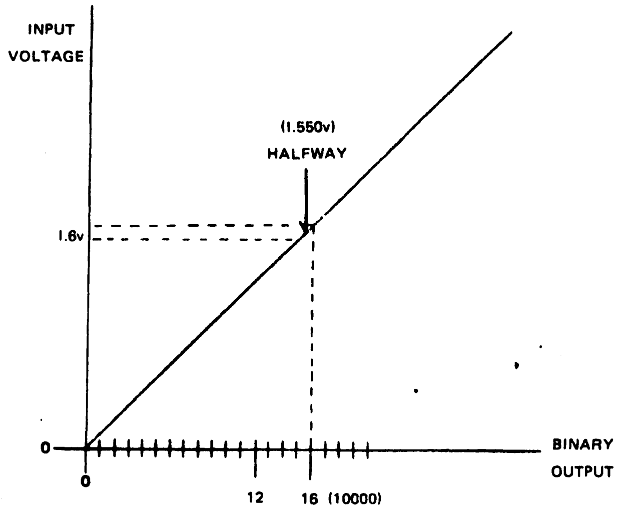


Figure 2-1



ANY VOLTAGE THAT IS HALFWAY BETWEEN 2 BINARY VALUES WILL DITHER BACK AND FORTH BETWEEN THE 2 VALUES AND CREATE FALSE PRIORITIES

Figure 2-2

As can be seen from ***Figure 2-1***, there are 7 processes through which the data must go before it is encoded on tape:

Block 1	Analog Multiplexing
Block 2	Signal tapering, filtering
Block 3	A/D Conversion
Block 4	Anti-Dithering
Block 5	Digital Multiplexing
Block 6	RAM Storage
Block 9	Encoding

The remaining 2 blocks (Priority Detector, Address Generator) are decision making and control functions, which affect the sequence in which the channels are recorded on tape.

Block 1 is the Analog Multiplexer. The control voltage outputs from your system hook up to the inputs of the multiplexer. The impedance of each input is 10K ohms. (This relatively low impedance is to prevent any unused (open) channels from generating data due to hum and R.F.I.) Each input has a diode clamp which limits input voltages to 5.6v maximum. The multiplexer places the voltage present at its input onto the encode analog (EA) line. Just which input is on the EA line is determined by system timing.

The 65K system timing places the multiplexer's inputs onto the EA line in sequence from Channel 1 to Channel 64. Each channel remains on the line about 64us (64/1,000,000ths seconds.) This process continues any time the programmer is operating.

The encode analog line is then filtered to remove any trash which might be with it, tapered to increase resolution in the 0 - 3.75v range, then it is fed to the analog - to - digital converter. The A/D converter performs an 8-bit conversion on the samples fed to it. It performs a complete conversion in 40us. At the end of its conversion, it places an 8-bit binary representation of the input in its output latches.

Due to the nature of A/D converters, it is possible to provide an input which can and (will) cause two different outputs. This situation happens any time the input voltage lies exactly halfway between 2 output binary values. (***Figure 2 - 2*** illustrates these conditions.)

While this occurrence by itself is not particularly bad, it will become evident as we start to explore the priority systems' workings that this condition should not be allowed to occur. This digital indecision in the A/D converter is known as DITHERING. Function Block 4 (***Figure 2 - 1***) is circuitry designed to prevent dithering. Its operation can be briefly explained as follows: The circuitry looks at what the A/D converter says a channel's output is, then compares it to what it was the last time it was converted. If it has changed more than one bit in value, the new value is sent to the digital multiplexer; if not, the old value is sent instead. This causes only genuine changes to be sent to the encoding circuitry.

After the signal has been anti-dithered, it is digitally multiplexed into 4-bit wide pieces. This is mainly a design convenience because many of the parts used internally operate in 4 bits per I.C.

The output 4 bits of the digital multiplexer are fed to a Random Access Memory. (RAM.) Before each piece is stored away, it is compared to its last value to see if it has changed. This is the priority detector block. If it did change, a FLAG is set to tell the address generator function block that this channel needs attention fast, i.e., it is a priority one.

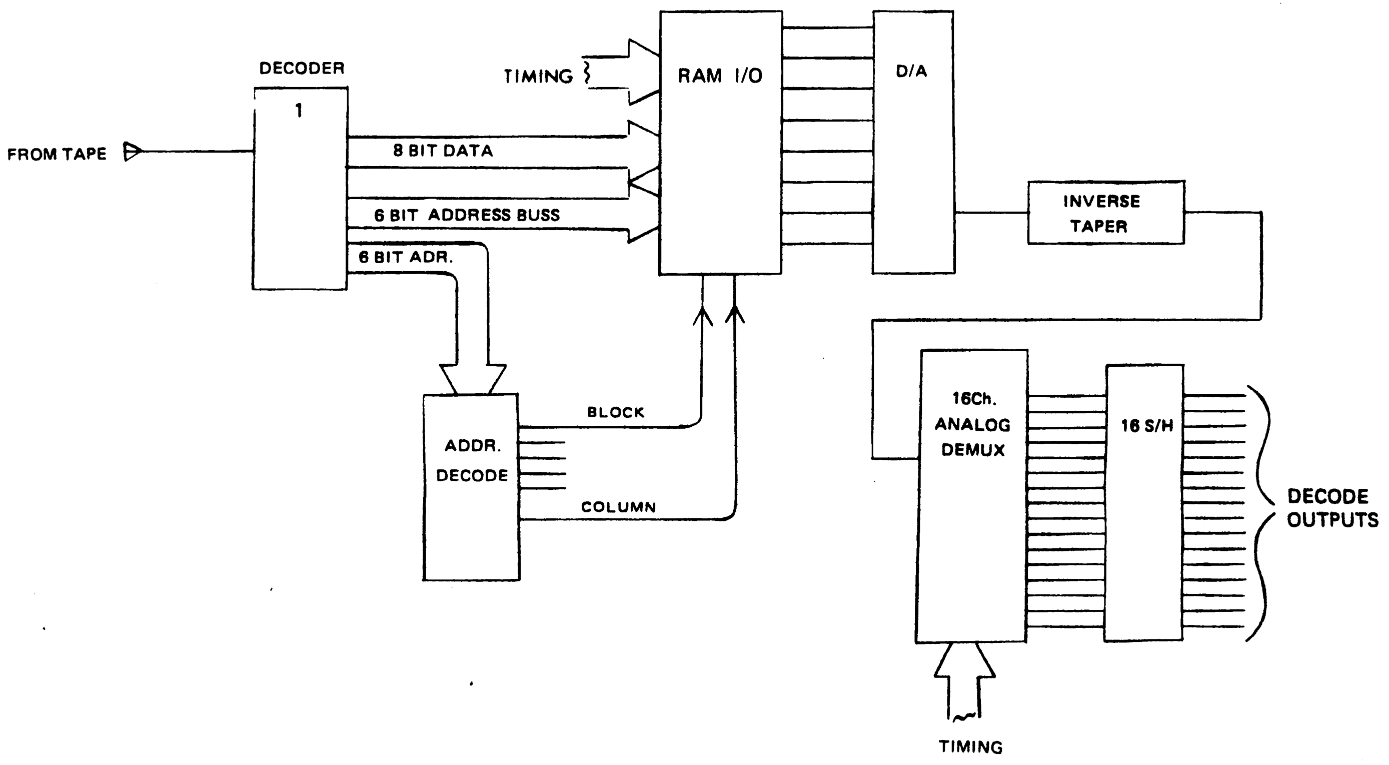

DECODER
1
TIMING
RAM I/O
D/A
FROM TAPE
8 BIT DATA
6 BIT ADDRESS BUSS
6 BIT ADR.
INVERSE TAPER
ADDR. DECODE
BLOCK
COLUMN
16Ch. ANALOG DEMUX
16 S/H
DECODE OUTPUTS
TIMING

Figure 2-3

The address generator makes the decision as to which channel's data is next to be encoded on the tape. How it makes this decision follows. In the time it takes to encode one word on tape, it scans through all 4096 possible channel locations, checking to see if there are any Priority 1 channels. If it locates one, it then begins its priority protocol. (This protocol establishes how the P1's are put on tape along with a certain amount of static (unchanged) data, and will be covered in depth in our discussion of Address Space and the Priority System.) When all information is static, the address generator sequences from Channel 1 to Channel 32 until a P1 occurs.

Once the address generator decides who is next to be encoded, the digital data is sent to the encoder. The encoder places the data along with parity in a serial (one bit at a time) form at the TO TAPE terminal on the rear of the programmer. The encoding process will be covered in detail in the Code Structure portion of this manual.

Decode Path

We will refer to ***Figure 2 - 3***, Decode Path Block Diagram, for the following discussion.

The decode path tends to be simpler than the encode path because it need not deal with priorities or address generators. The first block in the system is the Decoder itself. This very complex piece of gear takes the serial information from the tape machine, corrects for any speed variations, checks for errors, and then places the decoded digital word in its output register until it is time to send it to the I/O port terminal.

When it is time to send its data, the RAM on the I/O port will place itself in a state to receive information. The data is then written into a RAM location corresponding to the appropriate channel. The RAM then reverts to its refresh mode of operation until another data word is to be written. In this refresh mode the RAM is stepped (by system timing) from Channel 1 through Channel 64 sequentially. Each channel's digital data then appears at the RAM's output where a digital-to-analog converter turns it back into the voltage which it orginally represented.

If we were to watch the output of this D/A, we would see a display very similar to the one we looked at (encode analog) when we set up the unit in Channel 1. This decode analog line must be redistributed to the channel from which it came, so we use an analog de-multiplexer.

The de-multiplexer operates in the inverse manner as the multiplexer described in the encode path. It takes an analog voltage at its input and places it at the selected output. This output only remains valid as long as the multiplexer is told to put its input there, so we must add an analog memory (sample/hold unit) to each multiplexer output. These buffered analog memories then form the actual decode outputs on the rear of the programmer to which your system is connected.

This covers the encode-decode path on a very fundamental level. Specific hardware details have been purposely omitted in order to establish a basis upon which the highly technical chapters may build.

CALIBRATING YOUR PROGRAMMER

(Test procedure requires an X-Y scope with triggered sweep.)

Display Capabilities/Interpretations

Calibrating your programmer consists of a process of adjusting the analog interface circuitry to obtain optimum performance in your system. There are three trimpot adjustments used for this procedure. The adjustment should be performed with the 65K Programmer completely connected to your system. This will insure that the system operates as a whole.

The first step in the calibration procedure is to patch the "tape out" to the "tape in" terminals on the programmer. The decode light should now be lit. We must now make scope connections to the programmer:

1. Connect the trigger input of the scope to the terminal marked RESET.
2. Connect the vertical input to the terminal marked ENCODE ANALOG with 1 v/div sensitivity.
3. Set horizontal timebase to .5ms/div and adjust for a stable display. (***Figure 1***)
4. Enter the "WRITE" mode.
5. Move one fader up and down and note that its position moves down and up in voltage. (Top of fader is 0 volts.)
6. Arrange the faders such that their voltages form a staircase waveform. (***Figure 2***)
7. Switch the entire board between READ and WRITE slowly several times. (If your system does not have master READ/WRITE switching, you will have to have some help switching. You will also use fewer channels.) Note what happens to the display.
8. Under ideal conditions the displayed levels will jump up or down a little and then stop. (Null lights will not change.) If the displayed levels all climb or fall all the way, your system is in need of adjustment.
9. Refer to ***Figure 3.*** The card that we are concerned with is the Analog I/O card. (Before we start, it is a good idea to mark the starting positions of the three trimpots.) While the board is being switched between READ and WRITE turn the offset control until the display stabilizes.

LEFT

CENTER If the display splits about the middle such that the upper voltages rise and lower ones fall, it is necessary to adjust the breakpoint control to adjust for a stable display.

RIGHT If only the higher voltages change, it is necessary to adjust the full scale adjustment.

If the pots have been marked ahead of time, you can adjust all three for the most stable display. This is a rather sensitive adjustment procedure and it will probably take several passes to get it right. Remember that it is virtually impossible to make the display stand perfectly still. This adjustment will have to take the best average display since adjustment of each individual channel is prohibitive.

Once you are satisified that the display is adjusted to its optimum, we can use some of the programmer's other display capabilities to check its operation. To do this, you will need a scope with X-Y capabilities.

Watching All The Bits

Your programmer has built into it the ability to observe all 28 information carrying bits as they are decoded. The address buss (12 bits) and data buss (16 bits) are divided in half and each half is fed to a digital analog converter. When these analog outputs are properly displayed a form of signature analysis is available.

These outputs are terminated on the rear of the programmer. Their markings and interpretation are as follows:

A1-A6 — lower 6 bits of the address. They define 64 different locations with 64 different analog voltages. In a system containing 64 channels or less, this voltage is used to drive the horizontal (X) axis. This gives us 64 locations corresponding to each channel when displayed in this manner.

A7-A12 — these are the upper 6 bits of address, and define 64 I/O locations. This analog voltage is normally used to drive the vertical axis.

The combination of the above 2 outputs will allow you to watch your priority system in action. As was discussed earlier in the manual, the order in which channels are being encoded on tape is based on which channel(s) require(s) attention. When the programmer is looped "tape out" to "tape in" the analog displays show channels in the exact order that they would be encoded. So to watch the priority system you would connect A1-A6 to the X input (external horizontal) and A7-A12 to the vertical input (.6v/cm).

With your oscilloscope connected in that manner you will see a display like that in ***Figure 4.*** There will be one dot per channel in each I/O port. There will be one row of dots for each I/O port or address location in the system. Dots will appear to be sweeping from left to right at a good pace (¼sec/64 channels). If you now take one channel fader and begin to move it you will see its dot intensify and the scan of unchanging channels slow down. This indicates that the moving channel fader is now being recorded almost continuously, making its response and resolution very good. If you were to take all channel faders and move them all continuously, you would see what appears to be a dual scan rate. The fast scan is Priority 1 information being recorded; the slow is Priority 2 (unchanging) information. This display is continuously available on the E3-A test code card that can be obtained for your programmer. (See next section on the E3-A Card).

One other useful display is data vs. address. This is a double duty display in that it shows the priority system working along with the data that corresponds to the channel. (NOTE: This display can only be seen in systems containing one I/O card.) Connect the scope horizontal to A1-A6 and the vertical to D1-D8. If you apply the staircase as discussed in the calibrating portion of this chapter, you will see a display as in ***Figure 5.*** Note that the voltage is inverted; 0v at the fader is 5v on the display, and it is not linear. The tapering is caused in the analog adapter to create a greater resolution. An inverse taper is applied before the sample/hold on the MUX cards in order to maintain overall system linearity. You should note that D9-D16 contain no information in standard analog programmers.

There is only one other adjustment on the programmer that you might concern yourself with, and that is the master clock frequency pot. It is located on the E-1 Encode card. Connect the scope's external trigger (horizontal = .5ms/div) to the RESET terminal and the vertical inout to the TO TAPE terminal. When the scope is properly triggered you will see a display like ***Figure 6.*** As you turn the clock pot you will see the word length vary. This adjustment is nominally 4ms/word. You will probably never need to change this pot setting. The only time being when the tape media does not have the necessary frequency response (cassette) or if you need to mix at a lower tape speed.

The remaining trimpot is a level control for the FROM TAPE to decoder. This is normally set mid-range or until the DECODE light remains lit and stable.

TROUBLESHOOTING A PROGRAMMER

In this section we will cover some basic steps in fixing a malfunctioning programmer. The steps and hints outlined here will be an aid to the experienced troubleshooter, but due to the complexity of the programmer, we cannot possibly outline a step-by-step procedure; should the information not be complete enough, we recommend that you consult the factory.

The first step in fixing the programmer is to figure out as specifically as possible the problem encountered. These problems usually fall into two categories: 1) Programmer is dead, and 2) Programmer doesn't work right. If we outline a procedure for each situation, then you will have the groundwork to attempt a repair. Whenever you suspect a problem, the first thing to do is to hook TAPE OUT to TAPE IN on the programmer. If the decode light lights, your programmer isn't dead, so proceed to the "Doesn't Word Right" section.

Programmer Is Dead (No Decode Light)

1. Check the power supply voltages.
   - A. If O.K., go to Step 2.
   - B. If not O.K.:
     1. Turn off programmer for 20 seconds, then on and check again. If O.K., go to "Doesn't Work Right."
     2. Turn off programmer and unplug cards but leave them in their slots. Recheck voltages. If O.K., go to C.
     3. Remove leads from mother board and check.
        - a. if still nothing, suspect a power supply problem and remove it for closer scrutiny.
        - b. if it clears, carefully inspect mother board for shorts and after clearing any, replace power leads. (red = +12, orange = −12, white = +5, brown or black = ground.)
   - C. Power down and replace the cards one at a time, checking voltages after each card is installed. If one card then causes the power supply to drop, something on that card is dead.
     1. Use the extender card to get the bad card out to work on.
     2. Replace the card. If it contains a bad I.C., then the I.C. in question will probably be quite warm.
     3. If none are hot then they must be removed, one at a time, supplies checked and replaced one at a time until the fault is located.
     4. If the above tests fail to come up with anything, we suggest you contact the factory.
2. Check the TAPE OUT and TAPE IN terminals for a signal as described in Chapter Four **"Code Structures and Parity Sytems."**
   - A. If there is code present on both terminals then:
     1. Verify that it is 4ms./word
     2. Extend the decoder (D 65K) and attempt to follow the signal path through. If you would rather, you can consult the factory for more detailed trouble-shooting.
   - B. If there is no signal present on the T0 TAPE:
     1. Extend the E1 Encoder card and check all time periods (1 MHz through Q10).
     2. Check RESET pulse.
     3. Check Input to Output buffer amp (14503).

Programmer Doesn't Work Right

1. Look at the encode analog line and check to see that each fader causes an independent corresponding movement.
   - A. If the encode analog is not functioning, the problem lies either on the ANA-

LOG I/O or MUX cards.

B. If the encode analog is O.K., then we must check the digital signal path:
   1. Look at A1-A6 vs. A7-A12 to insure that only channels present in the system are being serviced. The display should only show one row of dots just above ground in potential. If E3 card is present its display should be scanning smoothly.
      If it is not operating the first suspect chips are the 14028 block, column chips on the E1 card, after those any 14503 that feeds an address buss (RAM or console).
   2. If the priority system appears to be functioning correctly then it is necessary to test the data path. To do this you will need to have an E3-A test code card in the system. Turn the card to the test position. The display on the card should now be showing the dual P1, P2 scan. If it is not you should re-check as in B1 above. Attach the scope to D1-D8 (horizontal) vs. D9-D16 (vertical). The display should be a lower left to upper right sweep of dots.
      a. if it is not, the output buffers on the D 65K card or any of the ANALOG I/O cards are suspect.
   3. The above tests do verify the operation of the programmer. Due to the sheer complexity of the programmer, it is impossible to write down a troubleshooting procedure to the I.C. level. If you have trouble locating a problem, remember to check signals that write RAMs or latch latches and clock counters. The combinational logic is more prone to failure than the complex devices because complex devices are usually well isolated from the outside world. We are always available at the factory to assist you.

GLOSSARY

(All definitions are as they apply to 65K Programmers)

A/D – shortened form meaning Analog to Digital converter – a device that takes an analog input and converts it to a binary number representation. Each output increment represents some per cent of the maximum input value. An 8 bit converter has 256 possible outputs; each step (increment) is 1/256 of the reference full scale value.

ANALOG – refers to a voltage that can be any value within the supply voltage limits. The value of voltage is determined by input and modifying circuitry. (see Ditigal) Analog also applies to the type of circuitry which processes varying voltages.

ADDRESS – a term referring to a group of binary bits specifying the location of some particular information, by the number which the bits represent.

BIT – (shortened form of binary digit – one line of digital information; it can only be a one or a zero, a one being represented by 5 volts, and a zero by zero volts (in 65K programmers.)

BINARY – the numbering system where ones and zeros are the only possible values that a line might have. Each line is one bit. A group of bits defines one binary number.

BUSS – a line (or lines) that connects to several devices, any of which can use these lines to communicate with another device on the buss. Who is occupying the buss is determined by system hardware. The information on a buss may be very specific (address buss), or dependent on the source (data buss).

C-MOS – a family of logic devices characterized by extremely high input impedance, and extremely low power consumption. The programmer is constructed almost exclusively of 4000 Series C-MOS parts.

CLAMP – a circuit that does not allow an applied voltage to exceed a pre-determined value. Most voltage clamps in the programmer are simply diodes to the power supply. This limits the voltage to approximately .6v higher than the supply to which they are wired.

CLOCK – a digital circuit that supplies a square-wave output used for a timing reference by all other digital circuits in the programmer.

COUNTER – a digital circuit that takes a square-wave input and outputs a frequency divided by some power of two, depending on which counter output is selected. If the counter's input is a constant pulse stream, its outputs are the binary representation of the number of pulses that have occurred (counting).

CODE – refers to the particular scheme by which information is recorded.

DECODE – the method of returning information to its original form from coded information. (In this case, from the serialized code back to its parallel binary representation.)

D/A – short for Digital to Analog converter – it is a device that takes a binary number input and outputs a voltage corresponding to the input number. (If the input is a binary "nine" then the output might be 9 mv.)

DEMULTIPLEX (DEMUX) – taking several analog (or digital) values that happen sequentially on one line and redistributing back to the many lines from which they came.

DIGITAL – refers to any system operating with binary "ones" and "zeroes" rather than continuous analog voltages.

ENCODE – refers to the process of turning binary numbers into the serial form to be recorded on tape. It also refers to the time period when new data to be encoded is gathered at each I/O port.

ENCODE ANALOG – the line from the input analog multiplexers to the A/D converter on each I/O-pair. The lines carries the current analog samples to be processed.

FLAG – a signal that may be read by other electronics. This bit signals that an event has occurred; in a programmer, it signals Priority 1's.

FLUX REVERSAL – on magnetic tape, a flux reversal occurs any time that the signal being recorded changes polarity (from + to – or – to +).

HIGH – refers to a voltage that represents a binary "one" (5v).

LATCH – a digital device that takes a series of binary bits presented to it and, upon command (latch command), remembers it on its output until it is instructed to accept a new number.

LOW – refers to a voltage that represents a binary "zero" (0 v).

MUB – Make Up Bit, a bit in the serial word recorded on tape that serves to indicate the end of each code word.

MULTIPLEXING (MUX) – taking a number of inputs (analog or digital) and placing them one at a time onto a single line.

PARITY – a method of checking a word that has been decoded. A parity system performs some sort of test and decides whether or not the data transfer was successful.

PARALLEL – implies many things present at the same time. In the case of a parallel binary value, it means that all of the bits are available simultaneously rather than one at a time (serial).

PORT – circuitry that serves to send data to and receive data from some device external to the digital internal workings.

RAM – Random Access Memory, a device used to store digital words. It contains a number of locations, each one accessed through a digital address presented to the RAM.

REFRESH – in programmer systems, refers to the sample/hold on each MUX card. Each S/H unit must be returned to the desired voltage periodically, due to the discharging of the storage capacitor.

RESET – the master timing signal in the programmer, used to indicate the start of a new programmer cycle. This is the trigger point used on an oscilloscope to watch the internal workings of the programmer.

SAMPLING – looking at and recording the value of a waveform at specific periodic intervals.

SERIAL — one bit at a time. It implies taking a number of bits present simultaneously and placing them one at a time on one line (digital multiplexing).

S/H — Sample and Hold, a device that is normally used as an analog memory. It will remember a voltage fed to it for some amount of time and then it must be refreshed (returned to original value) or changed.