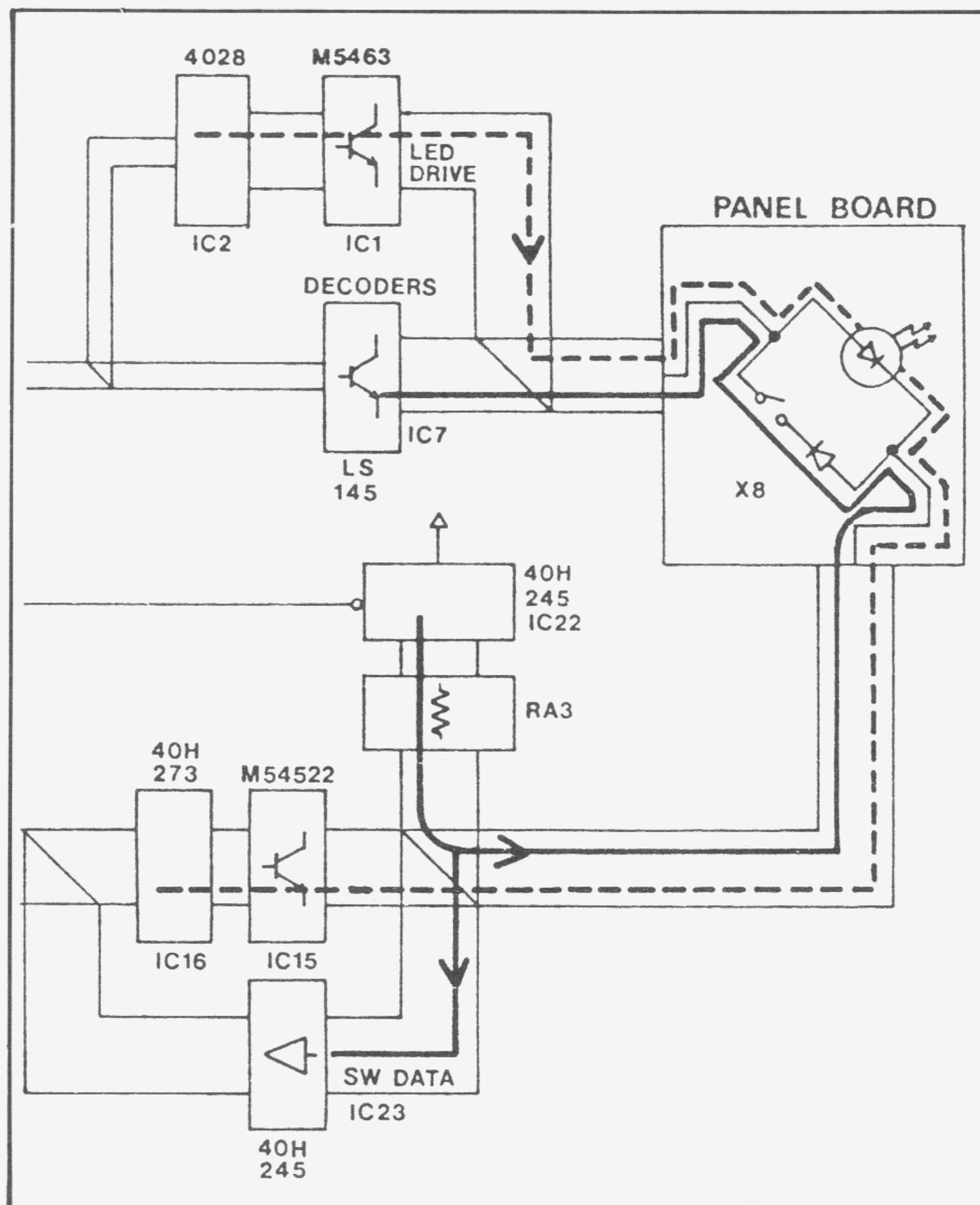


Reading switch states and driving LEDs



Reading switch states and driving LEDs are alternately repeated through 8 x 8 matrix (divided into the R and L PANEL BOARDs) using a single line.

1. Reading panel switch states

Turned on by the CPU, IC22 pulls the bus positive through RA3. Simultaneously, a designated bit of IC7 is pulled low. A closed switch contact in the low bit effectively lowers one of input pins of IC23. The combination of bits (at IC7 output and IC23 input pins) informs the CPU which switch has been pressed not pressed.

2. Lighting LEDs

IC22 is turned off by the CPU and the bus is now in a float state. At this time, IC2 (4028) decodes the applied address and has a high at the corresponding output of LED driver IC1.

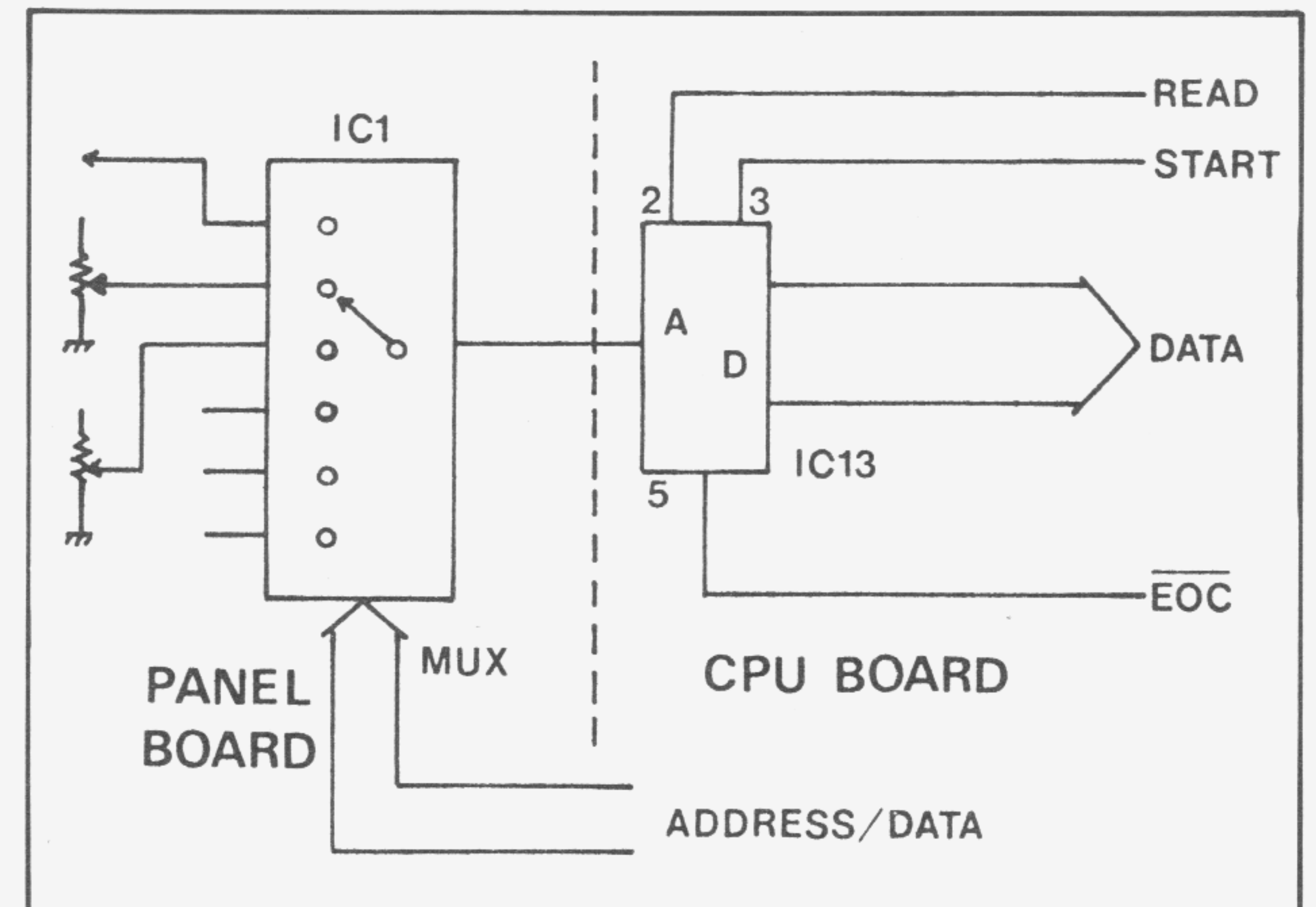
When an output of IC16 goes high, a transistor in IC15 saturates, allowing one of the 8 LEDs (max) to be lit for 2ms.

The above-mentioned operations, reading of panel switch states and lighting of LEDs, are repeated eight times (one cycle).

Reading potentiometer data

IC1 (Multiplexer) sequentially connects Panel potentiometers to IC13 (A/D converter). IC13 starts conversion when signaled by START derived from IC18 (Write Address Decoder) with \overline{WR} .

After A/D conversion, \overline{EOC} of IC13 goes low to inform the CPU of completion of conversion. Upon receiving the \overline{EOC} , the CPU outputs READ to accept the digital equivalent of a control setting.



Cassette interface

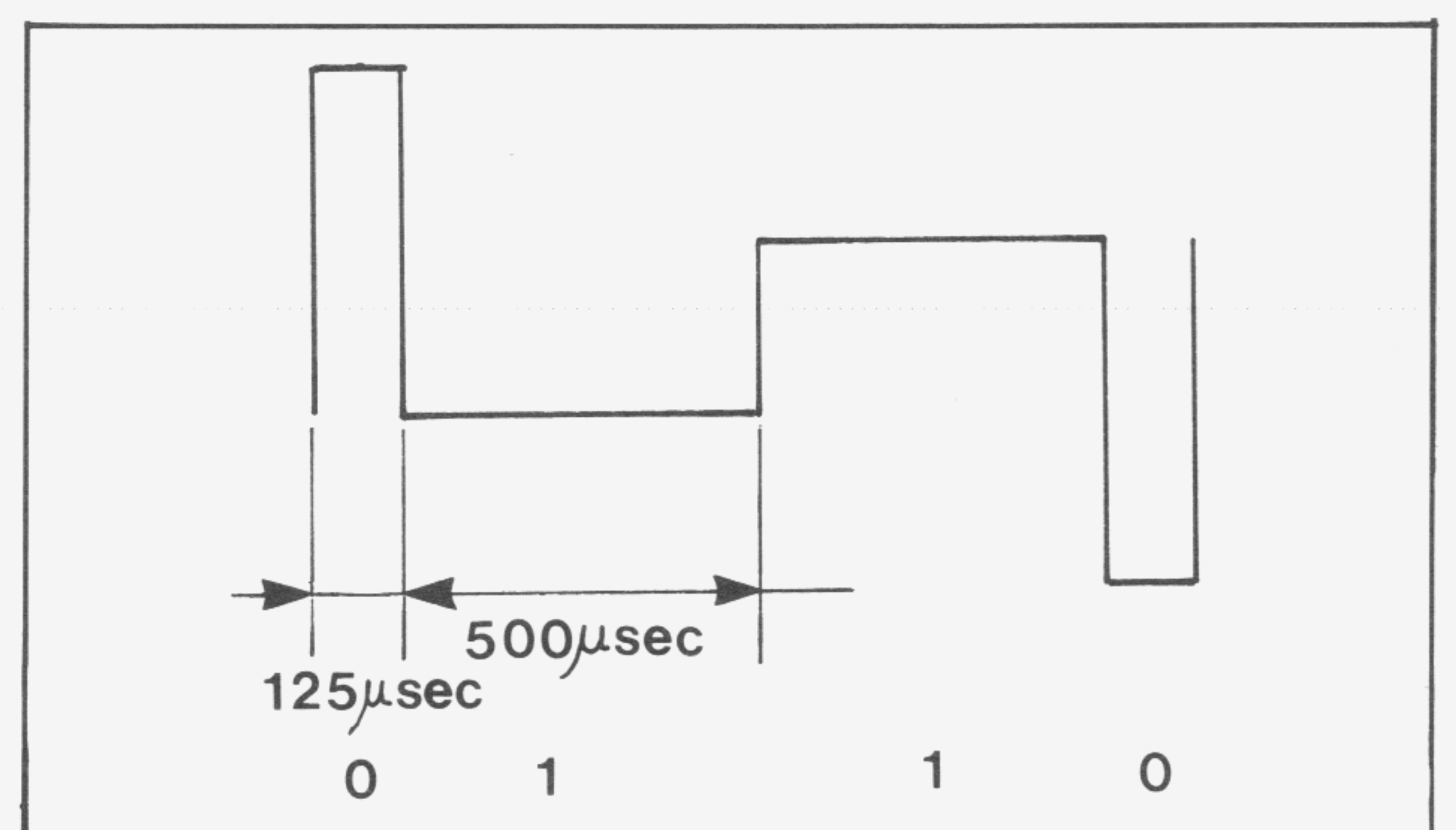
SAVE

The CPU (IC12) converts data from the RAM (IC24) into two kinds of pulses with different widths (0 to 125μs and 1 to 500μs) as shown in the figure.

Accordingly, the average transmitting speed (signalling speed) is calculated as follows:

$$T = \frac{125 + 500 (\mu s)}{2} = 312.5 \mu s$$

$$\text{Thus } \frac{1}{T(312.5)} = 3.2 \text{ k baud}$$



LOAD, VERIFY

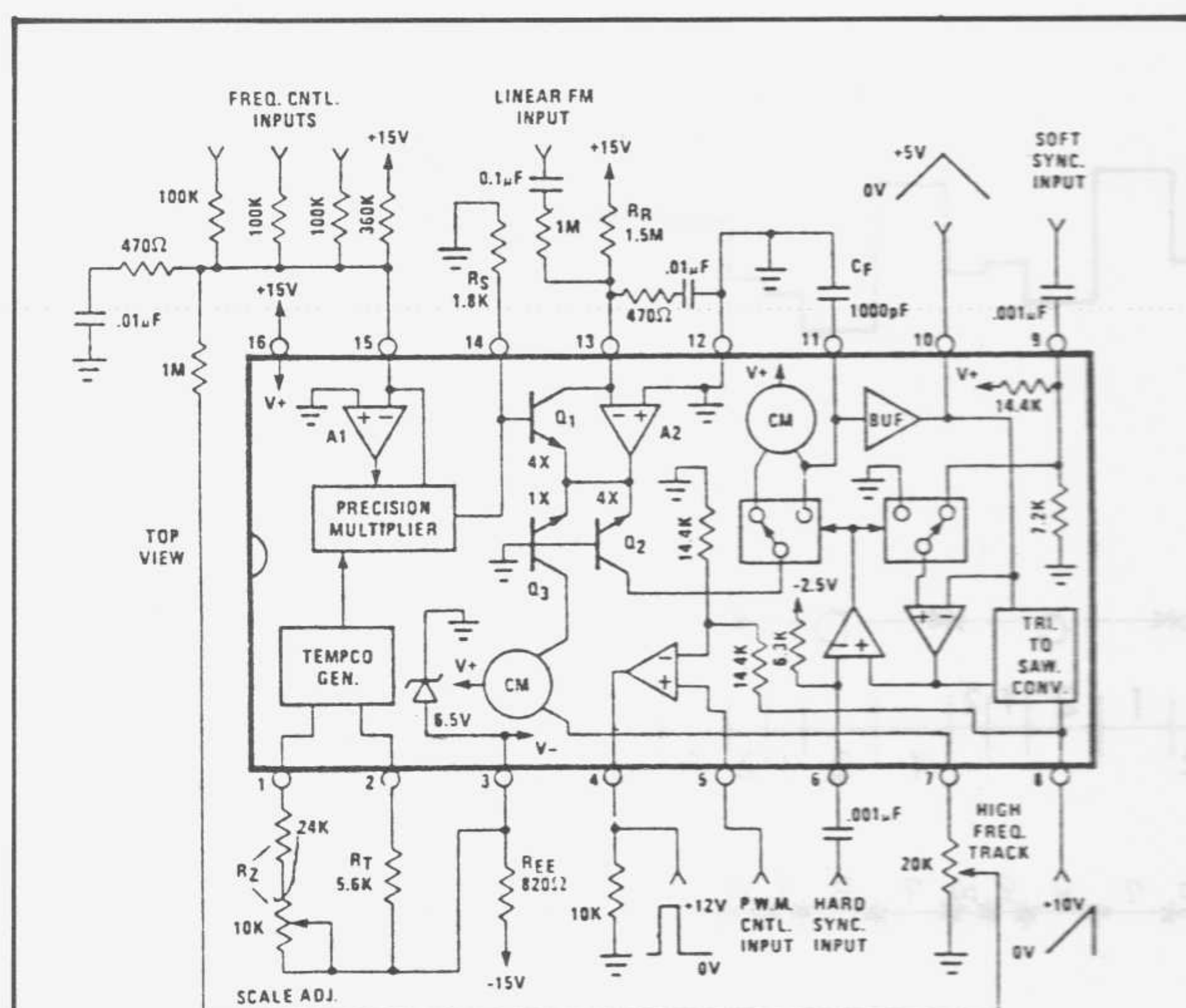
IC4, TR2 and associated circuits shape the input signal from the cassette interface into a pulse wave. IC12 (CPU) reads the shaped waveform through INT 1 and measures the period between waveform edges to determine whether the data is 1 or 0.

When detecting an error by summation check, the program skips the block in which the error exists, lighting an indicator, then loads the next block. If there is no error through loading, the program returns to the normal mode. If an error occurs, error indicator(s) remain lit and the program cannot escape the TAPE mode until the TAPE button is pressed.

MODULE BOARD

VCO

Each VCO (IC33, IC36) is composed of a single chip IC, CEM3340. Three waveforms from the VCO are unequal in amplitude, which is compensated in the next stage (IC34 or IC37) for uniformed levels. Synchronization with the associated VCO is accomplished by external connections, leaving the internal SYNC disabled.



CEM 3340 Circuit Block and Connection Diagram

COMPUTUNE

When the TUNE button is pressed, the sawtooth wave selected among the outputs from the VCOs by IC20 passes through the comparator (IC4) then to CPU (IC6). The CPU measures the frequency of the wave and delivers a corrected CV data for that VCO to D/A converter IC11. The CPU repeats the cycle for the remainder of VCOs.

VCF

VCF is comprised of two series-connected filters of basically the same configuration. Each can function as either LPF or HPF of 12dB/oct slope when its output point is suitably selected.

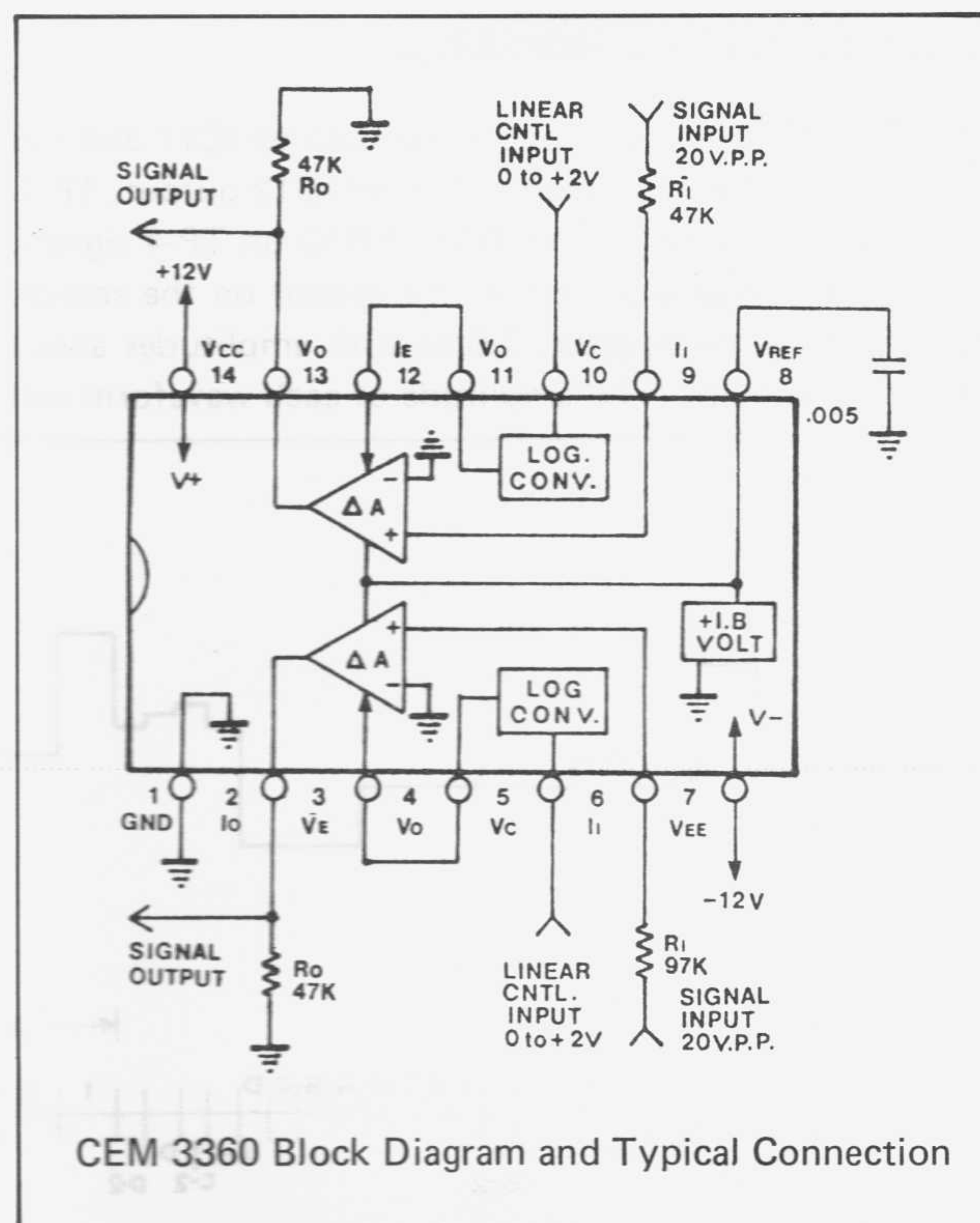
Moreover the VCF will serve as a BPF by configuring one filter into LPF and the other HPF. In the JP-6 the 1st becomes HPF and 2nd LPF when VCF-MODE selectors are in BPF. Slight difference between two stages in circuit diagram illustrates compensation means for level and prevention against peak clips.

VCA

1st VCA

This device functions as a linear VCA accepting control signal through its linear control terminal.

The signal is called ENV-2, a combination of A, D, S, R and K.F data.



CEM 3360 Block Diagram and Typical Connection

2nd VCA

This device is controlled by the control knobs, VCA ENV-2 LEVEL and VCA LFO, and determines the entire output level of the MODULE BOARD.

Block Diagram

