UCRL-19230

ey. 2



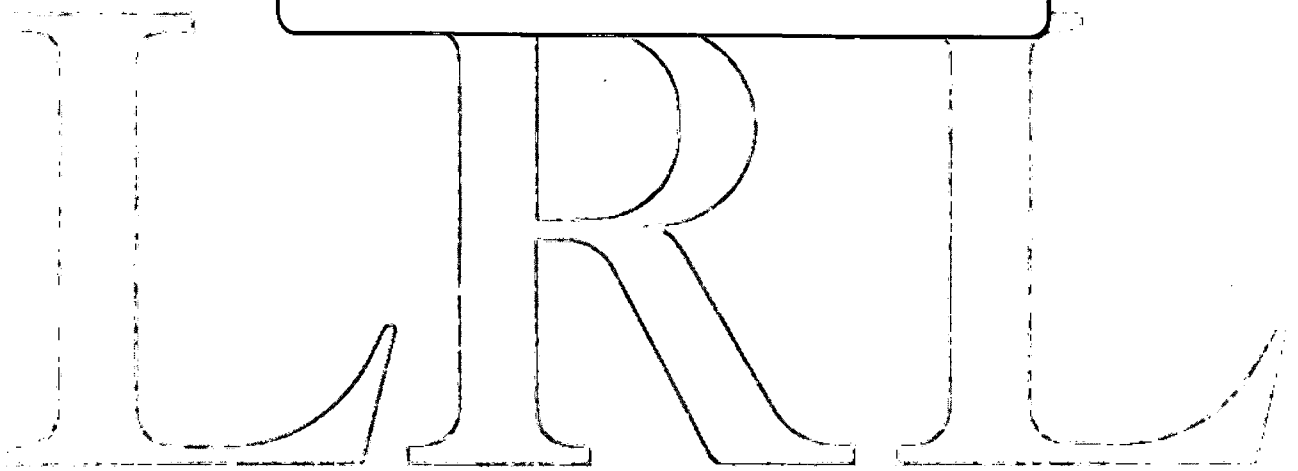
# AN INEXPENSIVE MAGNETIC TAPE FOR SMALL COMPUTER USE

M. A. Lechaczynski and L. B. Robinson

June 1969

AEC Contract No. W-7405-eng-48





LAWRENCE RADIATION LABORATORY
UNIVERSITY of CALIFORNIA BERKELEY

UCRL-19230

ey. 2

## DISCLAIMER

UCRL-19230
UC-32
Mathematics and
Computation
TID-4500 (54th Ed.)

UNIVERSITY OF CALIFORNIA

Lawrence Radiation Laboratory
Berkeley, California

AEC Contract No. W-7405-eng-48

# AN INEXPENSIVE MAGNETIC TAPE FOR SMALL COMPUTER USE

M. A. Lechaczynski and L. B. Robinson

June 1969

Printed in the United States of America
Available from
Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards, U. S. Department of Commerce
Springfield, Virginia 22151
Price: Printed Copy $3.00; Microfiche $0.65

# AN INEXPENSIVE MAGNETIC TAPE FOR SMALL COMPUTER USE

M. A. Lechaczynski† and L. B. Robinson††

Lawrence Radiation Laboratory*
University of California
Berkeley, California

June 1969

## ABSTRACT

The report describes an inexpensive magnetic tape system for a small computer (SCC 650, 4K, 12 bit), using a DECTAPE transport TU55.

This system will also be connected to a larger computer (SCC 660 16K, 24 bit).

---

† Present Address: SPNME Bat 72, CEN SACLAY 91 - Gif/Yvette, France

†† Present Address: Lick Observatory, University of California
Santa Cruz, California 95060

# I. INTRODUCTION

Small computers, often costing less than $15,000, are now used in many engineering and laboratory applications. The low initial cost is often obtained by provision of a minimum amount of input-output equipment. Usually programs must be loaded from punched paper tape, a frustrating and time consuming process. One would prefer magnetic tape, but most tape transports cost as much or more than the small computer.

Inexpensive tape transports have often proved unsatisfactory because of limitation in reliability, incompatibility with all other computers, and inconvenience in use.

This report describes a tape system for a Scientific Control Corporation computer using a Digital Equipment Corporation "DEC TAPE" transport TU55. This transport has the advantage of very high reliability, and also provides a standard format used by several computers at the Lawrence Radiation Laboratory. An additional, very important advantage is that editing and modification of data on these tapes can be done without disturbing adjacent information. Most of the functions commonly served by paper tape can be handled by this magnetic tape system at a price less than the cost of a fast paper tape reader and punch.

A magnetic tape system similar to the one described here will also be connected to a larger computer (SCC 660 - 16K, 24 bit) where it will augment the capabilities of a magnetic disc, for program and data storage.

## II. DEC TAPE Description

Five parallel tracks are written on the tape, three for data, one for timing and one (Mark Track) for control information. Each track is duplicated when writing for a total of 10 tracks and the average of two duplicate tracks is used when reading, for high reliability. The Timing and Mark Tracks allow exact location of data at any point on the tape. A detailed description of DEC TAPE is given in the PDP-9 Instruction Manual "TC02 Dectape Transport Control" published by Digital Equipment Corporation.

FORMAT:

The tape is normally divided into blocks of 256 18-bit data words, the blocks being separated by checksum and location information. The exact type of information at any point is indicated by the bit pattern on the Mark Track. The Mark Track pattern is broken up into 6-bit sequences, and by detecting a particular 6-bit pattern, one can locate a particular part of a data block. Fig. 1 shows the normal sequence of Mark Track patterns. Note that the patterns are oriented to 18-bit computer words (6 x 3 bit characters) but a 12- or 24-bit word length can be written or read from the tape. With the SCC 650, a 12-bit computer, each block contains 384 computer words.

At the end of each block, one word is devoted to a checksum. The checksum is compressed into only the first 6 bits of the word, so again no problem is met when using 12-, 18- or 24-bit words. The "block number"

occurs at the beginning of each block and, since only $(1101)_8$ blocks are written on one tape, a 12-bit word is adequate to contain any possible block number.

Controller Design Philosophy:

This system was designed for relatively low cost, so that it depends heavily on programming for data transfer. (A versatile program for reading and writing tape on the SCC 650 uses $(155)_{10}$ words and is given in Appendix A.) In order to simplify the circuitry, only 6 bits at a time are read from tape. When writing, 12 bits at a time are transferred. To ensure format compatibility, Timing and Mark tracks are not written by the SCC 650, but are pre-recorded by a DEC computer (PDP-7) or pre-formatted tapes can be purchased from DEC.

Description of the Hardware System:

The interface is built in the form of a NIM plug-in unit connected with the TU55 dectape transport by a single cable and with the SCC 650 computer by two cables.

Connection to the TU55 Transport:

Several minor modifications were made on the TU55. These are indicated in Table 1. A special voltage level conversion card has been added in location A06 (fig. 2).

| MODIFICATIONS ON TU55 WIRING (Front Panel) | |
|---|---|
| *1. DISCONNECT | |
| | A1-C to B1-C |
| | A2-C to B2-C |
| | A3-C to B3-C |
| CONNECT | |
| | A3-T to B3-A |
| | A3-U to B3-B |
| | A3-V to B3-C |
| 2. In order to by pass the unit selector: | |
| DISCONNECT | |
| | A07-L from B07-R and T |
| CONNECT | B07-R and T to Ground |
| 3. DISCONNECT | |
| | 100Ω Resistor between A6-K and A09-S |
| CONNECT | A06-K to Ground (A3-C) |
| 4. In order to provide -15V to level changer A6 | |
| CONNECT | A6-B to A4-B |

TABLE 1.

* This modification is not needed if a double-sized flip chip is used to connect the heads with the amplifiers in the interface.

Connection With the SCC 650 Computer:

A 50-wire cable connects the output signals from the computer and another 50-wire cable connects the inputs to the computer.

Controller Cards:

The interface is built of seven 86-pin cards, PC1 to PC7. The dimensions were chosen to mount conveniently in a NIM module (11X1751 P-1).

| | | |
|---|---|---|
| Cards PC1 and PC3 | triple amplifier | Figs. 3,4 (11X6500 S-1,11X6631 P-1) |
| Card PC5 | mark track decoding | Fig. 5 (11X7201 P-1) |
| Card PC7 | read logic | Fig. 6 (11X7211 P-1) |
| Card PC2 | 12-bit write buffer | Fig. 7 (11X7161 P-1) |
| Card PC4 | write logic | Fig. 8 (11X7141 P-1) |
| Card PC6 | write drivers | Fig. 9 (11X7131 P-1) |

## III. TAPE MOTION CONTROL
(See Block Diagrams Fig. 10, 11)

Two flip-flops select "STOP" or "GO" and "FORWARD" or "REVERSE". A 300 ms START DELAY inhibits reading and writing until the tape is up to speed. These functions are set by bits 09, 10 and 11 transmitted from the A register when the instruction EXU 14 is executed.

Bit 11 = 1 : GO  Bit 11 = 0 : STOP

Bit 10 = 1 : REVERSE  Bit 10 = 0 : FORWARD

Bit 09 = 1 set motion delay (Motion Command)

Special codes on the end of the tape will stop the transport after a delay of 300 ms. Pressing "START" button on the SCC 650 will also stop the tape motion.

## IV. READ CIRCUITRY

### 1. Read Amplifiers (PC1 to PC3):

On PC1, two amplifiers are used for the Timing and Mark Tracks. On PC3, three amplifiers are used for the three dual data tracks.

The capacitors at the input of the amplifiers reduce the effect of high-frequency noise generated by the drive motors.

The gain of the amplifiers is such that the 15 mV signals from the tape will produce a 6-volt square wave on the output.

### 2. Mark Track Decoding (PC5):

The mark track is fed into a 6-bit shift register. The six outputs are tested during each timing interval which is obtained from the output of the timing amplifier. In the middle of each timing interval one bit from the Mark Track is shifted into the shift register.

Octal codes 51 : guard mark
45 : reverse block mark
25 : interblock sync. mark
26 : forward block mark
32 : reverse guard mark
22 : End of Tape.

are decoded and generate pulses when present.

To shape the timing waveform it is necessary to eliminate perturbations which follow each change of current during the writing. The circuit Q1, Q2 of fig. 5 prevents any transients in the timing waveform for about 6 μs after the beginning of each timing interval.

When the tape motion starts, the START DELAY inhibits all of these codes for a period of 300 ms to be sure that the tape is moving at full speed.

3. Read Logic (Card PC7):

a. Synchronization (Fig. 12):

The trailing edge of first Code 25 encountered after the end of the START DELAY sets the SYNC FF, which has been previously reset at the beginning of the START DELAY. This then enables the generation of READ and $\overline{\text{READ}}$ signals, which transfer data into the 6-bit read register; at each time "$t_1$" the 3 bits in the read register feeding to "A" register bits 6, 7 and 8 are loaded. At each time $t_2$, the next 3 bits are loaded into bits feeding "A" register bits 9, 10 and 11. At this time 6 bits of data are ready to be transferred into the A register.

b. Transfer of 6 bits by TTA:

Each timing pulse $t_2$ sets flip-flop "ONE TTA" which allows the next TTA 14 generated from the computer to generate an IDRDY which enters the 6 bits into the A register and inhibits the skip.

The trailing edge of the IDRDY resets the "ONE TTA" flip-flip, which will be set again by the next $t_2$, when the next 6 bits will be ready to be transferred.

CAUTION: Data must be transferred by TTA within $\approx 33$ μs after $t_2$, otherwise they will be lost.

The rate of reading is about 6 bits per 66 μs.

c. Block Number Search:

The first thing to do when reading or writing is to find a specific location on the magnetic tape, by finding a certain block number. This is done by finding code 26 which indicates that the preceding 12 data bits represent not data but the number of a block. Code 26 is detected by the computer input/output instruction SDF 14. If the preceding 12 bits correspond to the block desired, reading or writing may be started. Since block numbers are prewritten on the tape in both forward and reverse directions, block numbers can be read going forward or reverse. Normally one searches reverse until one finds a block number smaller than that desired then searches forward for the exact block so reading and writing are always done in the forward direction.

d. Finding Block Number Zero:

To read or write most blocks, one searches reverse for a block having a number 2 or 3 less than the block desired, so that when the tape turns around to read forward it will certainly

be up to speed before passing the beginning edge of the desired block. This procedure is obviously inappropriate for blocks 0, 1, 2. A STOP DELAY and a stop indicator to the computer are necessary to solve this problem.

As the end of tape is appraoched moving in reverse, block 0 is passed and then "end of tape" code 22 triggers the stop delay of about 300 ms. At the end of the stop delay the GO-STOP flip-flop is set to STOP. When the flip-flop is set to STOP, the TTA 14 response is enabled continually, with the most significant bit transmitted to the "A" register as 0 instead of the normal 1. The program detects the end of tape by simply testing this bit (the sign bit), and if it is zero (Positive "A" register) a read-forward command is given. In about 100 ms the tape will be stopped and started moving forward again. Since the "START" delay lasts for about 300 ms also, the read circuits are enabled about 200 ms before block 0 is reached moving forward.

## V. WRITING CIRCUITRY (Fig. 11)

Introduction:

The writing is done with the aid of Instruction TFA 14 which transmits 12 bits at a time from the "A" register to the controller buffer. These 12 bits are then written as four successive lines of 3 bits each on the tape.

a. Synchronization of Timing (Fig. 13 and Fig. 1):

The sequence TA, TB, TC, TD controls the write timing during each block. The sequence generator is synchronized by code 32 at the beginning of each block. The first TA commences on the leading edge of the timing pulse that occurs during code 32. Each block of the tape is thus divided into W1, W2, ... to W391, including the 384 12-bit data words and several checksum and guard words, (W1 : Word 1).

b. TFA and Data Writing:

During a sequence TA, TB, TC, TD (around 130 μs) only one TFA can be executed, that is to say only 12 bits can be transferred from the "A" register to the controller.

The leading edge of each TA sets the flip-flop "ONE TFA" which permits the first TFA 14 from the computer to transfer data. The trailing edge of TFA 14 resets the flip-flop which can only be set again by the next TA. The leading edge of this next TA transfers the 12 data bits to the EXEC register, which feeds successive 3-bit groups of data at a time to the writing drivers during times TA, TB, TC, TD.

c. Manchester Method:

Dec tape is written using the Manchester method (opposite polarity for ONE and ZERO). For this reason during each interval TA, TB, TC, TD, if a bit 1 is to be written, one synchronizes the

writing drivers with the timing signal, that is to say positive during the first half and negative during the second half of one time cycle. To write a bit ZERO the opposite polarity is used.

d. Write Enable:

Several conditions are necessary to enable writing:

1. Manual write enable switch on transport TU55.
2. "DO WRITE" flip-flop must be set by EXU 14 instruction with bit 8 set in the "A" register.
3. Writing is permitted only when the tape is going forward.
4. Writing is prohibited in the zone containing the pre-written block numbers.
5. Finally writing is only permitted during a 12-bit word timing cycle $W_n$ if a TFA 14 has been executed during the preceding cycle $W_{n-1}$. The timing diagram for the WREN-TFA is shown on fig. 14 and fig. 1.

# APPENDIX

## SOFTWARE NOTES - SCC 650 DECTAPE

A basic set of subroutines that read and write any number of words starting at any block on a Dectape and provide PDP-9 and PDP-7 compatible checksums is reproduced on the following pages. Several points in the listing are marked and special difficulties are discussed below.

1. The block-find routine spaces 7 or 9 six-bit data characters, after detecting a block mark. This synchronizes the start of data transfer with the first data location in the block.

2. Commands are transmitted by EXU'14 instruction.

   code 1 - move forward

   code 3 - move reverse

   code 4 - trigger start-up delay (may be combined)

   code 10 - enable writing (may be combined)

3. When reading, only 6 bits are transmitted at a time, so two TTA 14 commands are needed for each 12-bit word. When writing, 12 bits at a time are transmitted by a TFA 14 command.

4. Each block passes the head in about 50 ms. By searching for a block number 3 smaller than that desired, we allow 150 ms extra time for turn around.

5. If the block is not found after 7 or 9 tries for reading or writing respectively, the routine exits with: $(7740)_8$ in the "A" register.

6. In order to get a correct checksum, a partial block is filled out with the final word.

7. Only the most significant 6 bits are non-zero in a PDP-9 checksum.

8. The most significant bit of the "A" register always receives a one during TTA 14 unless the tape is halted. Then the bit receives zero, and TTA 14 will always be answered by the controller.

9. "CKSUM" is used as a counter and ends up preset to zero here.

10. The number of the first block to be transferred is stored in location 5. The address of the first word is in location 2. The negative word count is in location 3.

11. If writing is disabled, TFA does not respond, and after 4096 tries, routine exits with $(7760)_8$ in the "A" register.

12. The XOR of the calculated and stored checksums should be zero. The actual value is stored in DTESTE and placed in the "A" register when the read routine exits.

```
                *SUBROUTINE TO READ AND WRITE DECTAPE
                *CALL DTWRIT  OR   DTREAD
                *
                *AC=0 FOR CORRECT RETURN
                *A=40 FOR NOT FOUND. A='60 FOR WRITE UNRESPONSIVE
                *
                *WRITE SUBROUTINE
                *
                        NAME    DTWRIT,DTREAD,DTFIND,DRBLOK
     00000      DTWRIT  BSS     2
00002 6113              LDA     DWRITR
00003 3755              STA*    DOPACR          SET WRITE COMMAND
00004 3271      (1)     LDL     -7
00005 5157              JSL     DTFIND          FIND REQUIRED BLOCK
00006 0301              SRA     0,Z
00007 2117              JMP     DWEXIT          NO RESPONSE
00010 3211      (2)     LDL     '11             WRITE FORWARD
00011 0614              EXU     '14
00012 0003      DWWRD   CLA
00013 3746              STA*    DTESTR
00014 6343              LDA*    DPOINT          WRITE ONE WORD
00015 0454      DWRITR  TFA     '14
00016 2105              JMP     DWCONT
00017 4342              MIN*    DTESTR
00020 2503              JMP     DWRITR
00021 3260      (11)    LDL     '60
00022 2104              JMP     DWEXIT
00023 5337      DWCONT  JSL*    DTOPRR          SAVE CHECKSUM ETC.
00024 0301              SRA     0,Z             TEST END OF DATA
00025 2513              JMP     DWWRD
00026 0614      DWEXIT  EXU     '14             STOP TAPE
00027 7127              JRT     DTWRIT          EXIT
                *
                *
                *
                *
                *READ SUBROUTINE
                *
     00030      DTREAD  BSS     2
00032 3267      (1)     LDL     -9
00033 5131              JSL     DTFIND
00034 0301              SRA     0,Z
00035 7105              JRT     DTREAD          NOT FOUND
00036 6102              LDA     DTRGO
00037 3721              STA*    DOPACR          SET READ CHECKSUM
00040 0414      DTRGO   TTA     '14
00041 2102      (3)     JMP     *+2
00042 2502              JMP     *-2
00043 0020              XHA
00044 3517              STA     DTEMPQ
00045 0414              TTA     '14
00046 2102      (3)     JMP     *+2
00047 2502              JMP     *-2
00050 7513              AND     DTEMPQ
00051 3706              STA*    DPOINT
00052 5310              JSL*    DTOPRR
```

```
PAGE 001
00053 0301          SRA     0,Z             ZERO FOR END OF DATA
00054 2514          JMP     DTRGO
00055 6304    (12)  LDA*    DTESTR
00056 7126          JRT     DTREAD
                    *
00057 0000    DPOINT PAR    0
00060 0146    DOPACR PAR    DOPACT
00061 0164    DTESTR PAR    DTESTE    (12)
00062 0123    DTOPRR PAR    DTOPER
00063 0000    DTEMPQ PAR    0
                    *FIND DECTAPE BLOCK WHOSE NO. IS IN ADDRESS 5
                    *EXITS MOVING FORWARD IN CORRECT BLOCK
                    *
     00064    DTFIND BSS    2
00066 3734          STA*    DTSKPR          -7 TO WRITE;-9 TO READ
00067 3575          STA     DTESTE
00070 3207    DTBACK LDL    7               READ REVERSE
00071 0614          EXU     '14
00072 5176    DTWATB JSL    DRBLOK          READ BLOCK NO.
00073 3303     (4)  ADL     3               FIND BLOCK NO, 3 SMALLER
00074 0201          SRA     0,N
00075 2503          JMP     DTWATB          TOO BIG
00076 3205    DTFORW LDL    5
00077 0614          EXU     '14             READ FORWARD
00100 5170    DTWATF JSL    DRBLOK          GET BLOCK DIF.
00101 0101          SRA     0,G
00102 2106          JMP     DFPLUS
00103 4161     (5)  MIN     DTESTE          REVERSION COUNTER
00104 2514          JMP     DTBACK          MISSED IT
00105 3240          LDL     '40
00106 0614          EXU     '14             STOP
00107 2112          JMP     DFEXIT
00110 0301    DFPLUS SRA    0,Z             IS THIS IT?
00111 2511          JMP     DTWATF
00112 3552          STA     DTESTE          CLEAR ERROR TEST
00113 4153          MIN     DTEST
00114 6002    (10)  LDA     2               FIRST CORE ADDRESS
00115 3752          STA*    DPOINR          DPOINT
00116 6003    (10)  LDA     3               NEG WORD COUNT
00117 3546          STA     DTCONR
00120 0003          CLA     GOOD EXIT
00121 7135    DFEXIT JRT    DTFIND          EXIT
00122 0227    DTSKPR PAR    DTSKIP
                    *
                    *
                    *DO CHECK SUM ETC FOR ONE WORD
     00123    DTOPER BSS    2
00125 5702          XOR*    CKSUMR
00126 3701          STA*    CKSUMR
00127 0226    CKSUMR PAR    CKSUM
00130 6136          LDA     DTEST
00131 0303     (6)  SRAC    0,Z             DONT ADVANCE IF DATA DONE
00132 4335          MIN*    DPOINR          ADVANCE DATA POINTER
00133 4132          MIN     DTCONR
00134 2102          JMP     *+2
```

```
PAGE 002
00135 3531          STA     DTEST           END OF DATA
00136 3201          LDL     1
00137 4160          MIN     DCOUNT
00140 7115          JRT     DTOPER
00141 6165          LDA     CKSUM
00142 0020          XHA                     END OF BLOCK
00143 5563          XOR     CKSUM
00144 7565    (7)   AND     P7700
00145 3561          STA     CKSUM
00146 0000  DOPACT  PAR     0               READ OR WRITE CKSUM
00147 2102          JMP     *+2
00150 2502          JMP     *-2
00151 0020          XHA
00152 5554          XOR     CKSUM
00153 7556          AND     P7700
00154 4510          ADD     DTESTE          SHOULD BE ZERO
00155 3507    (12)  STA     DTESTE
00156 5301          JSL*    *+1
00157 0170          PAR     DRBLOK
00160 6106          LDA     DTEST           ALL DONE?
00161 0101          SRA     0,G
00162 0614          EXU     '14             0=DATA DONE;STOP TAPE
00163 7140          JRT     DTOPER
            *
00164 0000  DTESTE  PAR     0
00165 0000  DTCONR  PAR     0
00166 0000  DTEST   PAR     0
00167 0057  DPOINR  PAR     DPOINT
            *
            *
            *WAITS FOR NEW BLOCK,BYPASSES REVERSE CHECKSUM
     00170  DRBLOK  BSS     2
00172 6140  TSX     LDA     TEM1
00173 0020          XHA
00174 3537          STA     TEM2
00175 0414  TS1     TTA     '14
00176 2102          JMP     *+2
00177 2502          JMP     TS1
00200 0201          SRA     0,N
00201 2122    (8)   JMP     ENDTAP          BIT 0=0 FOR TAPE STOPPED
00202 3530          STA     TEM1
00203 7530          AND     TEM2
00204 0554          SDF     '14             TEST IF BLOCK NO.
00205 2102          JMP     *+2
00206 2514          JMP     TSX
00207 0240  FOUND   CAX
00210 6117          LDA     DTSKIP
00211 3515          STA     CKSUM
00212 0414          TTA     '14
00213 4113    (9)   MIN     CKSUM           BYPASS REVERSE SUM
00214 2502          JMP     *-2
00215 6113          LDA     M600
00216 3501          STA     *+1
00217 0000  DCOUNT  PAR     0
00220 0140          CXA
```

```
PAGE 003
00221 6405      (10)    SUB     5
00222 7132              JRT     DRBLOK          BLOCK DIF. IN A;BLOCK IN X
00223 6101      ENDTAP  LDA     *+1
00224 6000              PAR     '6000           FORCE FORWARD IF HALTED
00225 7135              JRT     DRBLOK          FORCE TURNAROUND IF STOPPED
                *
00226 0000      CKSUM   PAR     0
00227 0000      DTSKIP  PAR     0
00230 7200      M600    PAR     '-600
00231 7700      P7700   PAR     '7700
00232 0000      TEM1    PAR     0
00233 0000      TEM2    PAR     0
                *
     00000              END
0000 ERRORS IN PASS  2
UNDEFINED
UNREFERENCED
DTFORW
FOUND
PASS 3
```

## ACKNOWLEDGEMENTS

This work was done as part of the program of the Nuclear Chemistry Instrumentation Group. The assistance and co-operation of Blair Jarrett, Fong Fin and Leo Schifferle is gratefully acknowledged.

## FOOTNOTE AND REFERENCES

* This work was carried out as part of the research program of the Nuclear Chemistry Division of the Lawrence Radiation Laboratory, University of California, which is supported by U. S. Atomic Energy Commission Contract W-7405-eng-48.

1. Dectape Transport TU55 (H-TU55).*
2. TC02 Dectape Transport Control.*
3. SCC 650 Interface description (Scientific Control Corporation, Dallas, Texas).

---

* Digital Equipment Corporation, Maynard, Massachusetts

## FIGURE CAPTIONS

Fig. 1. Dectape format, Write and WREN-TFA timing.

Fig. 2. Micrologic to DEC converter card.

Fig. 3. Read Amplifier.

Fig. 4. Triple Amplifier card (PC1 and PC3).

Fig. 5. Mark track decoding card (PC5).

Fig. 6. Read logic card (PC7).

Fig. 7. 12-bit write buffer.

Fig. 8. Write logic card (PC4).

Fig. 9. Write drivers (PC6).

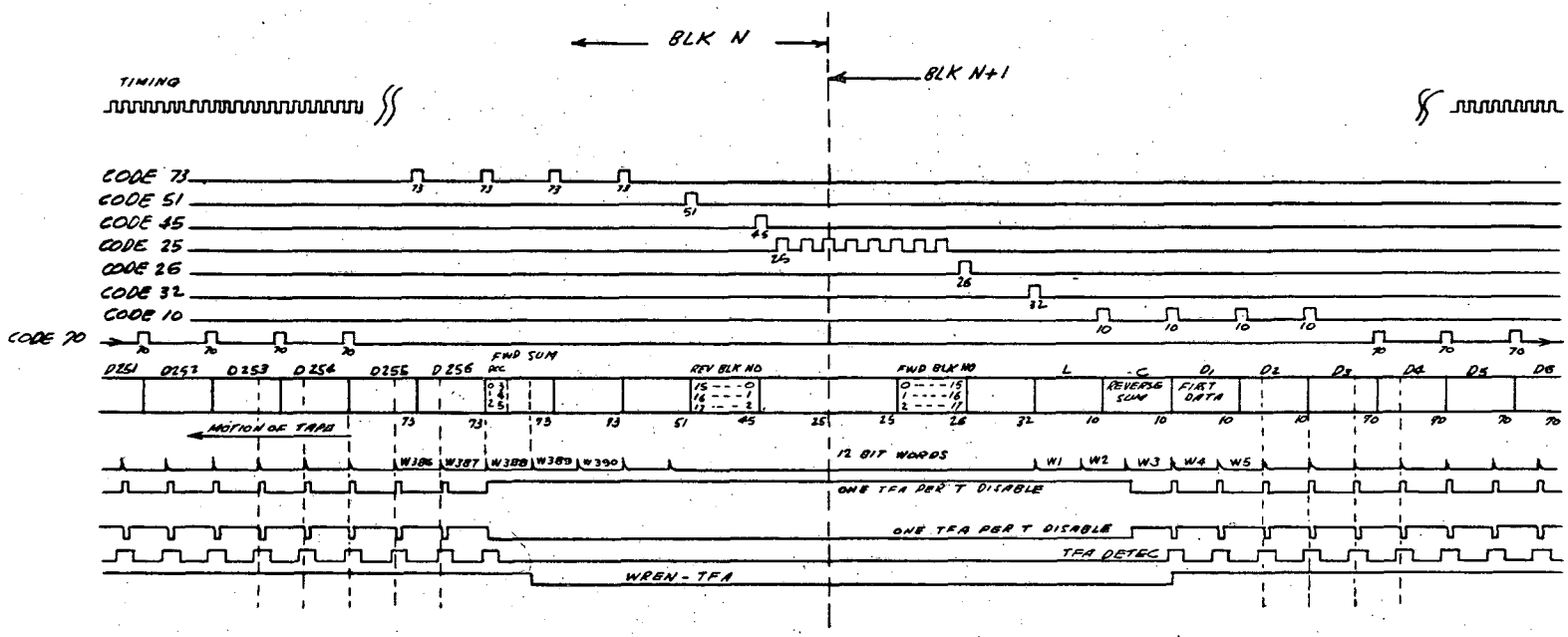
Fig. 10. Read logic block diagram.

Fig. 11. Write logic block diagram.

Fig. 12. Reading synchronization and timing.

Fig. 13. Writing synchronization and timing.

Fig. 14. WREN-TFA timing.




XBL 696-722

**Fig. 1. Dectape format, Write and WREN-TFA timing**

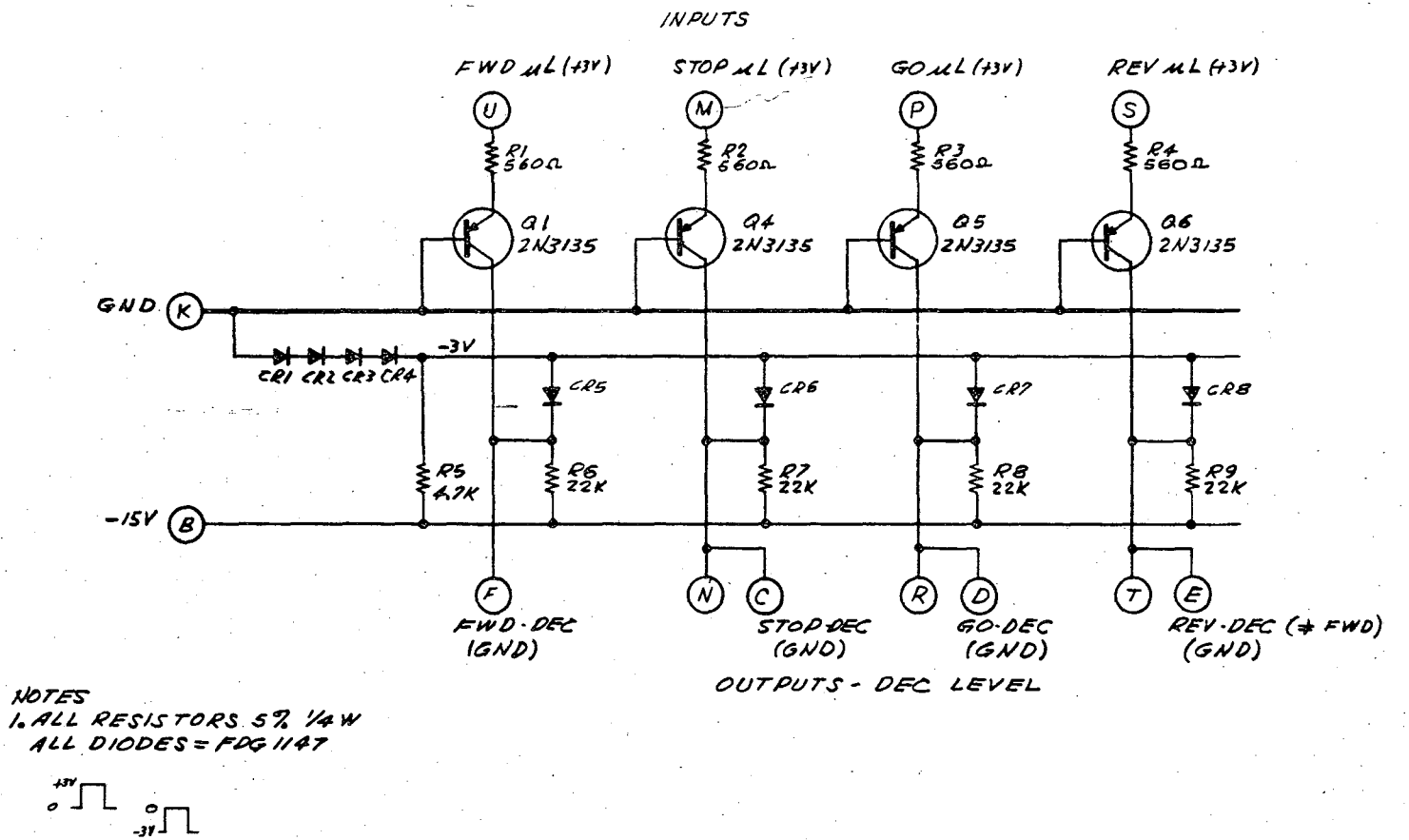
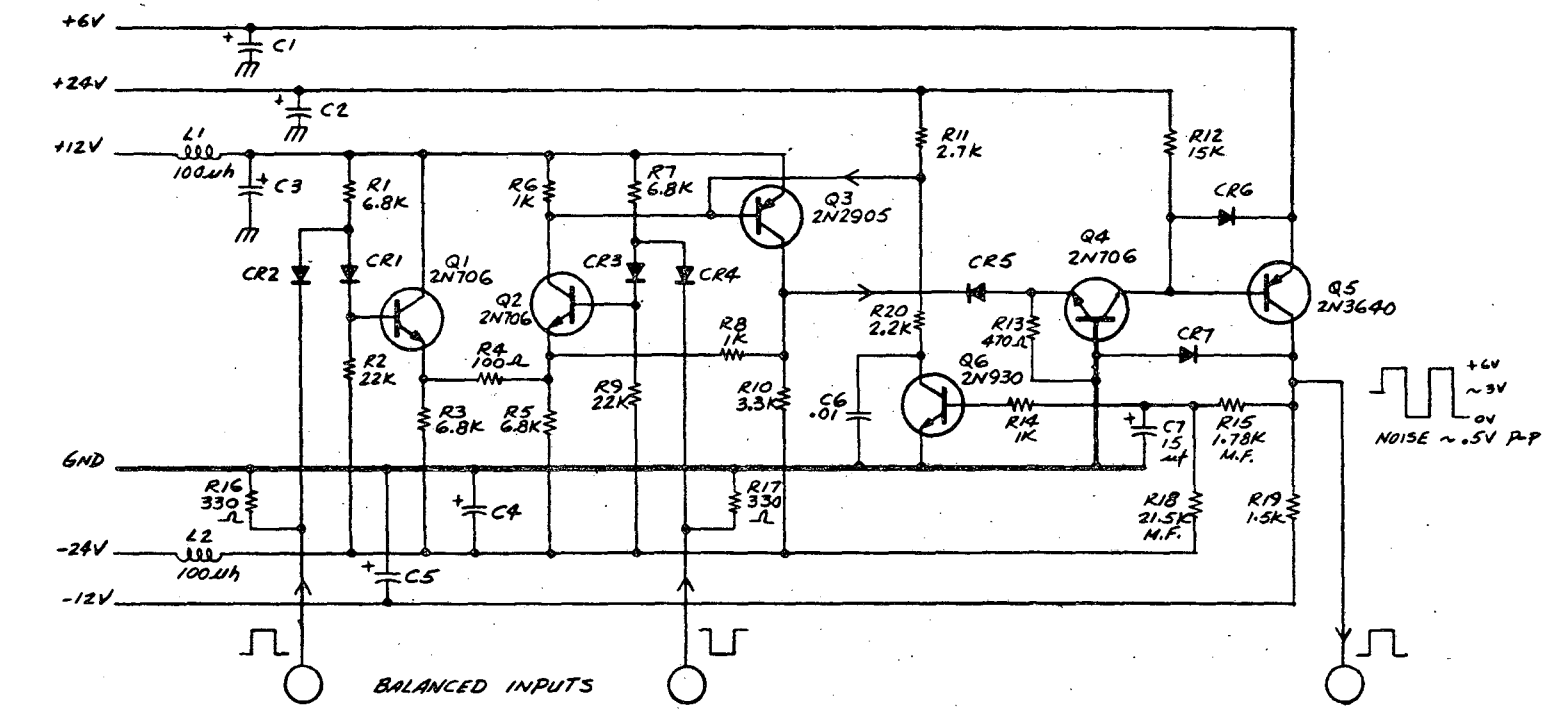



Fig. 2. Micrologic to DEC converter card.

XBL 696-723



UCRL-19230

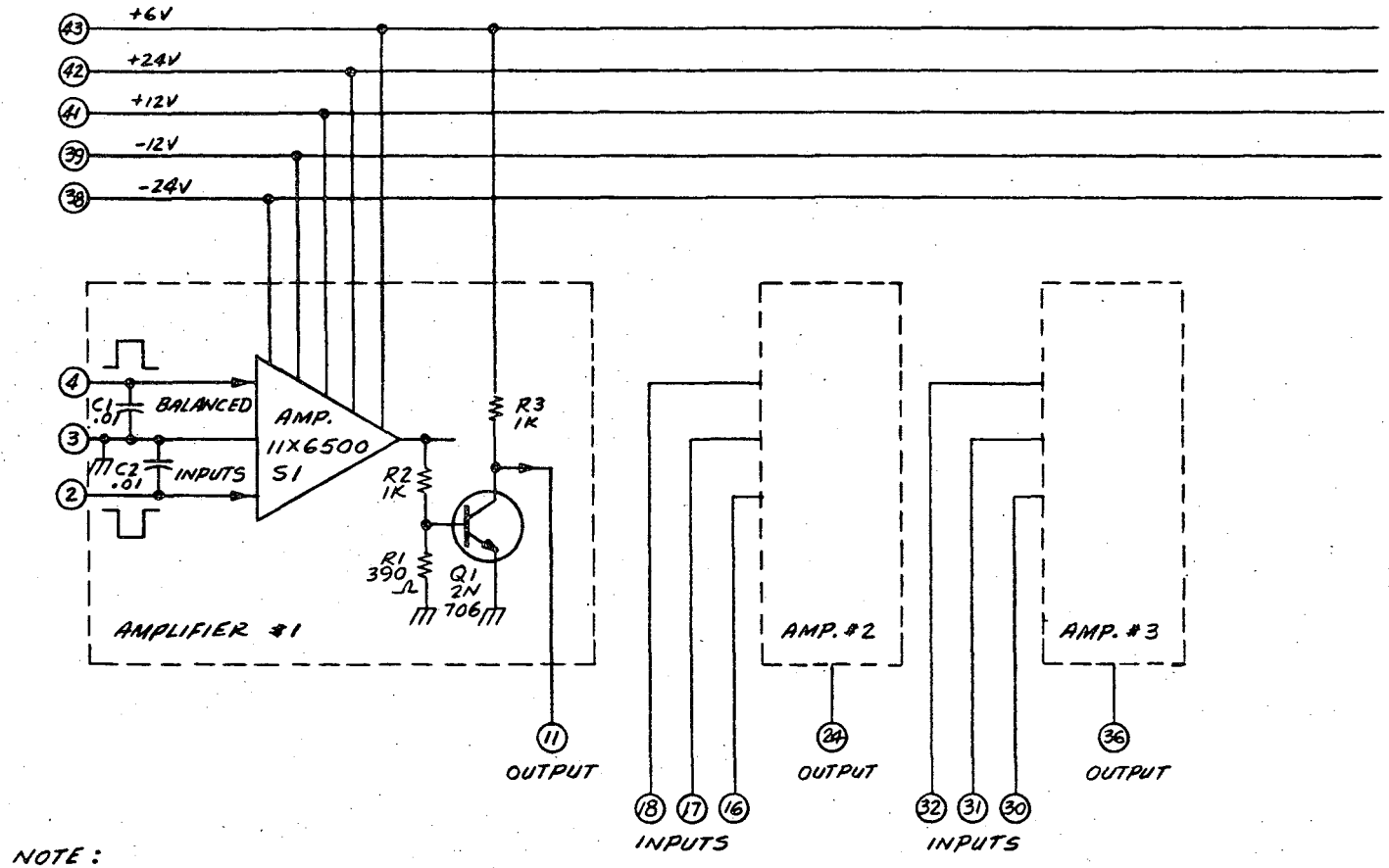



NOTES: UNLESS OTHERWISE NOTED

1. CAP. 6.8μf 35V TANT.
2. DIODES IN914 / OR EQUIV.
3. 2N706 / LED 9623
4. RESISTORS 1/4W 5% CARBON
5. Q6 PROVIDES DC LEVEL CONTROL
6. BASE LEADS OF Q3, Q2, Q1 SHOULD BE KEPT SHORT.
7. EMITTER LEADS OF Q1, Q2, SHOULD BE SHORT
8. CAP. LOAD AT COLLECTOR OF Q5 MUST BE AVOIDED.
9. GAIN ~ 5000
10. REDUCE GAIN BY REDUCING R8.

XBL 696-713

Fig. 3. Read amplifier.



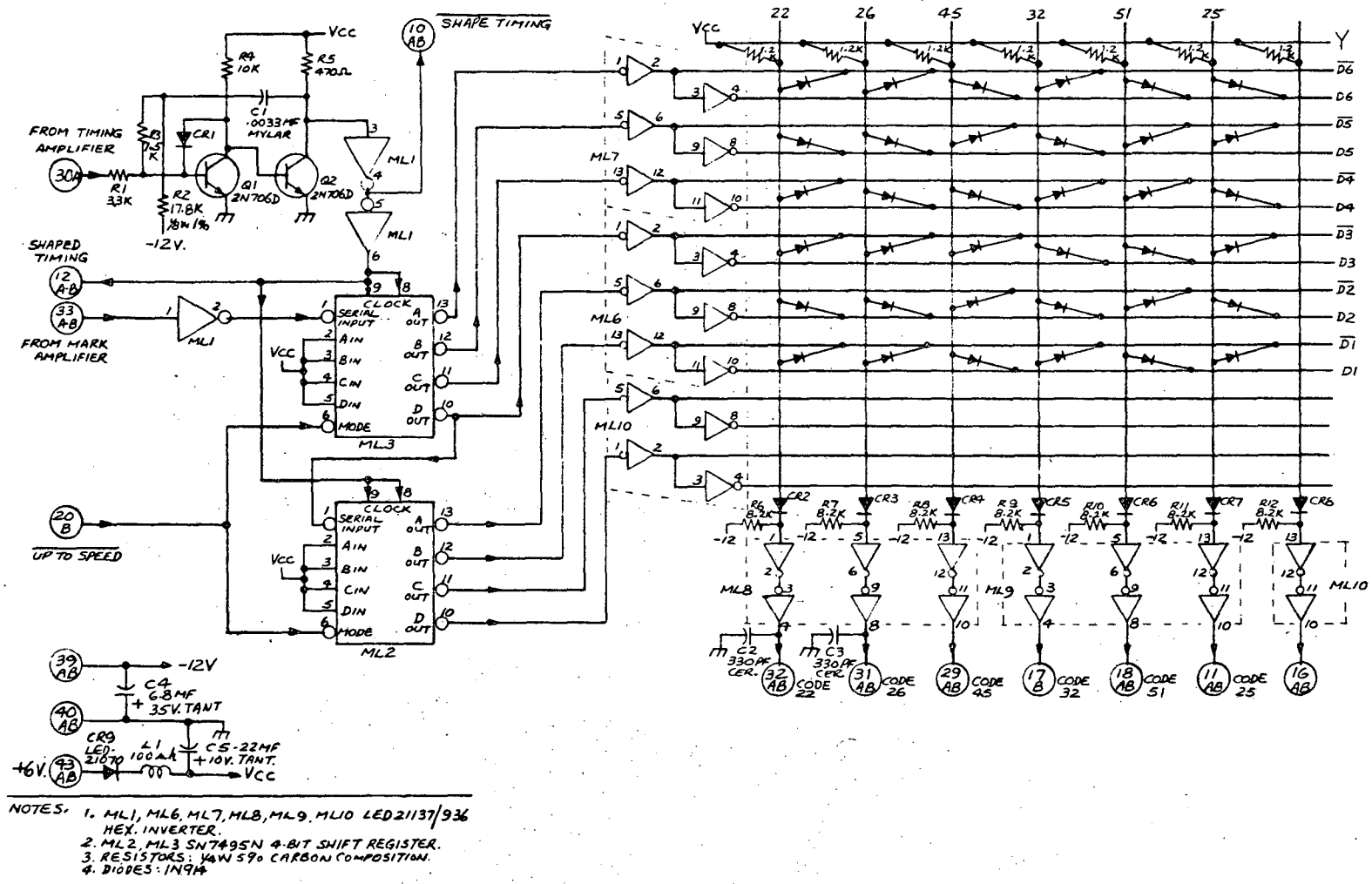

NOTE:
1. RESISTORS ¼W 5% CARBON
2. CAP. CER. DISC.
3. AMP. #1, 2, 3 ARE IDENTICAL
4. INPUTS AND OUTPUTS MUST BE WELL SEPARATED

XBL 696-717

Fig. 4. Triple amplifier card (PC1 and PC3).

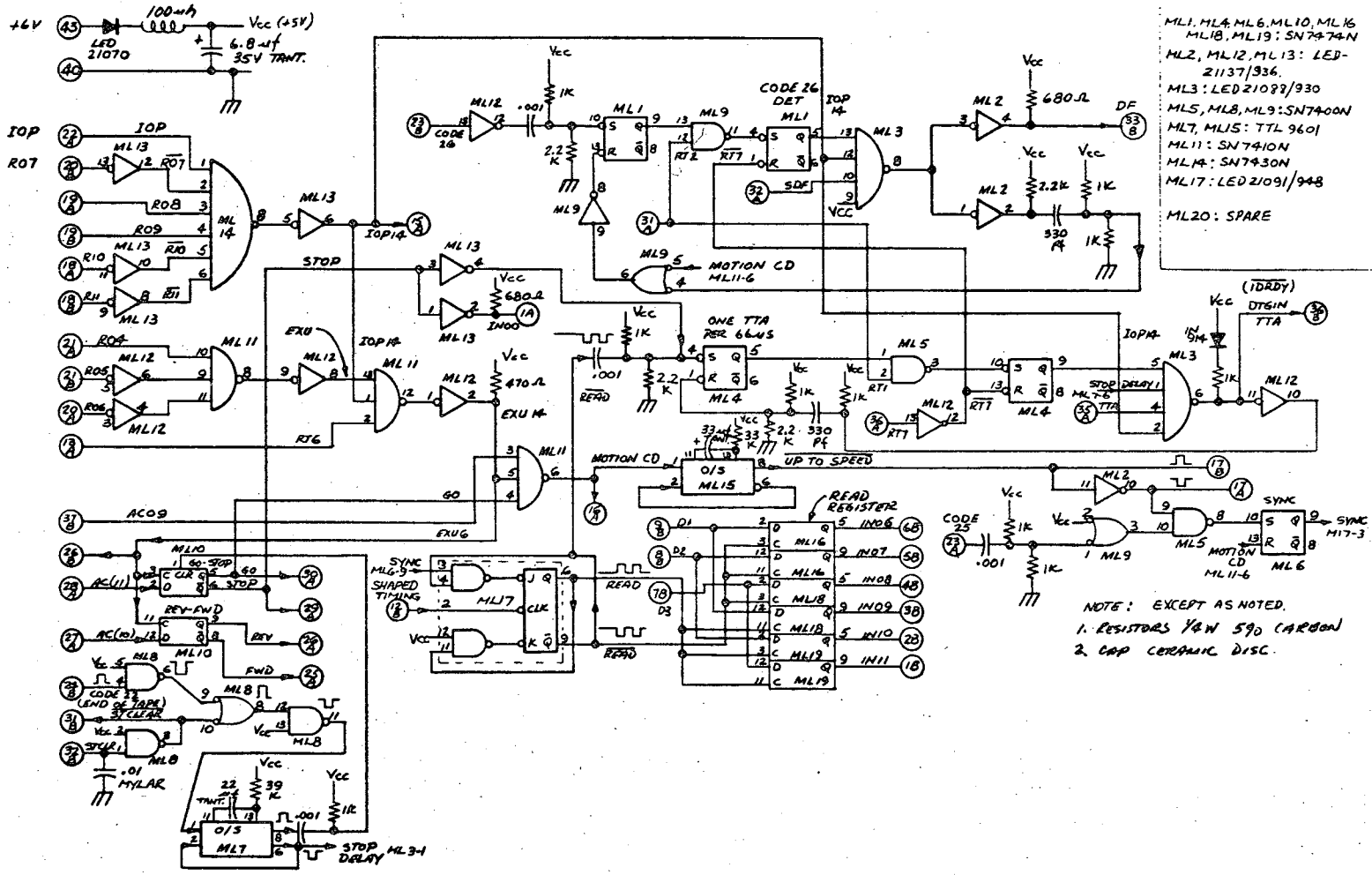


UCRL-19230




XBL 696-719

Fig. 5. Mark track decoding card (PC5).

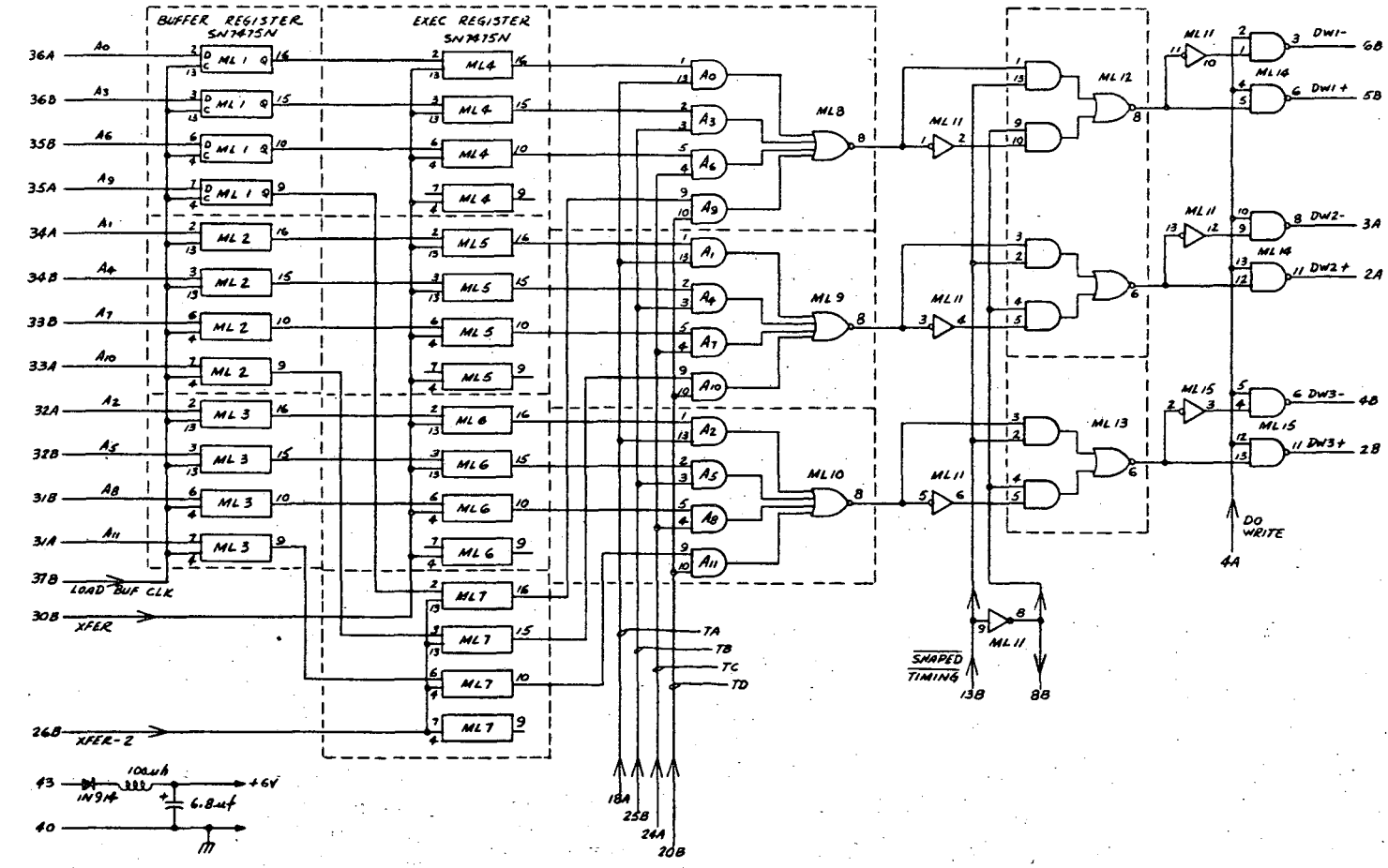



XBL 696-715

Fig. 6. Read logic card (PC7).



UCRL-19230



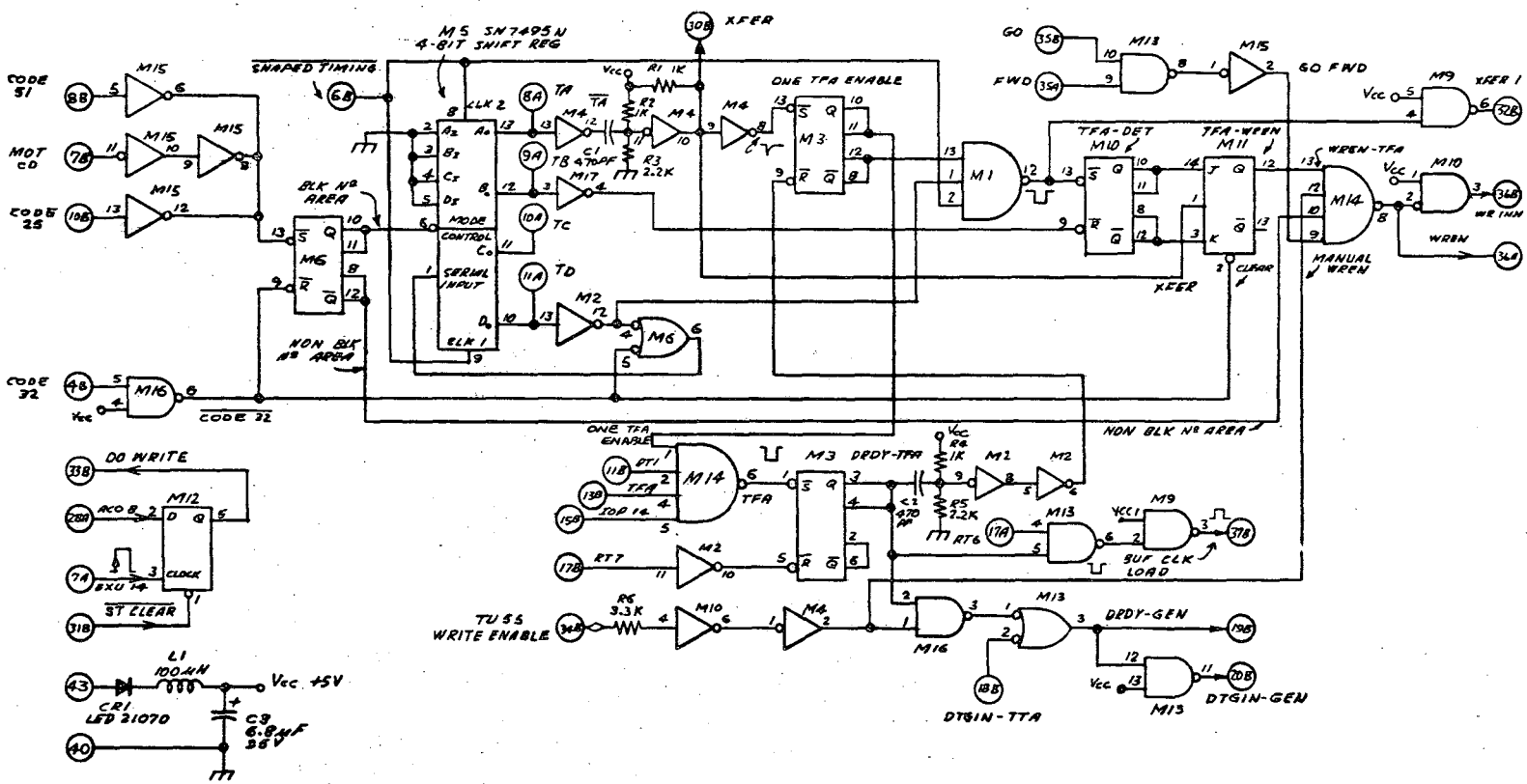

NOTE:
1. ML1-7 SN7475N (16 PINS) PIN 5 Vcc, PIN 12 GND
2. ML 8-10 SN7453N (14 PINS) PIN 14 Vcc, PIN 7 GND
3. ML 12, 13 SN7450N " "
4. ML 11 936 HEX INVERTER " "
5. ML 14, 15 SN7400N " "
6. A = COMPONENT SIDE
   B = WIRING SIDE

XBL 696-725

Fig. 7. 12-bit write buffer.



UCRL-19230

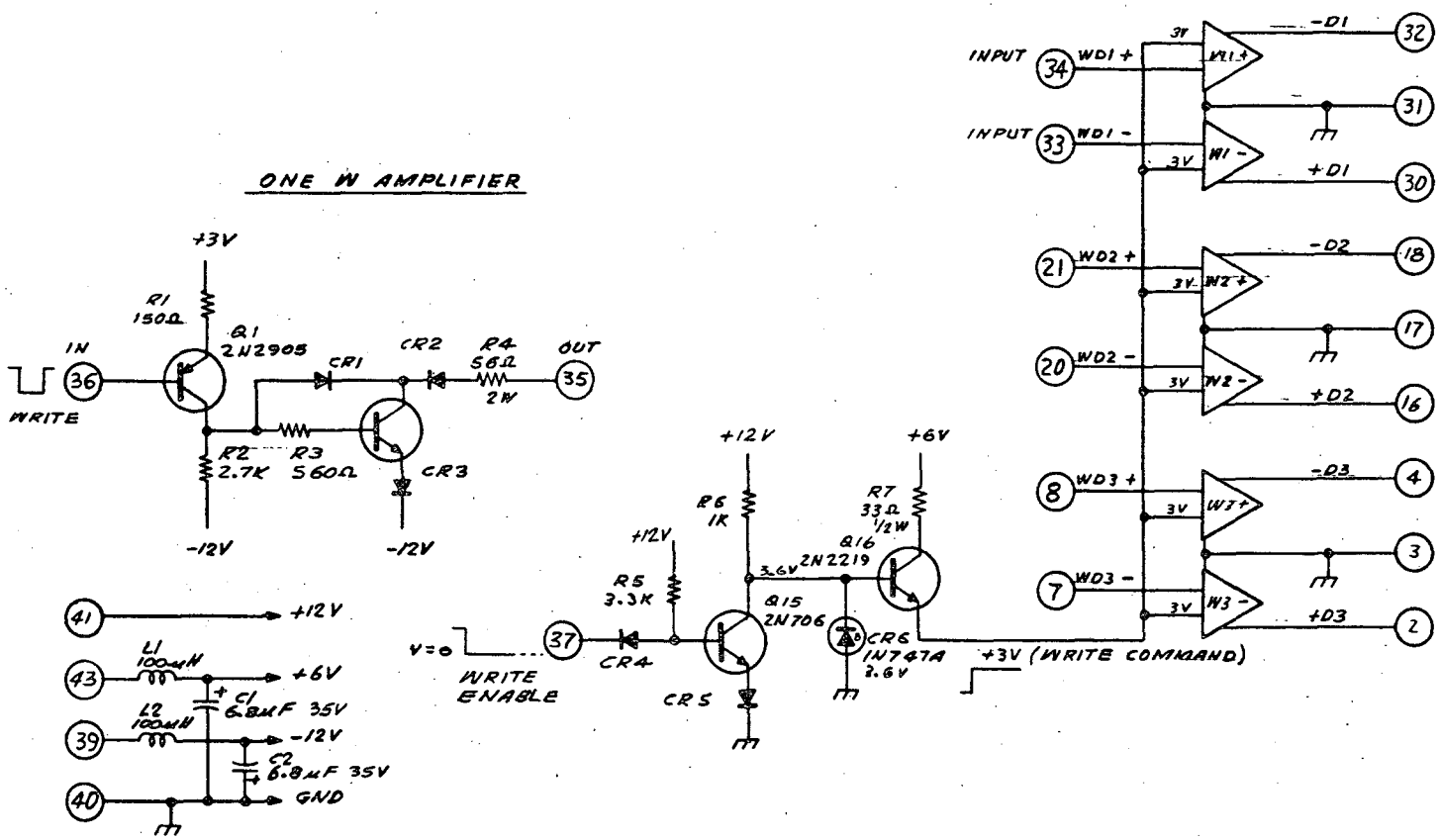



NOTES

1. ON ALL MICROLOGICS EXCEPT M11, Vcc IS ON PIN 14 & GND IS ON PIN 7. ON M11, Vcc = PIN 14 & GND = PIN 11
2. M1 = SN7410N; M2,4,15 = 936; M9, M3,6,10, M11 = SN7400N; M5 = SN7495N; SN7473N; M12 = SN7474A; M14 = SN7420N

XBL 696-724

Fig. 8. Write logic card (PC4).




XBL 696-720

Fig. 9. Write drivers (PC6).

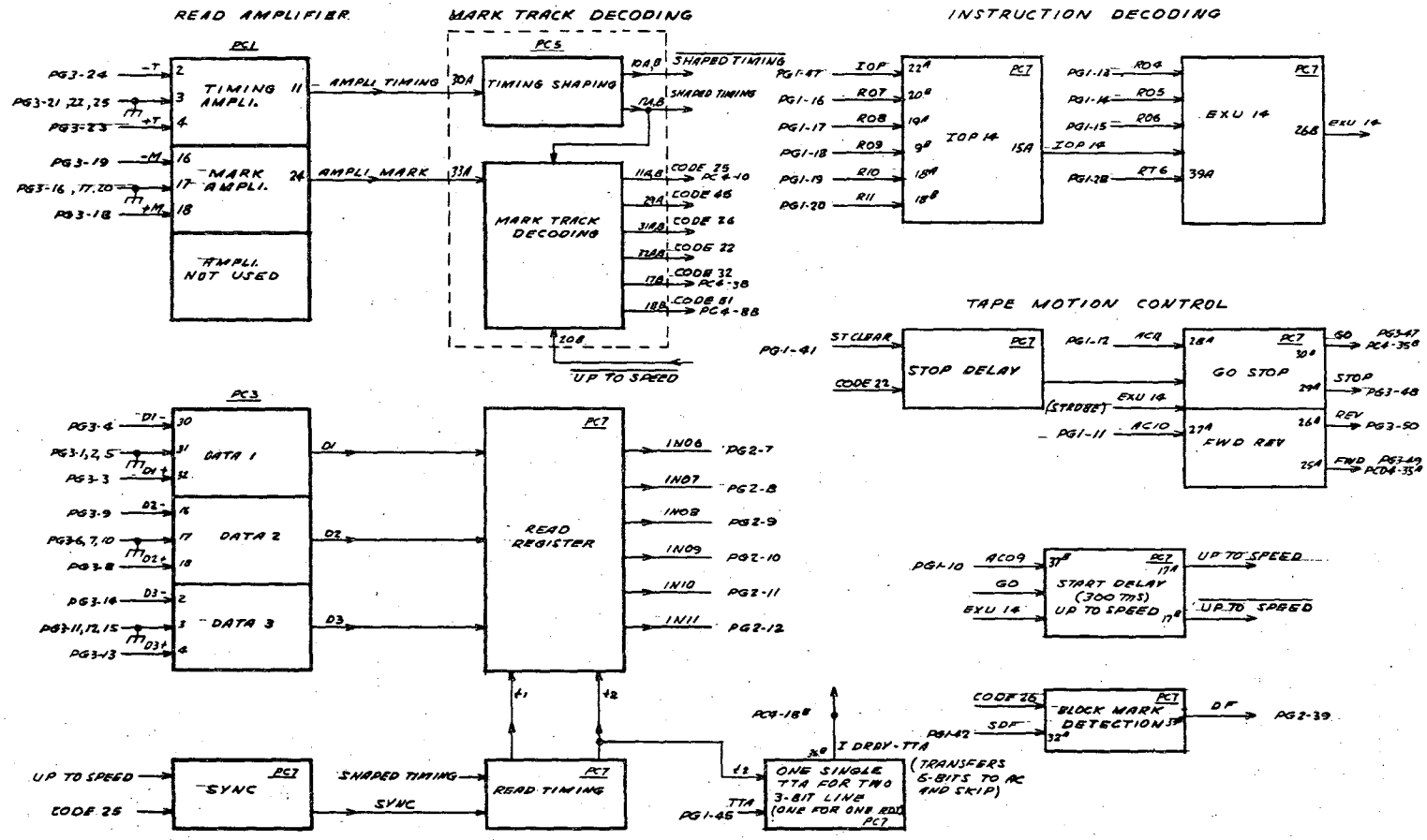
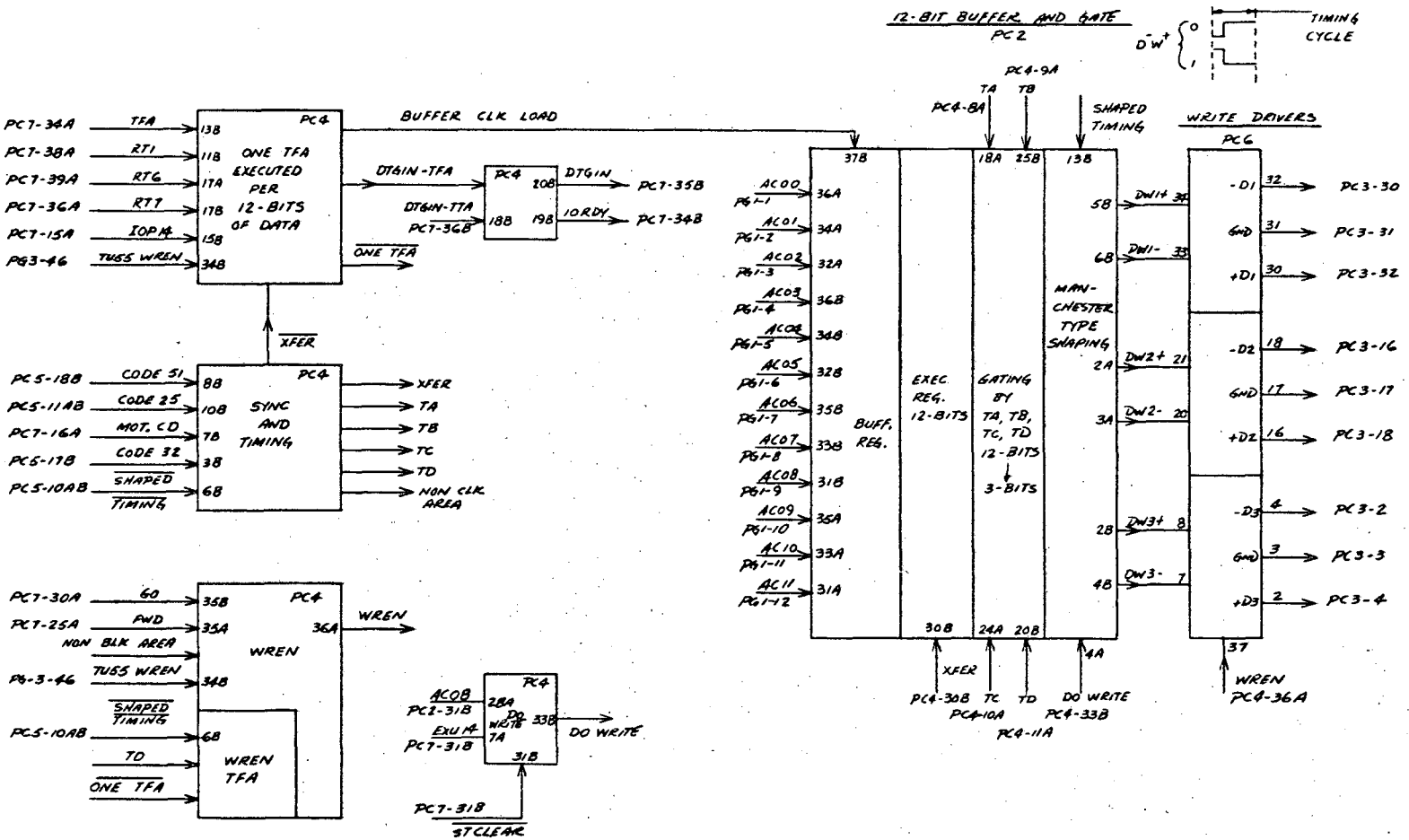



Fig. 10. Read logic block diagram.




XBL 696-726

**Fig. 11. Write logic diagram.**



UCRL-19230

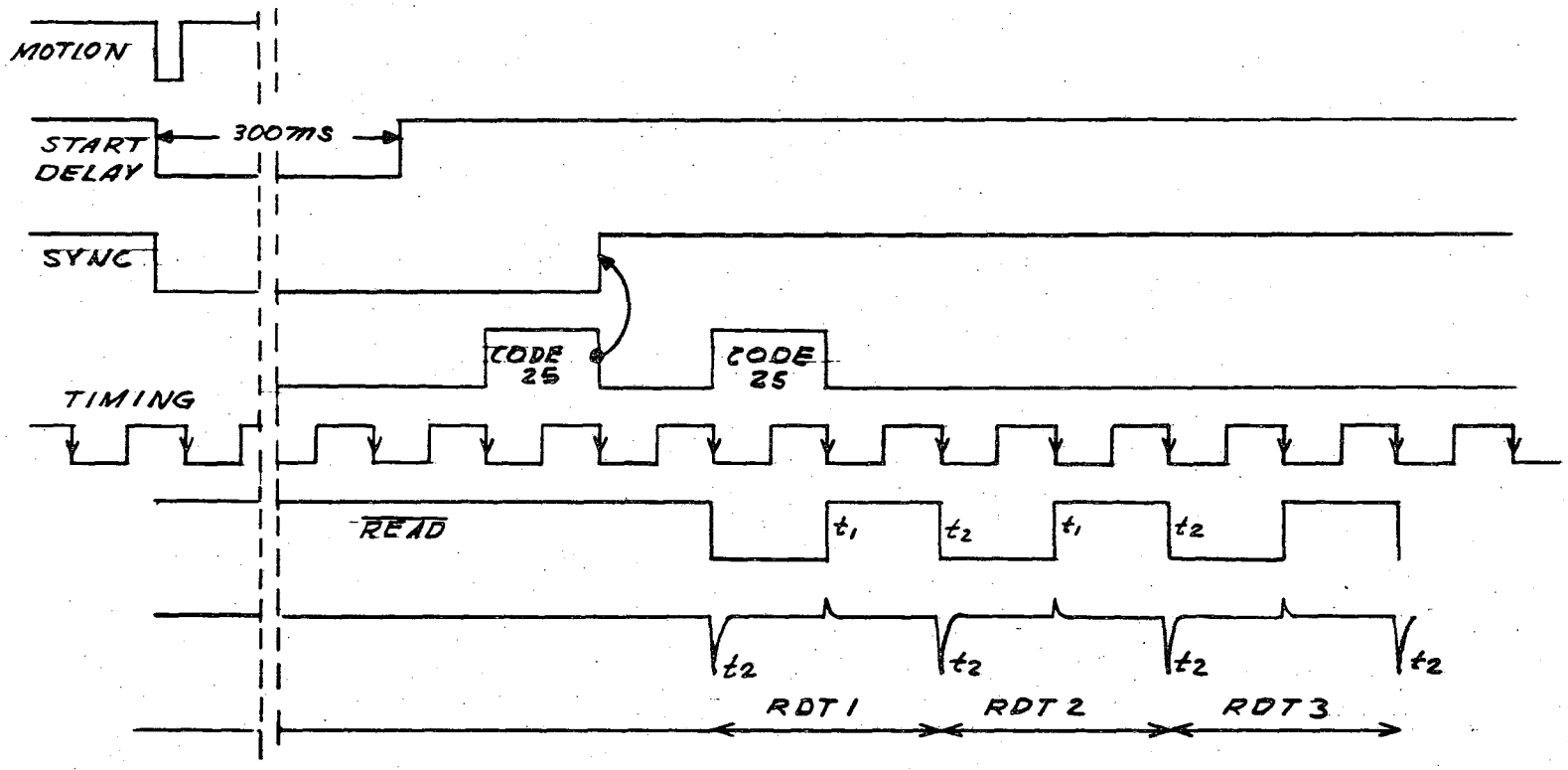
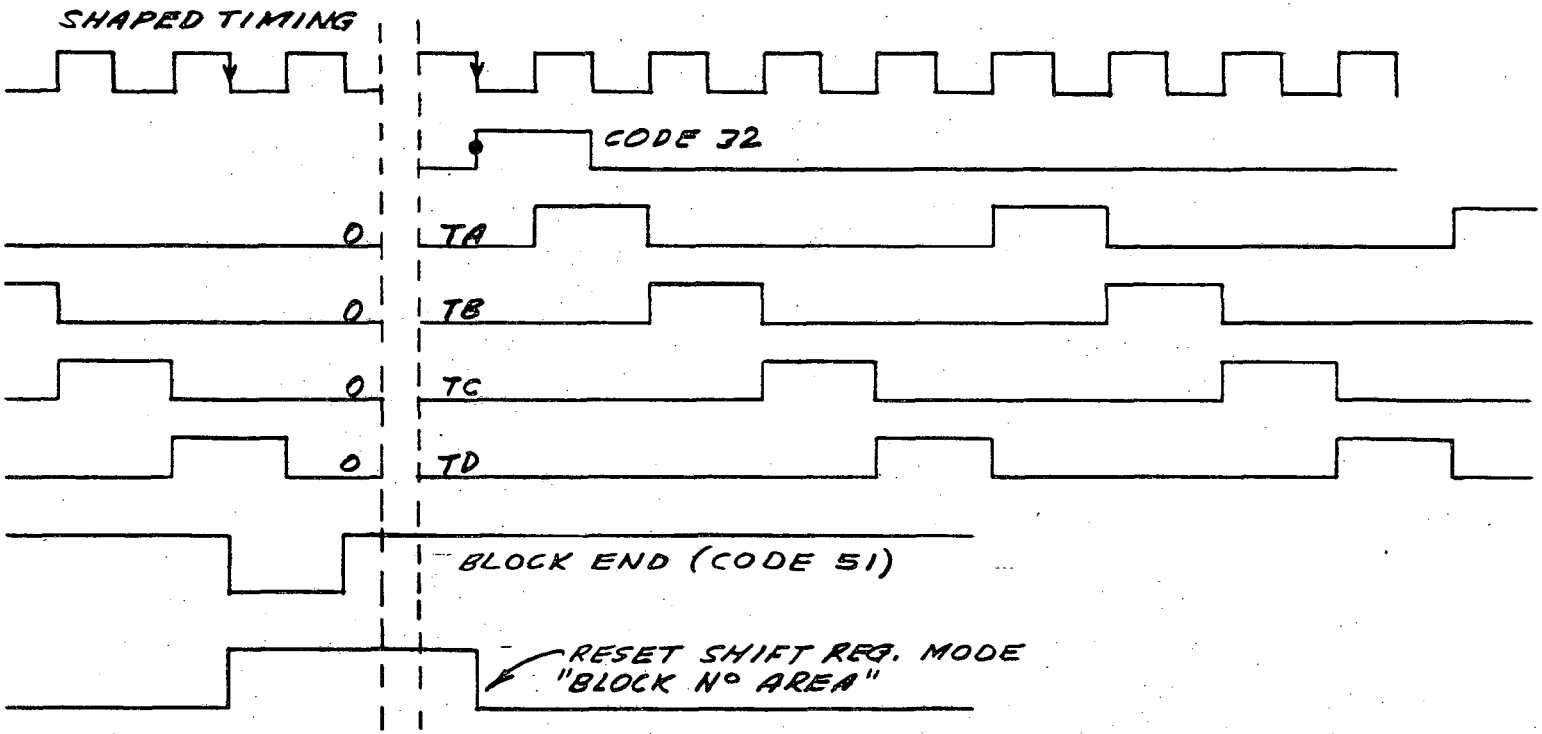
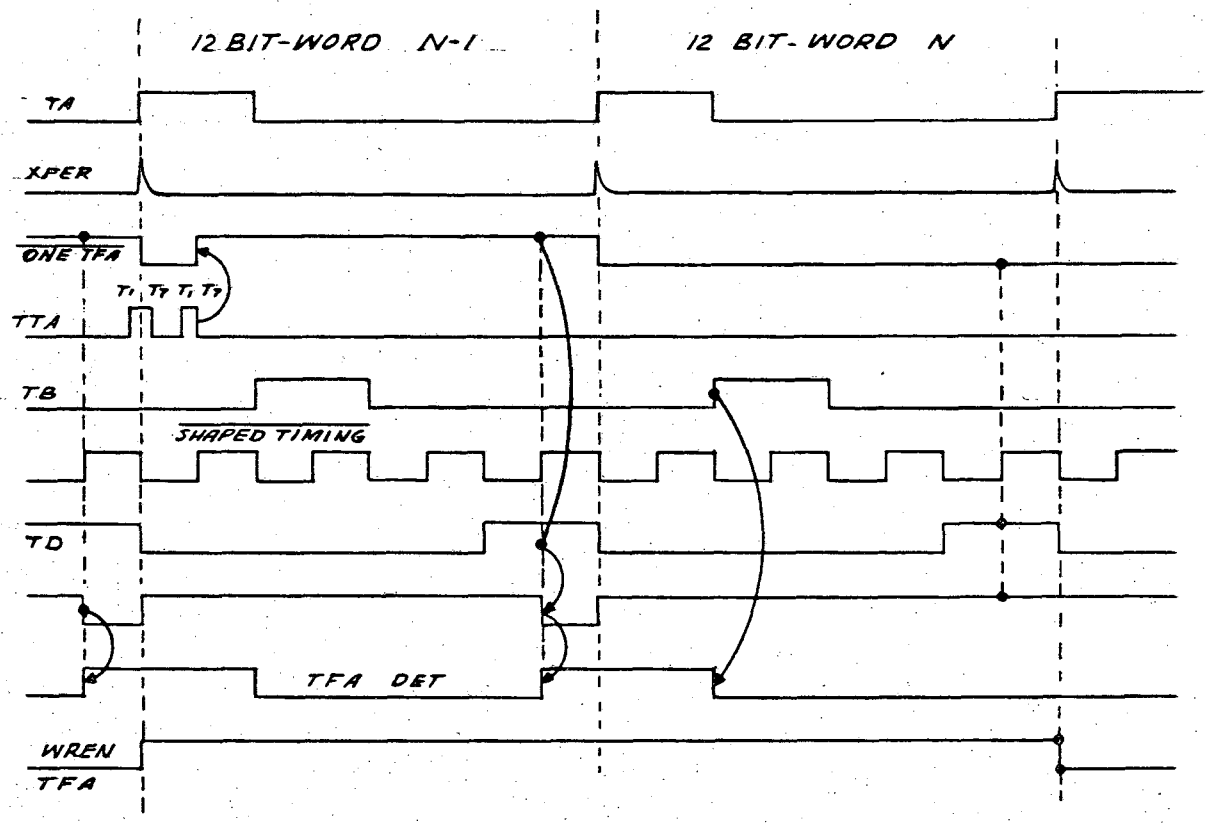



Fig. 12. Reading Synchronization and timing.




XBL 696-714

Fig. 13. Writing synchronization and timing.




A TFA (NOT SHOWN) WAS EXCUTED DURING WORD N-2 CAUSING WREN TFA TO BE TRUE DURING WORD N-1.

THE DIAG. SHOWS A TFA EXCUTED DURING WORD N-1, SETTING ONE TFA TRUE & CAUSING THE TFA DET. TO BE TRUE DURING XFER & CONSEQUENTLY SETTING THE WREN TFA FOR THE NEXT 12 BIT WORD # N UNTIL THE NEXT XFER (N+1)

XBL 696-716

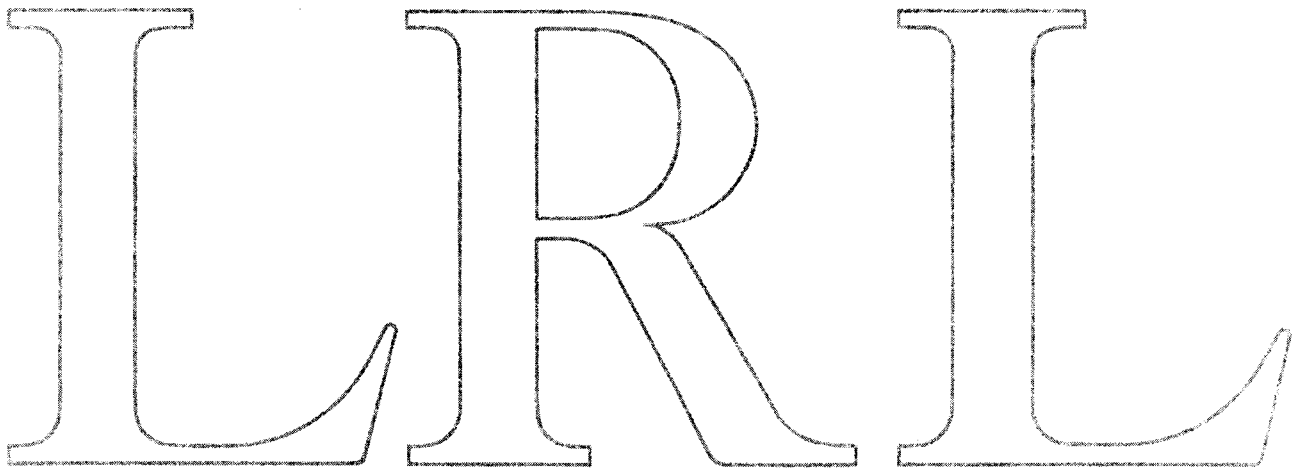
UCRL-19230

# AN INEXPENSIVE MAGNETIC TAPE FOR SMALL COMPUTER USE

M. A. Lechaczynski and L. B. Robinson

June 1969

AEC Contract No. W-7405-eng-48




LAWRENCE RADIATION LABORATORY
UNIVERSITY of CALIFORNIA BERKELEY

UCRL-19230