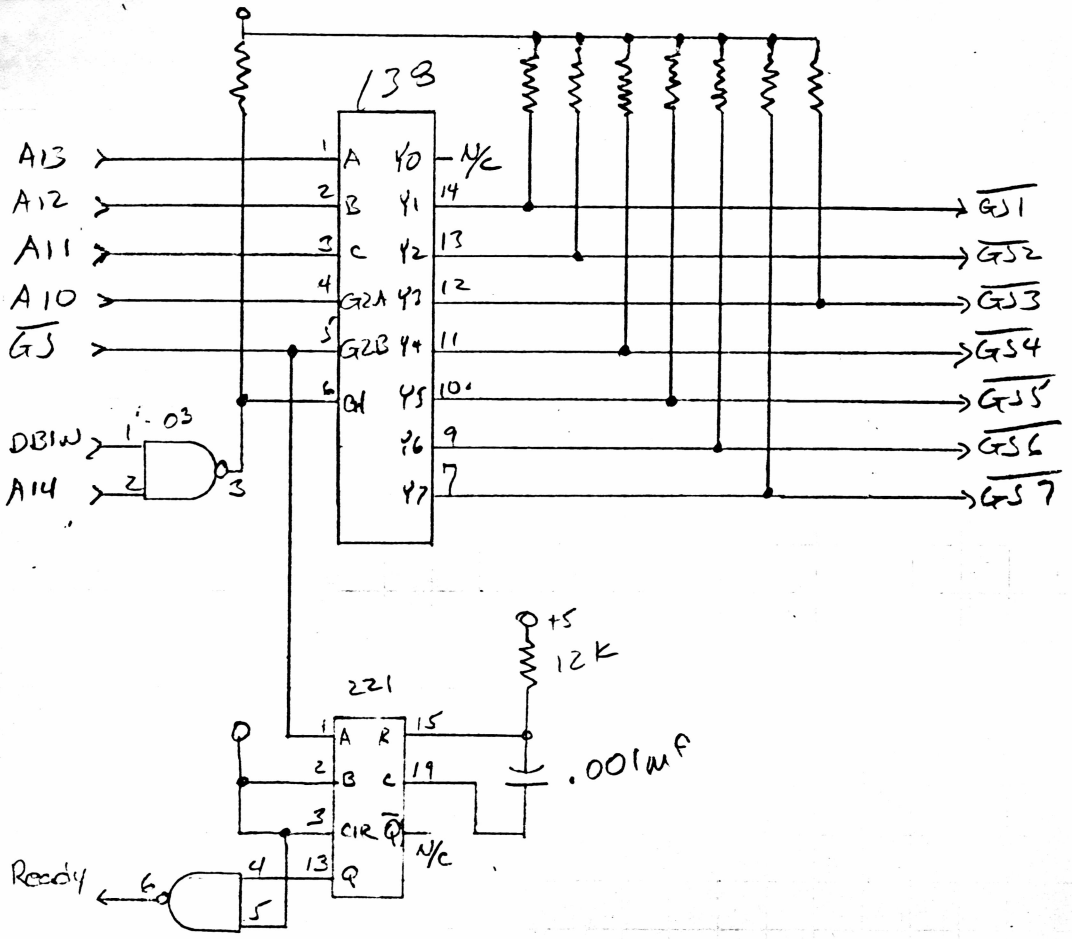
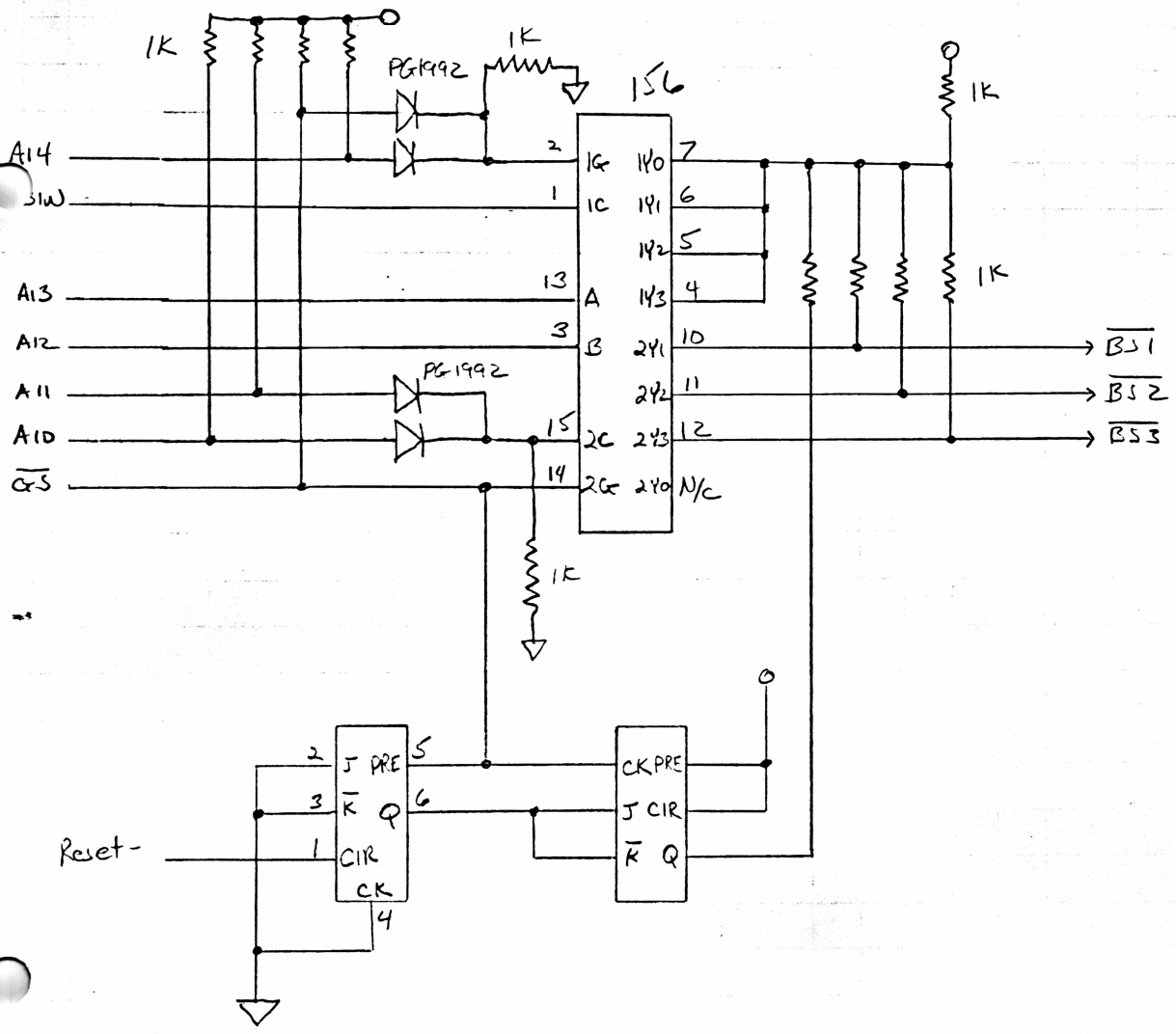

138
A13
A12
A11
A10
GJ
DBIN
A14
03
A
B
C
G2A
G2B
G1
Y0
Y1
Y2
Y3
Y4
Y5
Y6
Y7
N/C
GJ1
GJ2
GJ3
GJ4
GJ5
GJ6
GJ7
+5
12K
221
R
CLR
Q
.001MF
Ready

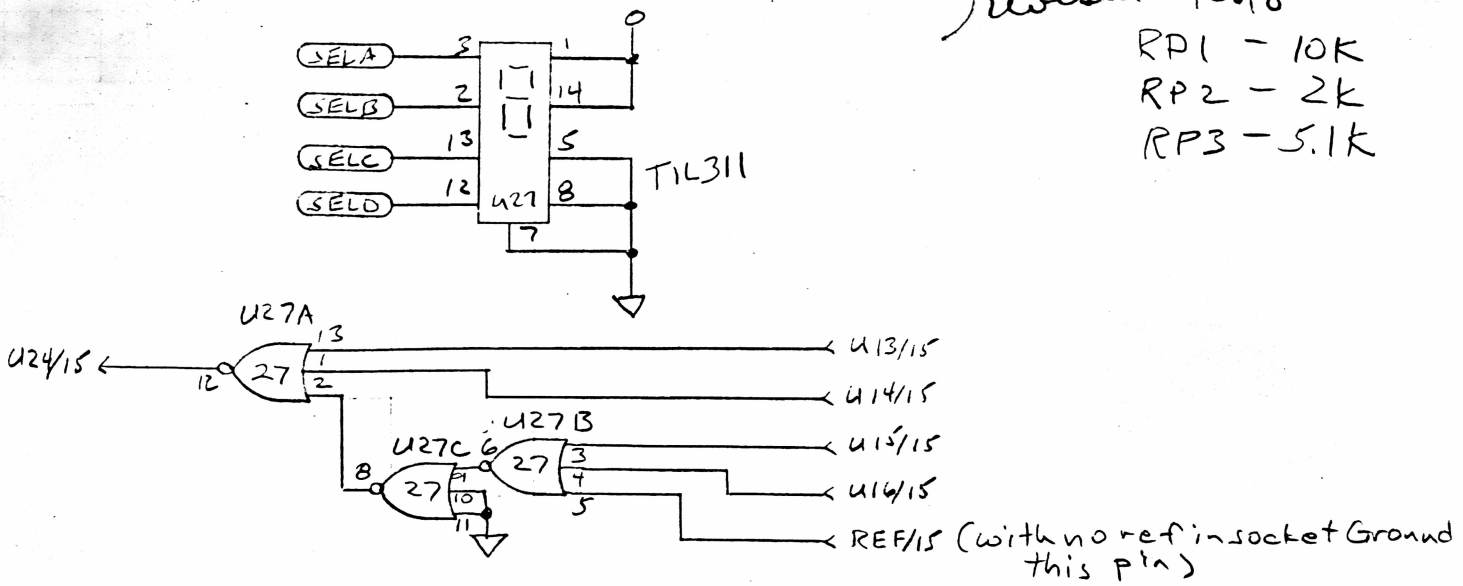


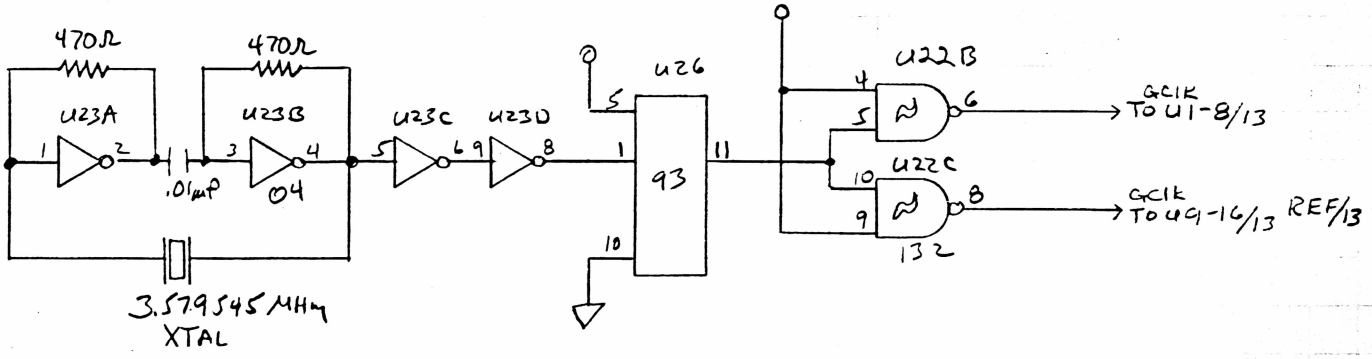
for 3 single or double grom pacs

3-8-80

revised 2/28/80

RP1 - 10K
RP2 - 2k
RP3 - 5.1k





1) U2-U16, REF - follow diagram for U1

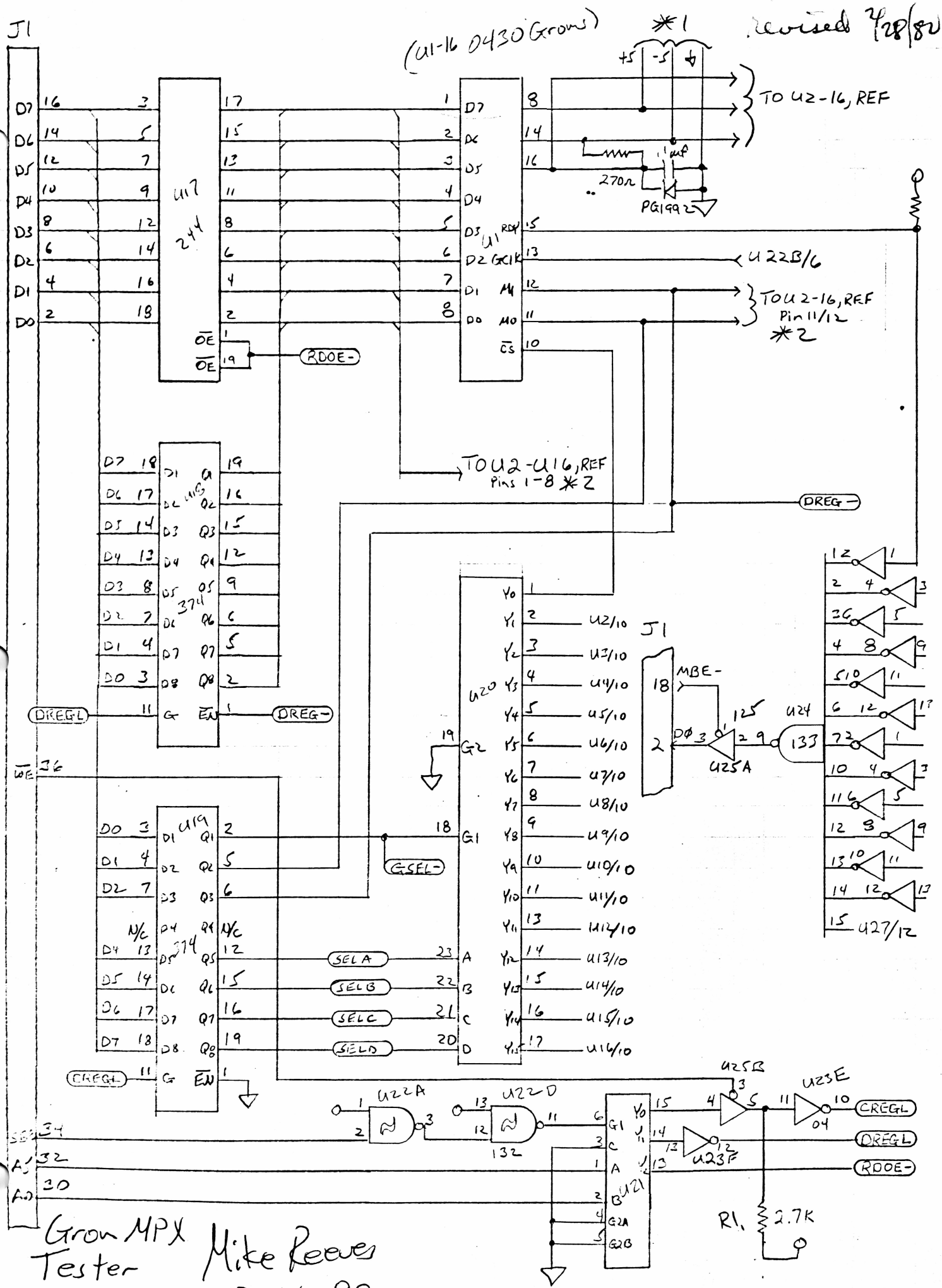
2) Power is supplied from an external variable supply

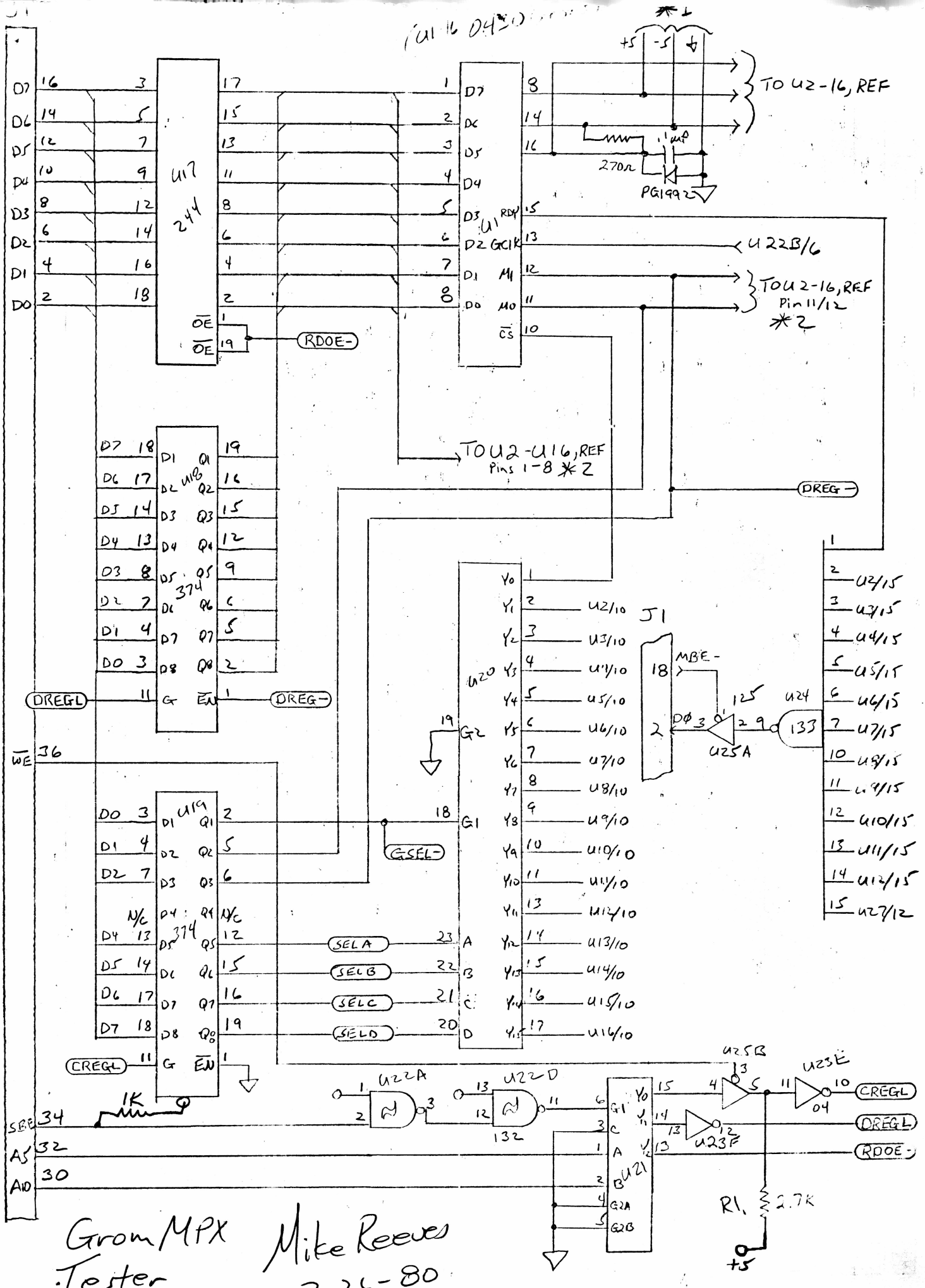
3) Cable used is 40 conductor twisted pair with every other pin ⏚.

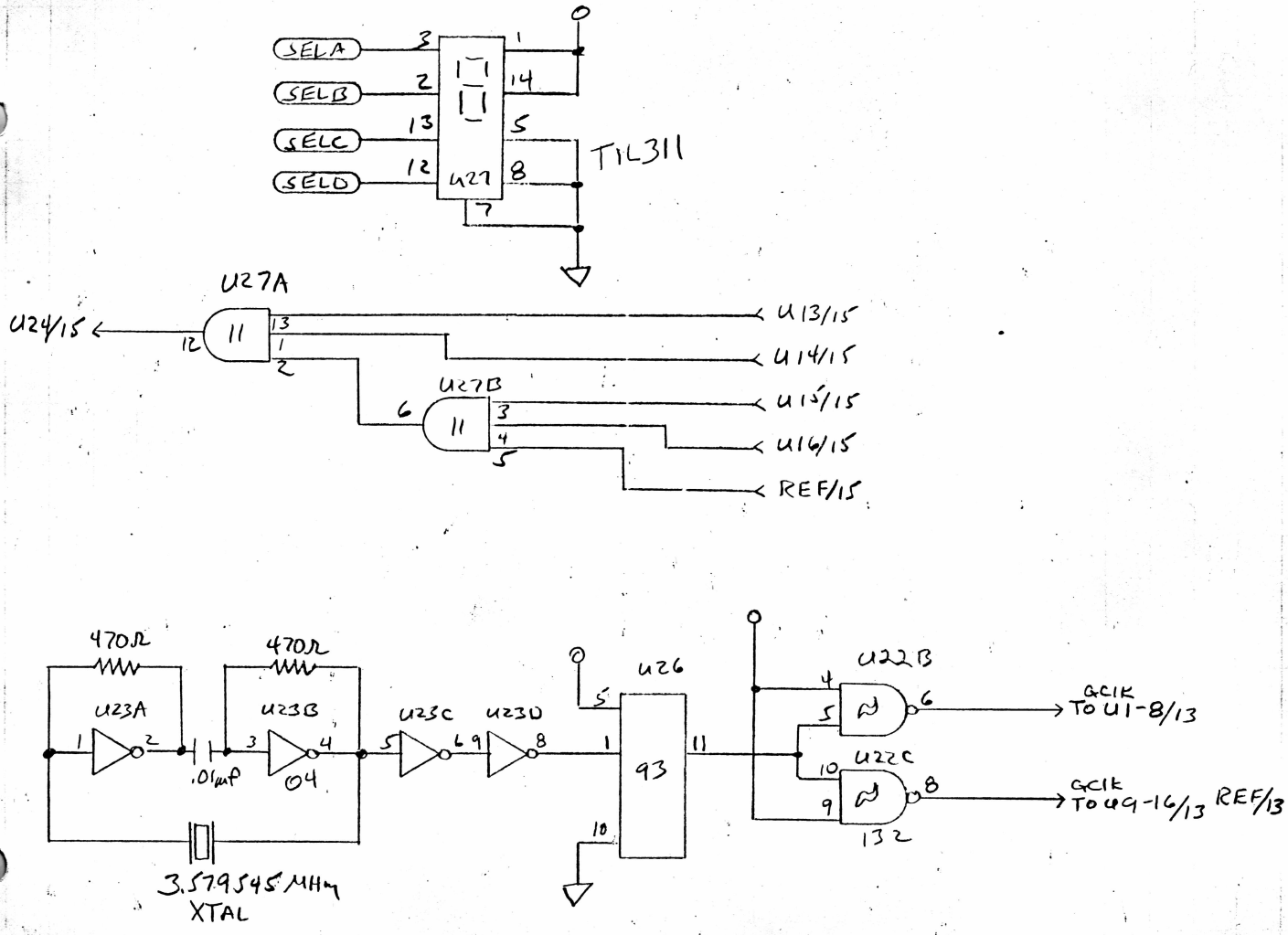
4) All input or I/O pins of U1-U16, REF are pulled up with 2.0-10K resistors in RP1-RP3 RP1-RP3 pin 16's are at +5 (Vcc)

5) U18, U20, U23, U26, & U22 are bypassed with .1μf cap Vcc to ⏚.

6) Supply inputs have 220mf caps to ⏚.


J1
(U1-16 0430 Grows)
*1
Revised 7/28/80
+5
-5
+5
TO U2-16, REF
1 µf
270Ω
PG1992
U1 RDY
U22B/6
TO U2-16, REF Pin 11/12 *2
U117 244
RDOE-
TO U2-U16, REF Pins 1-8 *2
DREG-
U118 374
DREGL
U120
U2/10
U3/10
U4/10
U5/10
U6/10
U7/10
U8/10
U9/10
U10/10
U11/10
U12/10
U13/10
U14/10
U15/10
U16/10
J1
18
MBE-
2
DØ
125
U25A
U24
133
U27/12
WE
36
U119 374
GSEL-
N/C
SELA
SELB
SELC
SELD
CREGL
U22A
U22D
132
U25B
U25E
04
U23F
CREGL
DREGL
RDOE-
SB
34
A1
32
A0
30
U121
R1, 2.7K
Grow MPX Tester
Mike Reeves
2-26-80


U16 D4-D7
+5
-5
TO U2-16, REF
270Ω
.1 μF
PG1992
U17 244
U1
RDY
D7
D6
D5
D4
D3
D2 GCLK
D1 M1
D0 M0
CS
TO U22B/6
TO U2-16, REF Pin 11/12 *2
TO U2-U16, REF Pins 1-8 *2
RDOE-
DREG-
U18 374
DREGL
DREG-
U19 374
CREGL
U20
G2
G1
A
B
C
SELA
SELB
SELC
SELD
GSEL-
U2/10
U3/10
U4/10
U5/10
U6/10
U7/10
U8/10
U9/10
U10/10
U11/10
U12/10
U13/10
U14/10
U15/10
U16/10
J1
MBE-
U24
133
U25A
U2/15
U3/15
U4/15
U5/15
U6/15
U7/15
U8/15
U9/15
U10/15
U11/15
U12/15
U27/12
WE 36
SBE 34
A5 32
A10 30
1K
U22A
U22D
132
U21
U25B
U23E
U23F
04
CREGL
DREGL
RDOE-
R1 2.7K
+5
Grom MPX Tester
Mike Reeves
2-26-80



1) U2-U16, REF - follow diagram for U1

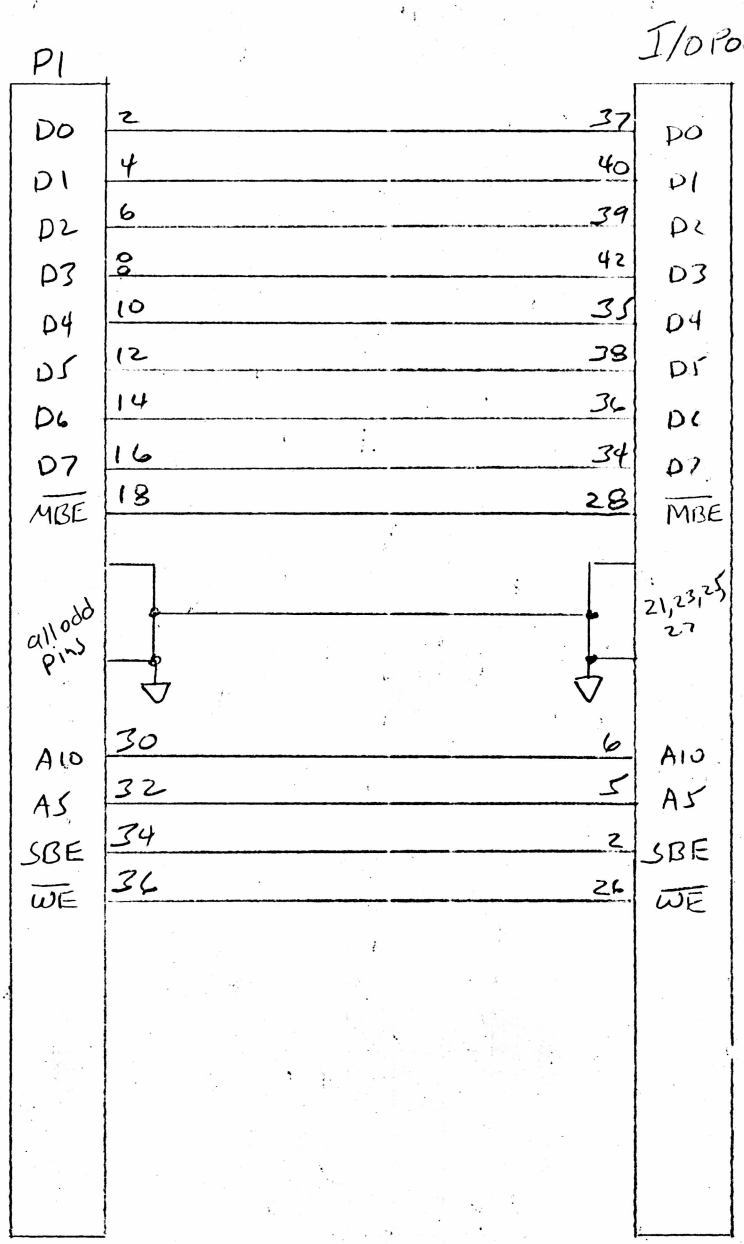
2) Power is supplied from an external variable supply

3) Cable used is 40 conductor twisted pair with every other pin ⏚.

4) All input or I/O pins of U1-U16, REF are pulled up with 5.1K resistors in RP1-RP3 RP1-RP3 pin 16 are at +5 (Vcc)

5) U18, U20, U23, U26, & U22 are bypassed with .1μf cap Vcc to ⏚.

6) Supply inputs have 220mf caps to ⏚.


P1
I/O Port
D0
D1
D2
D3
D4
D5
D6
D7
MBE
all odd pins
A10
A5
SBE
WE
2
4
6
8
10
12
14
16
18
30
32
34
36
37
40
39
42
35
38
36
34
28
21,23,25,27
6
5
2
26

Operation

Oscillator (GCLK)

Oscillator U23A-U23D is 3.579545 Xtal oscillator-buffer driving ÷8 counter U26. U26 output is 447.43 KHz which is fed into schmitt-trigger nand buffers to GClk inputs of U1-U16.

Select Logic

1 of 8 selector U21 is selected by SBE which is buffered by schimitt-trigger nands U22A-U22D. The outputs are selected by A5, A10 to provide 3 selects.

A) CREGL U21/15 output is enabled through tri-state buffer U25B by $\overline{WE}$. The output of U25B is pulled to +5 by R1 and fed into inverter U23E.

B) DREGL U21/14 output inverted.

C) RDOE- U21/13 output

Command Register Logic (CREGL)

D0-D2, D4-D7 are latched into U19 by CREGL and applied as follows:

Pin 2 is GSEL- which selects U20

Pin 5-6 are mode select lines for U1-U16, REF

Pin 12-19 are binary output select liner to U1-16

Data Register Logic (DREGL)

D0-D8 are latched into U18 by $\overline{DREGL}$ and outputs are enabled by DREG- (U19/6)

Read Data Logic (RDOE-)

D0-D8 from U1-16 is buffered by U17. U17 is enabled by RDOE-.

Status

Ready lines from U1-U16, REF are gated through U27A, U27B, & U24 to pin 2 of U25A. U25A is selected by MBE-. 1 on U25A pin 3 indicates a not ready condition.

Display

U27 TIL311 displays number (0-F) of UUT.

Command Format (CREGL)

D0 D1 D2 D3 D4 D5 D6 D7

$\overline{CS}$ M0 M1 X 

#of UUT (Binary)

$\overline{CS}$	M0	M1	
0	1	0	Write address (DREGL)
0	0	1	Read data (RDOE-)
0	1	1	Read address (RDOE-)
1	X	X	No operation

Addressing

CREGL - 9000 Hex
DREGL - 9400 Hex
RDOE- - 9020 Hex
MBE- - 4000 Hex

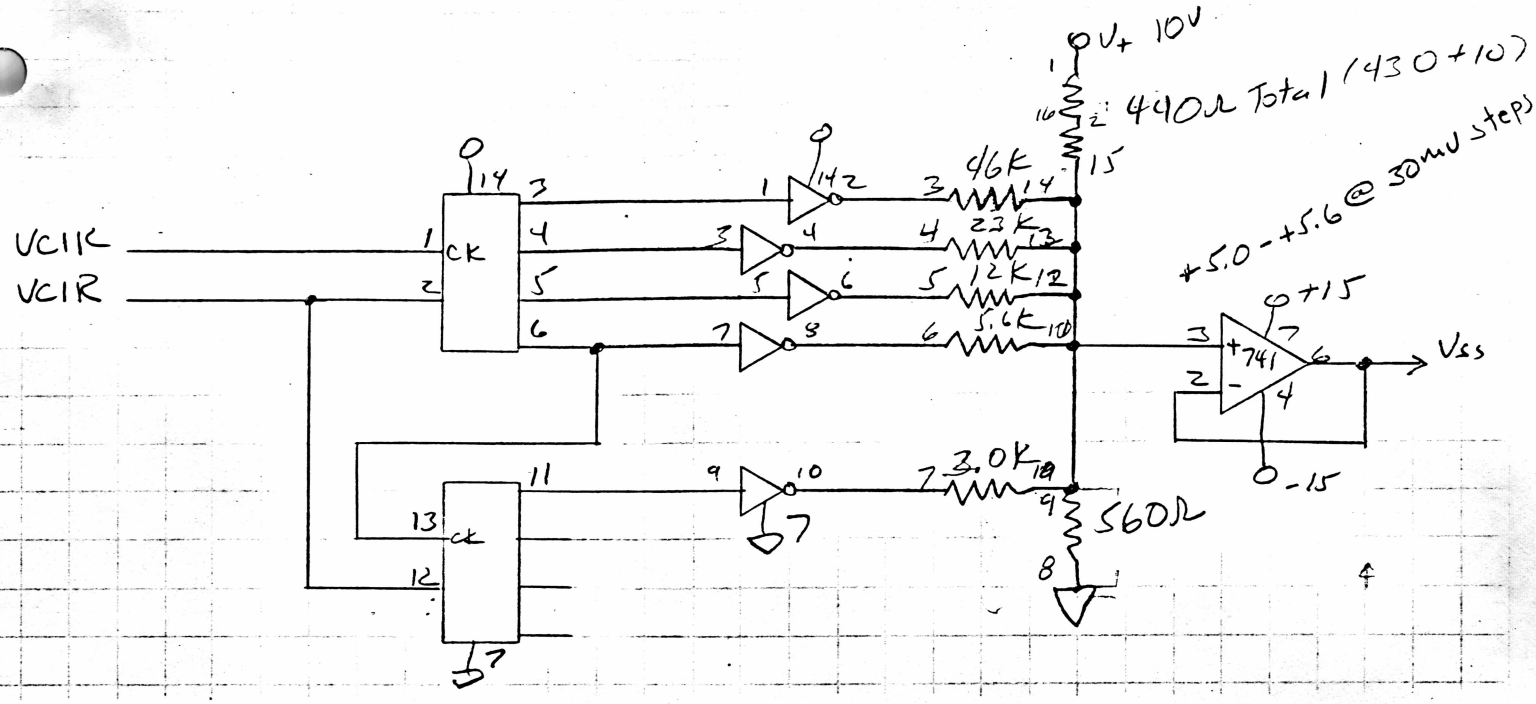
```
0001                        IDT 'GRMCHR'
0002                *
0003                *       THIS IS A CHARACTERAZATION TEST FOR
0004                *     0430 GROMS.  IT WILL ASK FOR AN INPUT
0005                *     FROM THE CONSOLE KEYBOARD AS TO THE
0006                *     GROM NUMBER.  IT WILL THEN BEGIN TO
0007                *     READ ADDRESSES AND READ DATA FOR 10^6
0008                *     PASSES AND THEN INCREMENT THE VOLTAGE
0009                *     (OOPS) AND REPEAT THE PROCESS.
0010                *
0011                *
0012                *
0013      9800  GROMRD EQU  >9800              GROM READ DATA
0014      9802  GROMRA EQU  >9802              GROM READ ADDRESS
0015      9C02  GROMWA EQU  >9C02              GROM WRITE ADDRESS
0016      F018  WHEX   EQU  >F018              HEX PRINT ROUTINE
0017      F00C  WMES   EQU  >F00C              PRINT MESSAGE
0018      F004  READ   EQU  >F004              KEYBOARD FETCH
0019      F122  CRLF   EQU  >F122              CARRIAGE RETURN
0020      C0F0  STAD   EQU  >C0F0              START ADDRESS BUFFER
0021      9000  VCLR   EQU  >9000 SP READ      MINIMUM VOLTAGE
0022      9400  VCLR   EQU  >9400 SP WRITE     INCREMENT VOLTAGE
0023                *
0024                ********************************************************
0025                *                                                      *
0026                *       START THE PROGRAM HERE                         *
0027                *                                                      *
0028                ********************************************************
0029                *
0030 0000 0200  START  LI   R0,MSG1
     0002 0130'
0031 0004 0420         BLWP @WMES              PRINT PROMPT
     0006 F00C
0032 0008 0420         BLWP @READ              GET THE CHARACTER
     000A F004
0033 000C C040         MOV  R0,R1
0034 000E 0200         LI   R0,CRLF
     0010 F122
0035 0012 0420         BLWP @WMES
     0014 F00C
0036 0016 0241         ANDI R1,>700            MASK OFF TOP 5 BITS
     0018 0700
0037 001A 0601         SWPB R1
0038 001C 0221         AI   R1,-2
     001E FFFE
0039 0020 04C2         CLR  R2
0040 0022 0601  STRT   DEC  R1                 GOT THE ADDRESS YET
0041 0024 1305         JEQ  STRT1              YEP! SO SKIP IT
0042 0026 0222         AI   R2,>2000           NO SO ADD SOME MORE
     0028 2000
0043 002A 10FB         JMP  STRT
0044 002C C802         MOV  R2,@STAD           SAVE START ADDRESS
     002E C0F0
0045 0030 0204  STRT1  LI   R4,>14             LOAD VOLTAGE COUNT
     0032 0014
0046 0034 04C6         CLR  R6
0047 0036 0205         LI   R5,>500
     0038 0500
0048 003A C805         MOV  R5,@VCLR           CLR VSS TO +5
     003C 9000
```

Selected by hardware (20 steps)

```
0049 003E C060  STRT2  MOV  @STAD,R1
     0040 C0F0
0050 0042 0221         AI   R1,>FFE          MIDDLE OF GROM PLEASE
     0044 0FFE
0051 0046 C081         MOV  R1,R2
0052 0048 0582         INC  R2
0053 004A 0209  STRT3  LI   R9,>4240         ONE MILLION READS
     004C 4240
0054 004E 020A         LI   R10,>000F                "
     0050 000F
0055 0052 0200-        LI   R0,MSG2
     0054 015A'
0056 0056 0420         BLWP @WMES
     0058 F00C
0057 005A C005         MOV  R5,R0
0058 005C 0420         BLWP @WHEX
     005E F018
0059 0060 0200         LI   R0,CRLF
     0062 F122
0060 0064 0420         BLWP @WMES
     0066 F00C
0061 0068 06A0  RD00   BL   @WTAD             WRITE ADDRESS
     006A 0104'
0062 006C 06A0         BL   @RA               READ ADDRESS
     006E 0114'
0063 0070 8083         C    R3,R2             IS IT GOOD
0064 0072 1301         JEQ  RD01              YEP! GO ON
0065 0074 0586         INC  R6                NO!! INC ERROR COUNT
0066 0076 0609  RD01   DEC  R9                IS IT
0067 0078 16F7         JNE  RD00              A MILLION
0068 007A 060A         DEC  R10               YET
0069 007C 16F5         JNE  RD00              NOP!
0070 007E 0225         AI   R5,>3             INCREMENT VOLTAGE
     0080 0003
0071 0082 C805         MOV  R5,@VCLK
     0084 9400
0072 0086 C186         MOV  R6,R6             ANY ERRORS
0073 0088 1302         JEQ  RD02
0074 008A 06A0         BL   @PNTERR           ALL AT ONCE THE ERRORS
     008C 0124'
0075 008E 0604  RD02   DEC  R4
0076 0090 16D6         JNE  STRT2             DO IT AGAIN
0077 0092 0204  STRT4  LI   R4,>14            RE-INITIALIZE FOR READ DATA
     0094 0014
0078 0096 04C6         CLR  R6
0079 0098 0205         LI   R5,>500
     009A 0500
0080 009C C805         MOV  R5,@CLR
     009E 9000
0081 00A0 C060  STRT5  MOV  @STAD,R1
     00A2 C0F0
0082 00A4 0221         AI   R1,>FFE
     00A6 0FFE
0083 00A8 0209  STRT6  LI   R9,>4240          THIS IS A READ DATA
     00AA 4240
0084 00AC 020A         LI   R10,>000F
     00AE 000F
0085 00B0 0200         LI   R0,MSG3
     00B2 0174'
0086 00B4 0420         BLWP @WMES
```

```
                00B6 F00C
0087            00B8 C005           MOV  R5,R0
0088            00BA 0420           BLWP @WHEX
                00BC F018
0089            00BE 0200           LI   R0,CRLF
                00C0 F122
0090            00C2 0420           BLWP @WMES
                00C4 F00C
0091            00C6 0420           BLWP @WTAD              WRITE ADDRESS
                00C8 0104'
0092            00CA D0A0           MOVB @GROMRD,R2         1ST BYTE
                00CC 9800
0093            00CE 06A0  RD04     BL   @WTAD              WRITE ADDRESS
                00D0 0104'
0094            00D2 D0E0           MOVB @GROMRD,R3         GET 2ND BYTE
                00D4 9800
0095            00D6 9083           CB   R3,R2              DO THEY AGREE
0096            00D8 1301           JEQ  RD05               OK GO ON
0097            00DA 0586           INC  R6                 NOP! SO INC R6
0098            00DC 0609  RD05     DEC  R9                 IS IT
0099            00DE 16F7           JNE  RD04               ONE MILLION
0100            00E0 060A           DEC  R10                READS YET
0101            00E2 16F5           JNE  RD04               NOP GO BACK
0102            00E4 0225           AI   R5,>3              INC VOLTAGE
                00E6 0003
0103            00E8 C805           MOV  R5,@VCLK
                00EA 9400
0104            00EC C186           MOV  R6,R6              ANY ERRORS THIS PASS
0105            00EE 1302           JEQ  RD06
0106            00F0 06A0           BL   @PNTERR
                00F2 0124'
0107            00F4 0604  RD06     DEC  R4                 ARE WE DONE
0108            00F6 16D8           JNE  STRT6
0109            00F8 0200           LI   R0,MSG4            AH! THAT'S THE RUB!!
                00FA 018E'
0110            00FC 0420           BLWP @WMES              LET EM KNOW
                00FE F00C
0111            0100 0460           B    START              AND THEN RESTART
                0102 0000'
0112            0104 D801  WTAD     MOVB R1,@GROMWA         WRITE ADDRESS ROUTINE
                0106 9C02
0113            0108 06C1           SWPB R1
0114            010A 1000           NOP
0115            010C D801           MOVB R1,@GROMWA
                010E 9C02
0116            0110 06C1           SWPB R1
0117            0112 045B           RT
0118            0114 D0E0  RA       MOVB @GROMRA,R3         READ ADDRESS ROUTINE
                0116 9802
0119            0118 06C3           SWPB R3
0120            011A 1000           NOP
0121            011C D0E0           MOVB @GROMRA,R3
                011E 9802
0122            0120 06C3           SWPB R3
0123            0122 045B           RT
0124            0124 0200  PNTERR   LI   R0,MSG5            PRINT ERROR ROUTINE
                0126 01A6'
0125            0128 0420           BLWP @WMES
                012A F00C
0126            012C C004           MOV  R6,R0
```

```
0127 012E 0420          BLWP @WHEX
     0130 F018
0128 0132 0200          LI   R0,CRLF
     0134 F122
0129 0136 0420          BLWP @WMES
     0138 F00C
0130 013A 045B          RT
0131 013C   49  MSG1    TEXT 'INPUT LAST DIGIT OF GROM---'
0132 0157   0D          BYTE >D,>D,0
0133 015A   52  MSG2    TEXT 'READ ADDRESS--        VSS='
0134 0173   00          BYTE 0
0135 0174   52  MSG3    TEXT 'READ DATA-----        VSS='
0136 018D   00          BYTE 0
0137 018E   45  MSG4    TEXT 'END OF CYCLE '
0138 019C 0D0D          DATA >0D0D,>0D0D,>0D0D,>0D0D,>0D00
0139 01A6   4E  MSG5    TEXT 'NUMBER OF ERRORS IS--'
0140 01BB   00          BYTE 0
0141 01BC
0142 01BC
0143 01BC
0144 01BC
0145 01BC
0146 01BC
0147 01BC
0148 01BC
0149 01BC
0150 01BC
NO ERRORS,      NO WARNINGS
```


V+ 10V
490Ω Total (430+10)
+5.0 - +5.6 @ 30mV steps
VC1K
VC1R
CK
46K
23K
12K
5.6K
3.0K
560Ω
+15
741
Vss
-15

BY MIKE REEVES 12/5/79

THIS IS THE THEORY OF OPERATION FOR THE I/O PARAMETER TEST SET.

TESTS ARE AS FOLLOWS:

1) I/O PARAMETER TEST
2) GROM CRC TEST
3) GROM WRITE ADDRESS TEST
4) CONSOLE RAM TEST
5) I/O LOADING CHARACTERISTICS

DISPLAY IS AS FOLLOWS:

```
        EOT   *      *  TIP
           FAIL  PASS
  I/O PT    *      *        DISPLAY INDICATES A
  GROM CRC  *      *        PASS OR A FAIL CON-
  GROM WA   *      *        DITION ON EACH TEST
  RAM MSB   *      *
  RAM LSB   *      *

  GROM #  ====
          I  I          """ START SWITCH
          I  I           O
          ====          """
```

1) I/O PARAMETER TEST

THIS TEST USES LM339 VOLTAGE COMPARATERS, THE COMPARATORS ARE SET AT 4.2V REFERENCE LEVEL BY THE REGULATOR CIRCUIT OF U12. VCC IS AT 12V FOR COMPARATOR ACCURACY. THE OUTPUTS OF THE LM339'S ARE OPEN COLLECTOR, THUS REQUIRING PULL-UPS TO SUPPLY U11 INPUTS WITH THE PROPER LEVELS. U7A IS USED TO CLOCK THE DATA INTO U7B AT THE PROPER TIME,(THE SECOND PULSE OF WRITE ENABLE).

2) GROM CRC TEST

THIS TEST USES A SOFTWARE CYCLIC REDUNDANCY CHECK TO VERIFY THE DATA IN EACH GROM. IF THE CRC TEST FAILS, THE DISPLAY WILL LEAVE THE FAIL LED ON AND DISLPAY THE GROM#(3,4,5) IN ERROR.

3) GROM WRITE ADDRESS TEST

THIS TEST USES A SOFTWARE RANDOM NUMBER GENERATOR TO WRITE A RAMDOM ADDRESS INTO EACH GROM RESPECTIVELY 5 TIMES. IT DOES 5 DUMMY READS FROM EACH GROM 5 TIMES ON THE FIRST PASS, 4 ON THE SECOND AND SO ON. EACH PASS WRITES DIFFERENT ADDRESSES INTO THE 3 GROMS, READS THE A NUMBER OF BYTES EQUAL TO THE COUNTDOWN. FAILURES WILL LEAVE THE FAIL LED ON AND DISPLAY THE GROM#. IF BOTH THE CRC AND WA TESTS FAIL, THE GROM# DISPLAYED WILL BE FOR THE WA TEST..

THIS TEST IS ALSO IN THE SOFTWARE. IT WRITES 128 BYTES THEN READS 128 BYTES ON THE FIRST PASS. THE DATA AT THE END OF THE FIRST PASS WILL BE 80H. THIS IS THE START DATA FOR THE SECOND PASS. THE TEST IS OFFSET BY 0101H EACH PASS AND CYCLES UNTIL EACH BYTE HAS HAD THE FULL 256 BIT PATTERN WRITTEN AND READ 4 TIMES. A FAILURE WILL LEAVE THE FAIL LED FOR THE MSB, LSB, OR BOTH ON.

5) I/O LOADING CHARACTERISTICS

I/O LOADING IS ACCOMPLISHED AS FOLLOWS:

DATA LINES- WERE MEASURED WITH A BOONTON CAPACITANCE METER AND SUFFICIENT CAPACITANCE WAS ADDED TO EACH LINE SUCH THAT THE TOTAL CAPACITANCE WAS EQUIVALENT TO THE CAPACITANCE SPECIFIED IN THE I/O SPEC. ALSO A LIGHT RESISTIVE LOAD(100KOHM) WAS ADDED TO SIMULATE HIGH IMPEDANCE LOADING AS LOWER RESISTANCE LOADS ACT AS VOLTAGE DIVIDERS AND DROP TOO MUCH VOLTAGE.

ADDRESS LINES- WERE TREATED IN A SIMILAR MANNER TO THE DATA LINES AND SUFFICIENT CAPACITANCE WAS ADDED TO MEET THE I/O SPEC ALSO.

```
SOURCE ACCESS NAME=      HC2.MIKE.SRC.GRCR
OBJECT ACCESS NAME=      HC2.MIKE.OBJ.GRCR
LISTING ACCESS NAME=     HC2.MIKE.LST.GRCR
ERROR ACCESS NAME=
OPTIONS=                 XREF
MACRO LIBRARY PATHNAME=
```

```
0001                    IDT  'GRCR'
0002               *
0003               *      THIS IS A IS A SERIES OF TEST FOR
0004               *    PRODUCTION.  TESTS ARE AS FOLLOWS:
0005               *      1    I/O PARAMETRIC TEST
0006               *      2    GROM CRC TEST
0007               *      3    GROM WRITE ADDRESS TEST
0008               *      4    CONSOLE RAM TEST
0009               *
0010               *
0011               *      11/29/79
0012               *
0013       9802    GRA    EQU  >9802          GROM READ ADDRESS
0014       9C02    GWA    EQU  >9C02          GROM WRITE ADDRESS
0015       9800    GRD    EQU  >9800          GROM READ DATA
0016       F800    DSPLY  EQU  >F800          DISPLAY ERROR ADDRESS
0017       8300    RAMUT  EQU  >8300          START TESTING HERE
0018       F000    CPURAM EQU  >F000          WORKSPACE/SCRATCH PAD
0019       F400    IOTAD  EQU  >F400          I/O TEST ADDRESS
0020       F018    WHEX   EQU  >F018          PRINT HEX ROUTINE
0021               *
0022 0000 02E0          LWPI CPURAM
     0002 F000
0023 0004 020A          LI   R10,>78
     0006 0078
0024 0008 C80A          MOV  R10,@DSPLY    LIGHT TIP LED
     000A F800
0025               *
0026               *      THIS IS AN I/O PARAMETER TEST.
0027               *    IT WILL READ A NON-EXISTANT MEMORY
0028               *    FROM THE I/O PORT.  THE HARDWARE
0029               *    WILL TEST THE LEVEL OF THE DATA BUS
0030               *
0031 000C 0201  DELAY  LI   R1,>FFF
     000E 0FFF
0032 0010 04C3  DEL1   CLR  R3
0033 0012 C803          MOV  R3,@IOTAD     .FAIL I/O TEST
     0014 F400
0034 0016 0601          DEC  R1            .TO CHANGE LED
0035 0018 16FB          JNE  DEL1
0036 001A 0203          LI   R3,>10
     001C 0010
0037 001E 0701          SETO R1
0038 0020 C801  IOT1   MOV  R1,@IOTAD     .TEST I/O 16 TIMES
     0022 F400
0039 0024 0603          DEC  R3
0040 0026 16FC          JNE  IOT1
0041               *
0042               *      THIS IS A CRC CHECK OF THE GROM.
0043               *    CRC CODES ARE ONLY GOOD FOR THE FOLLOWING:
0044               *      2003D
0045               *      2004C
0046               *      2005D
0047               *
0048       0028' GRCR   EQU  $             START OF PROGRAM
0049 0028 04C3          CLR  R3            PASS FAIL INDICATOR
0050 002A 04C1          CLR  R1
0051 002C 06A0          BL   @WTAD
     002E 0078'
0052 0030 06A0          BL   @CRCALC
```

```
      0032 0088'
0053  0034 56AD          DATA >56AD          CRC FOR GROM 1
0054  0036 020C          LI   R12,>03
      0038 0003
0055  003A 0201          LI   R1,>2000
      003C 2000
0056  003E 06A0          BL   @WTAD
      0040 0078'
0057  0042 06A0          BL   @CRCALC
      0044 0088'
0058  0046 CB50          DATA >CB50          CRC FOR GROM 2
0059  0048 020C          LI   R12,>04
      004A 0004
0060  004C 0201          LI   R1,>4000
      004E 4000
0061  0050 06A0          BL   @WTAD
      0052 0078'
0062  0054 06A0          BL   @CRCALC
      0056 0088'
0063  0058 4664          DATA >4664          CRC FOR GROM 3
0064  005A 020C          LI   R12,>05
      005C 0005
0065  005E C0C3          MOV  R3,R3          PASS ?
0066  0060 1606          JNE  EOT            NO
0067  0062 024A          ANDI R10,>68
      0064 0068
0068  0066 C80A          MOV  R10,@DSPLY
      0068 F800
0069  006A 0460          B    @WRAD
      006C 00BC'
0070  006E E28C  EOT     SOC  R12,R10
0071  0070 C80A          MOV  R10,@DSPLY
      0072 F800
0072  0074 0460          B    @WRAD
      0076 00BC'
0073        0078' WTAD    EQU  $              ROUTINE TO WRITE GROM ADDRESS
0074  0078 D801          MOVB R1,@GWA        MSB ADDR
      007A 9C02
0075  007C 06C1          SWPB R1
0076  007E 1000          NOP
0077  0080 D801          MOVB R1,@GWA        LSB ADDR
      0082 9C02
0078  0084 1000          NOP
0079  0086 045B          RT                  DONE
0080        0088' CRCALC  EQU  $              CRC CALC ROUTINE
0081  0088 0708          SETO R8
0082  008A 0206          LI   R6,>1800       # BYTES PER GROM
      008C 1800
0083  008E 04C7  CRLP    CLR  R7
0084  0090 D1E0          MOVB @GRD,R7
      0092 9800
0085  0094 2A07          XOR  R7,R8
0086  0096 C1C8          MOV  R8,R7
0087  0098 0947          SRL  R7,4
0088  009A 29C8          XOR  R8,R7
0089  009C 0247          ANDI R7,>FF00
      009E FF00
0090  00A0 0947          SRL  R7,4
0091  00A2 2A07          XOR  R7,R8
0092  00A4 0B77          SRC  R7,7
```

```
0093 00A6 2A07          XOR  R7,R8
0094 00A8 06C8          SWPB R8
0095 00AA 0606          DEC  R6
0096 00AC 16F0          JNE  CRLP
0097 00AE 823B          C    *R11+,R8      CRC OK ?
0098 00B0 1603          JNE  CRRET         NO, RETURN
0099 00B2 022B          AI   R11,2         BUMP RET ADDRESS
     00B4 0002
0100 00B6 045B          RT
0101 00B8 0703  CRRET   SETO R3            SET FAIL FLAG
0102 00BA 045B          RT
0103            *
0104            *       THIS ROUTINE DOES 5 RANDOM READ ADDRESSES
0105            *     FROM ALL GROMS.  IT DISPLAYS ERRORS IN THE
0106            *     LED DISPLAY.
0107            *
0108 00BC 0204  WRAD    LI   R4,>05        LOAD COUNT
     00BE 0005
0109 00C0 04CE          CLR  R14
0110 00C2 04C1  WA01    CLR  R1            FIRST GROM
0111 00C4 020D          LI   R13,>03
     00C6 0003
0112 00C8 06A0          BL   @RNDG         GET THE RAMDOM NUMBER
     00CA 00F8'
0113 00CC 0201  WA02    LI   R1,>2000      SECOND GROM
     00CE 2000
0114 00D0 058D          INC  R13
0115 00D2 06A0          BL   @RNDG
     00D4 00F8'
0116 00D6 0201  WA03    LI   R1,>4000      LAST GROM
     00D8 4000
0117 00DA 058D          INC  R13
0118 00DC 06A0          BL   @RNDG
     00DE 00F8'
0119 00E0 0604          DEC  R4            FIVE READ ADDRESSES
0120 00E2 16EF          JNE  WA01          DO IT AGAIN
0121 00E4 1030          JMP  ERDSP         DISPLAY ERRORS
0122 00E6 C249  RND     MOV  R9,R9         . RANDOM NUMBER GEN
0123 00E8 1601          JNE  RND1
0124 00EA 0589          INC  R9
0125 00EC 0A19  RND1    SLA  R9,1
0126 00EE 1702          JNC  RND2
0127 00F0 2A60          XOR  @CRC,R9
     00F2 00F6'
0128 00F4 045B  RND2    RT
0129 00F6 1021  CRC     DATA >1021
0130 00F8 C3CB  RNDG    MOV  R11,R15       PUSH RETURN REG
0131 00FA 06A0          BL   @RND          GET THE RANDOM NUMBER
     00FC 00E6'
0132 00FE C149          MOV  R9,R5
0133 0100 0245          ANDI R5,>1FFF      MASK OFF TOP 3 BITS
     0102 1FFF
0134 0104 0285  RNDG1   CI   R5,>17F0      ONLY 1800 BYTES IN GROM
     0106 17F0
0135 0108 1202          JLE  RNDG2
0136 010A 0605          DEC  R5
0137 010C 10FB          JMP  RNDG1
0138 010E A045  RNDG2   A    R5,R1         SET UP ADDRESS REG
0139 0110 C141          MOV  R1,R5         SET UP REG FOR COMPARE
0140 0112 06A0          BL   @WTAD
```

```
      0114 0078'
0141  0116 0585          INC   R5
0142  0118 C0C4  RND3    MOV   R4,R3          5,4,3,2,1 READ DATA
0143  011A C2CF          MOV   R15,R11        POP RETURN REG
0144  011C 0585  WA05    INC   R5             AUTO INC FOR COMPARE
0145  011E D1A0          MOVB  @GRD,R6        DUMMY READ
      0120 9800
0146  0122 0603          DEC   R3             DONE
0147  0124 16FB          JNE   WA05           NOPE GO BACK
0148  0126 D1A0          MOVB  @GRA,R6        READ ADDRESS
      0128 9802
0149  012A 06C6          SWPB  R6
0150  012C 1000          NOP
0151  012E 1000          NOP
0152  0130 D1A0          MOVB  @GRA,R6        READ ADDRESS
      0132 9802
0153  0134 06C6          SWPB  R6             MSB FIRST
0154  0136 8185          C     R5,R6          GOOD
0155  0138 1601          JNE   ERROR          NO DISPLAY IT
0156  013A 045B          RT
0157  013C 070E  ERROR   SETO  R14            SET ERROR MASK
0158  013E 024A          ANDI  R10,>F8        STRIP BOTTOM 3 BITS
      0140 00F8
0159  0142 E28D          SOC   R13,R10        BAD GROM #
0160  0144 045B          RT
0161  0146 C38E  ERDSP   MOV   R14,R14        WRITE ADDRESS ERROR
0162  0148 1303          JEQ   D001           NO SKIP IT
0163  014A C80A          MOV   R10,@DSPLY     YES SO DISPLAY IT
      014C F800
0164  014E 1004          JMP D002
0165  0150 024A  D001    ANDI  R10,>5F        CRC ERROR
      0152 005F
0166  0154 C80A          MOV   R10,@DSPLY     DISPLAY IT
      0156 F800
0167  0158 1000  D002    NOP                  NEXT TEST
0168               *
0169               *       THE TEST BRANCHES TO THE WRITE AND READ
0170               *  ROUTINES.  R3 HOLDS OFFSET FOR THE NEXT PASS,
0171               *  R1 HOLDS WRITE DATA,R4 HOLDS TEMPORARY WRITE
0172               *  DATA.
0173               *
0174               *
0175  015A 0208          LI    R8,>100              COUNT
      015C 0100
0176  015E 024A          ANDI  R10,>37              SET MASK FOR PASS
      0160 0037
0177  0162 04C3          CLR   R3
0178  0164 04C1  W001    CLR   R1                   CLEAR FOR THE WRITE
0179  0166 A043          A     R3,R1                ADD OFFSET
0180  0168 06A0          BL    @WRITE               WRITE ONE PASS
      016A 0192'
0181  016C C101          MOV   R1,R4                SAVE THE WRITE DATA
0182  016E 04C1          CLR   R1                   CLEAR FOR THE READ
0183  0170 A043          A     R3,R1                ADD OFFSET
0184  0172 06A0  R001    BL    @READ                READ ONE PASS
      0174 01AA'
0185  0176 C044          MOV   R4,R1                GET WRITE DATA FOR NEXT PASS
0186  0178 06A0  W002    BL    @WRITE               WRITE SECOND PASS
      017A 0192'
0187  017C C044          MOV   R4,R1
```

```
0188 017E 06A0  R002   BL   @READ              READ NEXT PASS
     0180 01AA'
0189 0182 0223         AI   R3,>101            INCREMENT OFFSET
     0184 0101
0190 0186 1701         JNC  TOPCHK             ARE WE OVER THE TOP
0191 0188 04C3         CLR  R3                 YEP SO CLEAR R3
0192 018A 0608  TOPCHK DEC  R8                 ARE WE DONE
0193 018C 16EB         JNE  W001               YES SO GO BACK
0194 018E 0460         B    @STOP
     0190 01DE'
0195              *
0196              *      THIS ROUTINE WRITES 128 BYTES IN
0197              *  CONSOLE RAM THEN RETURNS.
0198              *
0199 0192 0209  WRITE  LI   R9,RAMUT           START OF CONSOLE RAM
     0194 8300
0200 0196 0202         LI   R2,>80             LOAD COUNT
     0198 0080
0201 019A CE41  WRIT1  MOV  R1,*R9+            WRITE DATA
0202 019C 0221         AI   R1,>101            INCREMENT WRITE DATA
     019E 0101
0203 01A0 1701         JNC  WRIT2              CHECK FOR CARRY
0204 01A2 04C1         CLR  R1                 CARRY SO CLEAR R1
0205 01A4 0602  WRIT2  DEC  R2                 DECREMENT COUNT
0206 01A6 16F9         JNE  WRIT1              IF NOT DONE WRITE SOME MORE
0207 01A8 045B         RT
0208              *
0209              *      THIS ROUTINE READS 128 WORDS FROM
0210              *  CONSOLE RAM AND CHECKS THE  AGAINST
0211              *  GOOD WRITE DATA FOR ERRORS.
0212              *
0213 01AA 0209  READ   LI   R9,RAMUT           START OF CONSOLE RAM
     01AC 8300
0214 01AE 0202         LI   R2,>80             LOAD COUNT
     01B0 0080
0215 01B2 C179  READ1  MOV  *R9+,R5            GET DATA FOR COMPARISON
0216 01B4 8141         C    R1,R5              IS IT GOOD
0217 01B6 1301         JEQ  READ2              YEP SO SKIP IT
0218 01B8 1007         JMP  ERR1               DISPLAY ERROR
0219 01BA 0221  READ2  AI   R1,>101            INCREMENT GOOD DATA
     01BC 0101
0220 01BE 1701         JNC  READ3              CHECK FOR CARRY
0221 01C0 04C1         CLR  R1                 CARRY SO CLEAR R1
0222 01C2 0602  READ3  DEC  R2                 DEC COUNT
0223 01C4 16F6         JNE  READ1              IF NOT DONE READ SOME MORE
0224 01C6 045B         RT
0225 01C8 9141  ERR1   CB   R1,R5              IS IT THE MSB OR LSB
0226 01CA 1306         JEQ  DSP                LSB SO SKIP IT
0227 01CC 022A         AI   R10,>40            HIGH ORDER BYTE
     01CE 0040
0228 01D0 06C5         SWPB R5
0229 01D2 06C1         SWPB R1
0230 01D4 9141         CB   R1,R5
0231 01D6 1302         JEQ  DSP1
0232 01D8 022A  DSP    AI   R10,>8             LIGHT LOW ORDER NIBBLE
     01DA 0008
0233 01DC 10EE  DSP1   JMP  READ2
0234 01DE 022A  STOP   AI   R10,>80            SET EOT BIT
     01E0 0080
0235 01E2 C80A         MOV  R10,@DSPLY
```

```
      01E4 F800
 0236 01E6 10FF          JMP   $                ALL DONE SO JUMP HERE
NO ERRORS,      NO WARNINGS
```

LABEL	VALUE	DEFN	REFERENCES
$	01E6'		0048 0073 0080 0236
CPURAM	F000	0018	0022
CRC	00F6'	0129	0127
CRCALC	0088'	0080	0052 0057 0062
CRLP	008E'	0083	0096
CRRET	00B8'	0101	0098
DO01	0150'	0165	0162
DO02	0158'	0167	0164
DEL1	0010'	0032	0035
DELAY	000C'	0031	
DSP	01D8'	0232	0226
DSP1	01DC'	0233	0231
DSPLY	F800	0016	0024 0068 0071 0163 0166 0235
EOT	006E'	0070	0066
ERDSP	0146'	0161	0121
ERR1	01C8'	0225	0218
ERROR	013C'	0157	0155
GRA	9802	0013	0148 0152
GRCR	0028'	0048	
GRD	9800	0015	0084 0145
GWA	9C02	0014	0074 0077
IOT1	0020'	0038	0040
IOTAD	F400	0019	0033 0038
RO01	0172'	0184	
RO02	017E'	0188	
R1	0001		0031 0034 0037 0038 0050 0055 0060 0074 0075 0077 0110 0113 0116 0138 0139 0178 0179 0181 0182 0183 0185 0187 0201 0202 0204 0216 0219 0221 0225 0229 0230
R10	000A		0023 0024 0067 0068 0070 0071 0158 0159 0163 0165 0166 0176 0227 0232 0234 0235
R11	000B		0097 0099 0130 0143
R12	000C		0054 0059 0064 0070
R13	000D		0111 0114 0117 0159
R14	000E		0109 0157 0161 0161
R15	000F		0130 0143
R2	0002		0200 0205 0214 0222
R3	0003		0032 0033 0036 0039 0049 0065 0065 0101 0142 0146 0177 0179 0183 0189 0191
R4	0004		0108 0119 0142 0181 0185 0187
R5	0005		0132 0133 0134 0136 0138 0139 0141 0144 0154 0215 0216 0225 0228 0230
R6	0006		0082 0095 0145 0148 0149 0152 0153 0154
R7	0007		0083 0084 0085 0086 0087 0088 0089 0090 0091 0092 0093
R8	0008		0081 0085 0086 0088 0091 0093 0094 0097 0175 0192
R9	0009		0122 0122 0124 0125 0127 0132 0199 0201 0213 0215
RAMUT	8300	0017	0199 0213
READ	01AA'	0213	0184 0188
READ1	01B2'	0215	0223
READ2	01BA'	0219	0217 0233
READ3	01C2'	0222	0220
RND	00E6'	0122	0131
RND1	00EC'	0125	0123
RND2	00F4'	0128	0126
RND3	0118'	0142	
RNDG	00F8'	0130	0112 0115 0118
RNDG1	0104'	0134	0137

RNDG2	010E'	0138	0135			
STOP	01DE'	0234	0194			
TOPCHK	018A'	0192	0190			
W001	0164'	0178	0193			
W002	0178'	0186				
WA01	00C2'	0110	0120			
WA02	00CC'	0113				
WA03	00D6'	0116				
WA05	011C'	0144	0147			
WHEX	F018	0020				
WRAD	008C'	0108	0069	0072		
WRIT1	019A'	0201	0206			
WRIT2	01A4'	0205	0203			
WRITE	0192'	0199	0180	0186		
WTAD	0078'	0073	0051	0056	0061	0140

C. D. HERE IS THE COMPARE PGM. TO USE IT TO COMPARE TWO GROMS:

1) READ IN "VGCOMP" FROM TAPE

A) INSERT TAPE IN LEFT-HAND READER

B) PRESS "REWIND"

C) PRESS "LOAD/FF". TAPE SHOULD MOVE FWD THEN STOP; READY LIGHT GOES ON.

D) PRESS THE "RECORD/PLAYBACK" SWITCH DOWN

E) IN TIBUG, TYPE "L"; A space prints, then press "0".

F) IMMEDIATELY, PRESS THE "CONT/START" SWITCH UP
TAPE SHOULD READ T WHEN IT STOPS, TIBUG RESPONDS WITH "VGCOMP!"

2) TRANSFER GROM TO VDP

USE THE N COMMAND: N G<Address IN GROM> 0 V<Address in VDP>(CRLF)
- transfers from GROM to VDP.

3) Compare VDP memory to GROM.

A) TYPE "R" - SET P{C} TO >C100.

B) TYPE E 0.

```
SOURCE ACCESS NAME=      HC2.ALAN.SRC.COMPARE
OBJECT ACCESS NAME=      HC2.ALAN.OBJ.COMPARE
LISTING ACCESS NAME=     HC2.ALAN.LST.COMPARE
ERROR ACCESS NAME=
OPTIONS=                 XREF
MACRO LIBRARY PATHNAME=
```

```
0001                        IDT 'VGCOMP'
0002                *       COMPARES VDP MEMORY BLOCKS TO GROM.
0003                *       DESIGNED TO RUN ON TIBUG >C100 TO >C3FF.
0004                *
0005                *       COMPARES BYTE BY BYTE
0006                ****    EQUATES TO WORK WITH TIBUG:
0007        F004    READ    EQU  >F004          READ ASCII FROM TERMINAL
0008        F008    WRIT    EQU  >F008          WRITE ASCII TO TERMINAL
0009        F00C    MESG    EQU  >F00C          WRITE MESSAGE TO TERMINAL
0010        F010    RHEX    EQU  >F010          READ HEX FROM TERMINAL
0011        F014    WHX1    EQU  >F014          WRITE ONE HEX CHARACTER TO TERMINAL
0012        F018    WHEX    EQU  >F018          WRITE NUMBER TO TERMINAL IN HEX
0013        F020    MONTOP  EQU  >F020
0014                *
0015                ****    PROGRAM BELONGS AT >C100
0016 C100           AORG >C100
0017 C100 02E0      LWPI >C080
     C102 C080
0018                *
0019                ****    INFORM THAT AT START OF PROGRARM.
0020 C104 0200      START   LI R0,STRTM
     C106 C1D5
0021 C108 0420      BLWP @MESG
     C10A F00C
0022 C10C 0420      STRT1   BLWP @RHEX          GET START ADDRESS (IN VDP)
     C10E F010
0023 C110 C10C      DATA STRT1,ERROR
     C112 C1C6
0024 C114 C080      MOV R0,R2               R2: VDP ADDRESS.
0025 C116 0420      STRT2   BLWP @RHEX          GET START ADDRESS (IN GROM)
     C118 F010
0026 C11A C116      DATA STRT2,ERROR
     C11C C1C6
0027 C11E C0C0      MOV R0,R3               R3: GROM ADDRESS.
0028 C120 0420      STRT3   BLWP @RHEX
     C122 F010
0029 C124 C120      DATA STRT3,ERROR
     C126 C1C6
0030 C128 C100      MOV R0,R4               R4: BYTE COUNT.
0031 C12A 0200      LI R0,CRLF              OUTPUT CRLF
     C12C C1D2
0032 C12E 0420      BLWP @MESG
     C130 F00C
0033 C132 04C7      CLR R7                  CLEAR ERROR COUNT
0034                *
0035                ****    LOAD VDP ADDRESS (FOR THE LAST TIME)
0036 C134 06C2      SWPB R2
0037 C136 D802      MOVB R2,>8C02
     C138 8C02
0038 C13A 06C2      SWPB R2
0039 C13C C002      MOV R2,R0
0040 C13E 0260      ORI R0,>4000
     C140 4000
0041 C142 D800      MOVB R0,>8C02
     C144 8C02
0042 C146 D020      MOVB >8800,R0           GET GARBAGE OUT OF WAY.
     C148 8800
0043                *
0044                ****    LOAD GROM ADDRESS FOR THE FIRST AND LAST TIME
0045 C14A D020      MOVB >9800,R0
```

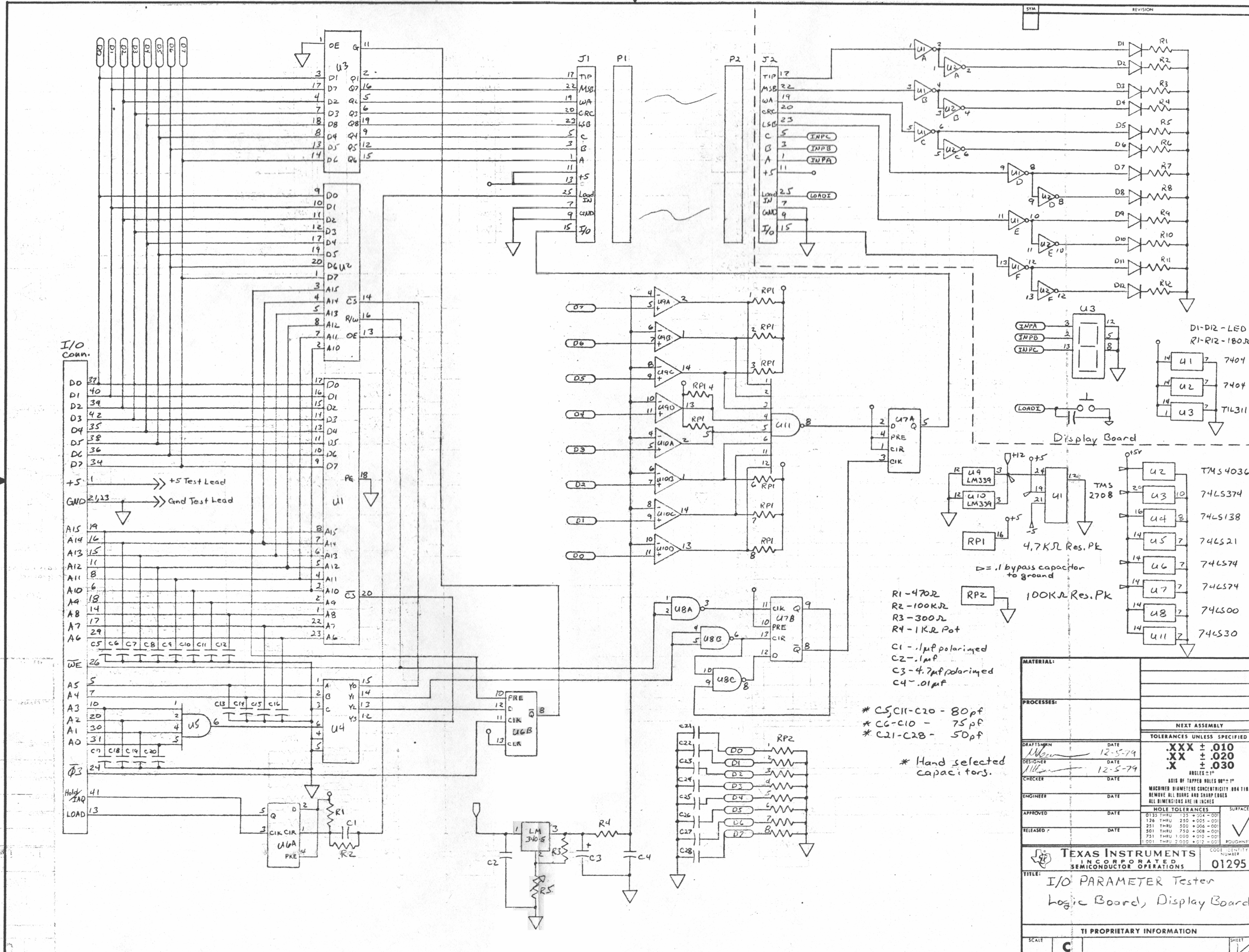
```
     C14C 9800
0046 C14E D803          MOVB R3,>9C02
     C150 9C02
0047 C152 06C3          SWPB R3
0048 C154 D803          MOVB R3,>9C02
     C156 9C02
0049 C158 06C3          SWPB R3
0050                 *
0051                 ****   COMPARE DATA
0052 C15A C160       COMPB  MOV >8800,R5       DATA IN VDP; AUTO INCR. INSIDE VDP
     C15C 8800
0053 C15E C1A0          MOV >9800,R6       DATA FROM GROM; AUTO INCR. IN GROM
     C160 9800
0054 C162 8185          C R5,R6            ARE THEY THE SAME?
0055 C164 160D          JNE COMBAD         NO. . . TELL USER.
0056 C166 0582       COMPI  INC R2             YES, INCREMENT VDP "ADDRESS"
0057 C168 0583          INC R3             AND GROM "ADDRESS"
0058 C16A 0604          DEC R4             FINISHED COMPARING?
0059 C16C 16F6          JNE COMPB
0060                 *
0061                 ****   FINISHED.  PUBLISH ERROR COUNT.
0062 C16E C007          MOV R7,R0
0063 C170 0420          BLWP @WHEX
     C172 F018
0064 C174 0200          LI R0,ERRCNT
     C176 C227
0065 C178 0420          BLWP @MESG
     C17A F00C
0066 C17C 0460          B @MONTOP          FINE!
     C17E F020
0067                 *
0068                 ****   INFOMR USER ABOUT DIFFERENCES IN DATA.
0069 C180 0200       COMBAD LI R0,BADDAT
     C182 C1EB
0070 C184 0420          BLWP @MESG
     C186 F00C
0071 C188 C003          MOV R3,R0
0072 C18A 0420          BLWP @WHEX
     C18C F018
0073 C18E 0200          LI R0,BADA
     C190 C207
0074 C192 0420          BLWP @MESG
     C194 F00C
0075 C196 C005          MOV R5,R0
0076 C198 06A0          BL @OUT2           OUTPUT VDP DATA
     C19A C1B8
0077 C19C 0200          LI R0,BADCON
     C19E C212
0078 C1A0 0420          BLWP @MESG
     C1A2 F00C
0079 C1A4 C006          MOV R6,R0
0080 C1A6 06A0          BL @OUT2           OUTPUT GROM DATA
     C1A8 C1B8
0081 C1AA 0200          LI R0,CRLF
     C1AC C1D2
0082 C1AE 0420          BLWP @MESG
     C1B0 F00C
0083 C1B2 0587          INC R7             INCREMENT ERROR COUNT
0084 C1B4 0460          B COMPI            CONTINUE LOOKING AT DATA.
     C1B6 C166
```

```
0085                    *
0086                    ****    TYPE OUT ONLY 2 HEX DIGITS.
0087 C1B8 0BC0  OUT2    SRC RO,12
0088 C1BA 0420          BLWP @WHX1
     C1BC F014
0089 C1BE 0BC0          SRC RO,12
0090 C1C0 0420          BLWP @WHX1
     C1C2 F014
0091 C1C4 045B          RT
0092 C1C6 0200  ERROR   LI RO,QUIT
     C1C8 C21B
0093 C1CA 0420          BLWP @MESG
     C1CC F00C
0094 C1CE 0460          B @MONTOP
     C1D0 F020
0095      000D  CR      EQU  >0D
0096      000A  LF      EQU  >0A
0097 C1D2   0D  CRLF    BYTE CR,LF,0
     C1D3   0A
     C1D4   00
0098 C1D5   0D  STRTM   BYTE CR,LF
     C1D6   0A
0099 C1D7   43          TEXT 'COMP.  VDP TO GROM:  '
     C1D8   4F
     C1D9   4D
     C1DA   50
     C1DB   2E
     C1DC   20
     C1DD   56
     C1DE   44
     C1DF   50
     C1E0   20
     C1E1   54
     C1E2   4F
     C1E3   20
     C1E4   47
     C1E5   52
     C1E6   4F
     C1E7   4D
     C1E8   3A
     C1E9   20
0100 C1EA   00          BYTE 0
0101 C1EB   44  BADDAT  TEXT 'DIFFERENCE.   GROM ADDRESS= '
     C1EC   49
     C1ED   46
     C1EE   46
     C1EF   45
     C1F0   52
     C1F1   45
     C1F2   4E
     C1F3   43
     C1F4   45
     C1F5   2E
     C1F6   20
     C1F7   20
     C1F8   47
     C1F9   52
     C1FA   4F
     C1FB   4D
     C1FC   20
```

```
       C1FD   41
       C1FE   44
       C1FF   44
       C200   52
       C201   45
       C202   53
       C203   53
       C204   3D
       C205   20
0102   C206   00          BYTE 0
0103   C207   20  BADA    TEXT ' VDP HAD:  '
       C208   56
       C209   44
       C20A   50
       C20B   20
       C20C   48
       C20D   41
       C20E   44
       C20F   3A
       C210   20
0104   C211   00          BYTE 0
0105   C212   20  BADCON TEXT '  GROM:  '
       C213   20
       C214   47
       C215   52
       C216   4F
       C217   4D
       C218   3A
       C219   20
0106   C21A   00          BYTE 0
0107   C21B   2A  QUIT    TEXT '**ERROR**'
       C21C   2A
       C21D   45
       C21E   52
       C21F   52
       C220   4F
       C221   52
       C222   2A
       C223   2A
0108   C224   0D          BYTE CR,LF,0
       C225   0A
       C226   00
0109   C227   20  ERRCNT TEXT ' ERRORS FOUND. '
       C228   45
       C229   52
       C22A   52
       C22B   4F
       C22C   52
       C22D   53
       C22E   20
       C22F   46
       C230   4F
       C231   55
       C232   4E
       C233   44
       C234   2E
0110   C235   0D          BYTE CR,LF,0
       C236   0A
       C237   00
NO ERRORS,      NO WARNINGS
```

BADA	C207	0103	0073								
BADCON	C212	0105	0077								
BADDAT	C1EB	0101	0069								
COMBAD	C180	0069	0055								
COMPB	C15A	0052	0059								
COMPI	C166	0056	0084								
CR	000D	0095	0097	0098	0108	0110					
CRLF	C1D2	0097	0031	0081							
ERRCNT	C227	0109	0064								
ERROR	C1C6	0092	0023	0026	0029						
LF	000A	0096	0097	0098	0108	0110					
MESG	F00C	0009	0021	0032	0065	0070	0074	0078	0082	0093	
MONTOP	F020	0013	0066	0094							
OUT2	C1B8	0087	0076	0080							
QUIT	C21B	0107	0092								
R0	0000		0020	0024	0027	0030	0031	0039	0040	0041	0042
			0045	0062	0064	0069	0071	0073	0075	0077	0079
			0081	0087	0089	0092					
R2	0002		0024	0036	0037	0038	0039	0056			
R3	0003		0027	0046	0047	0048	0049	0057	0071		
R4	0004		0030	0058							
R5	0005		0052	0054	0075						
R6	0006		0053	0054	0079						
R7	0007		0033	0062	0083						
READ	F004	0007									
RHEX	F010	0010	0022	0025	0028						
START	C104	0020									
STRT1	C10C	0022	0023								
STRT2	C116	0025	0026								
STRT3	C120	0028	0029								
STRTM	C1D5	0098	0020								
WHEX	F018	0012	0063	0072							
WHX1	F014	0011	0088	0090							
WRIT	F008	0008									

```
100 GO SUB 130
110 GO TO 100
120 REM  ARRAY TEST
130 DIM A$(2,2,2),B$(2,2,2)
140 CALL CLEAR
150 FOR X=1 TO 2
160 FOR Y=1 TO 2
170 FOR Z=1 TO 2
180 READ A$(X,Y,Z)
190 READ B$(X,Y,Z)
200 NEXT Z
210 NEXT Y
220 NEXT X
230 READ E$
240 READ I
250 ON I GOTO 270
260 STOP
270 PRINT E$
280 PRINT
290 GO SUB 1050
300 FOR X=1 TO 2
310 FOR Y=1 TO 2
320 FOR Z=1 TO 2
330 D$(I)=A$(X,Y,Z)&B$(X,Y,Z)
340 I=I+1
350 NEXT Z
360 NEXT Y
370 NEXT X
380 FOR X=1 TO 8
390 D$(X)=SEG$(D$(X),2,1)
400 C(X)=(ASC(D$(X)))
410 NEXT X
420 DATA 3," ",4,(,4,0,5,8
430 DATA 6,@,7,H,8,P,8,X
440 DATA HOME COMPUTER STRING
RAY TEST-8K,1
450 FOR I=1 TO 48
460 PRINT "COLOR CHECK";
470 NEXT I
480 GO SUB 1620
490 COL=0
500 FORE=2
510 BACK=16
520 CALL CLEAR
530 CALL COLOR(1,FORE,BACK)
540 CALL HCHAR(1,1,32,768)
550 PRINT BACK
560 GO SUB 1050
570 BACK=BACK-1
580 IF BACK>0 THEN 520
590 BACK=1
600 FORE=2
610 CALL CHAR(32,"FFC1A2948894A2
C1")
620 GO SUB 1620
630 CALL CHAR(32,"")
640 CALL CLEAR
650 FOR I=1 TO 20
660 PRINT "CHARACTER CHECK";
670 NEXT I
680 GO SUB 1620
690 CALL CLEAR
700 SET=0
710 SET=SET+1
720 CALL COLOR(SET,13,4)
730 IF SET<8 THEN 710
740 X=-2
750 FOR Y=1 TO 8
760 X=X+3
770 CALL HCHAR(X,1,C(Y),96)
780 NEXT Y
790 GO SUB 1050
800 FOR X=1 TO 8
810 C(X)=C(X)+1
820 NEXT X
830 IF C(1)<40 THEN 740
840 GO SUB 1160
850 GO SUB 930
860 IF T=R+1 THEN 900
870 PRINT "BEGIN TEST CYCLE #";T
880 GO SUB 1620
890 RETURN
900 GO SUB 1100
910 PRINT "END OF TEST AFTER";R;
"CYCLES"
920 END
930 IF FLAG=1 THEN 1040
940 CALL CLEAR
950 PRINT "ENTER TEST CYCLES DES
IRED";
960 INPUT R
970 H=INT(R*5.907/60)
980 M=INT(R*5.907-(H*60))
990 CALL CLEAR
1000 PRINT "THANK YOU";"YOUR TES
T WILL REQUIRE ADD";H;"HOURS";M;
"MINUTES TO COMPLETE"
1010 FOR DEL=1 TO 2000
1020 NEXT DEL
1030 FLAG=1
1040 T=T+1
1050 FOR DEL=1 TO 75
1060 NEXT DEL
1070 GO SUB 1620
1080 CALL CLEAR
1090 RETURN
1100 RANDOMIZE
1110 FOR DELAY=1 TO 294 STEP 2
1120 PRINT INT(100*RND);
1130 NEXT DELAY
1140 GO SUB 1620
1150 RETURN
1160 CALL CLEAR
1170 FOR Z=1 TO 12
1180 CALL COLOR(1,1,1)
1190 PRINT
1200 NEXT Z
1210 PRINT "MATH FUNCTION CHECK"
1220 CALL COLOR(1,2,1)
1230 CALL SOUND(200,440,10,472,1
5,880,20)
1240 DATA -440,472,880
1250 RESTORE 1240
1260 READ S,P,V
1270 XX=INT(3*EXP(4))
1280 W=ABS(S)
1290 V=100*INT(LOG(W))
1300 CALL SOUND(200,W,10,P,15,V,
20)
1310 PRINT W;P;V
1320 XX=1000*COS(XX)
1330 IF SGN(XX)=-1 THEN 1360
1340 CALL SOUND(200,XX,10)
1350 GO TO 1300
1360 XX=ABS(XX)
1370 PRINT "X=A"
1380 XX=1000*SIN(XX)
1390 IF SGN(XX)=-1 THEN 1410
1400 GO TO 1340
1410 XX=ABS(XX)
1420 PRINT "X=B"
1430 XX=105+LOG(XX)
1440 IF XX<110 THEN 1480
1450 PRINT "X=C"
1460 CALL SOUND(200,XX,10)
1470 GO TO 1430
1480 YY=INT(100*TAN(XX))
1490 IF YY=-26 THEN 1510
1500 STOP
1510 GO SUB 1620
1520 CALL CLEAR
1530 FOR I=1 TO 10
1540 CALL COLOR(1,16,16)
1550 CALL COLOR(1,2,2)
1560 PRINT I;
1570 NEXT I
1580 GO SUB 1620
1590 CALL CLEAR
1600 RESTORE
1610 RETURN
1620 IF TFLAG=1 THEN 1670
1630 CALL KEY(0,A,B)
1640 IF A=32 THEN 1660
1650 RETURN
1660 TFLAG=1
1670 CALL KEY(0,A,B)
1680 IF B<1 THEN 1670
1690 TFLAG=0
1700 RETURN
```



MATERIAL:	
PROCESSES:	
NEXT ASSEMBLY	
DRAFTSMAN	DATE 12-5-79
DESIGNER	DATE 12-5-79
CHECKER	DATE
ENGINEER	DATE
APPROVED	DATE
RELEASED	DATE

TOLERANCES UNLESS SPECIFIED
.XXX ± .010
.XX ± .020
.X ± .030
ANGLES ± 1°
AXIS OF TAPPED HOLES 90° ± 1°
MACHINED DIAMETERS CONCENTRICITY .004 TIR
REMOVE ALL BURRS AND SHARP EDGES
ALL DIMENSIONS ARE IN INCHES

HOLE TOLERANCES / SURFACE ROUGHNESS

TEXAS INSTRUMENTS INCORPORATED SEMICONDUCTOR OPERATIONS

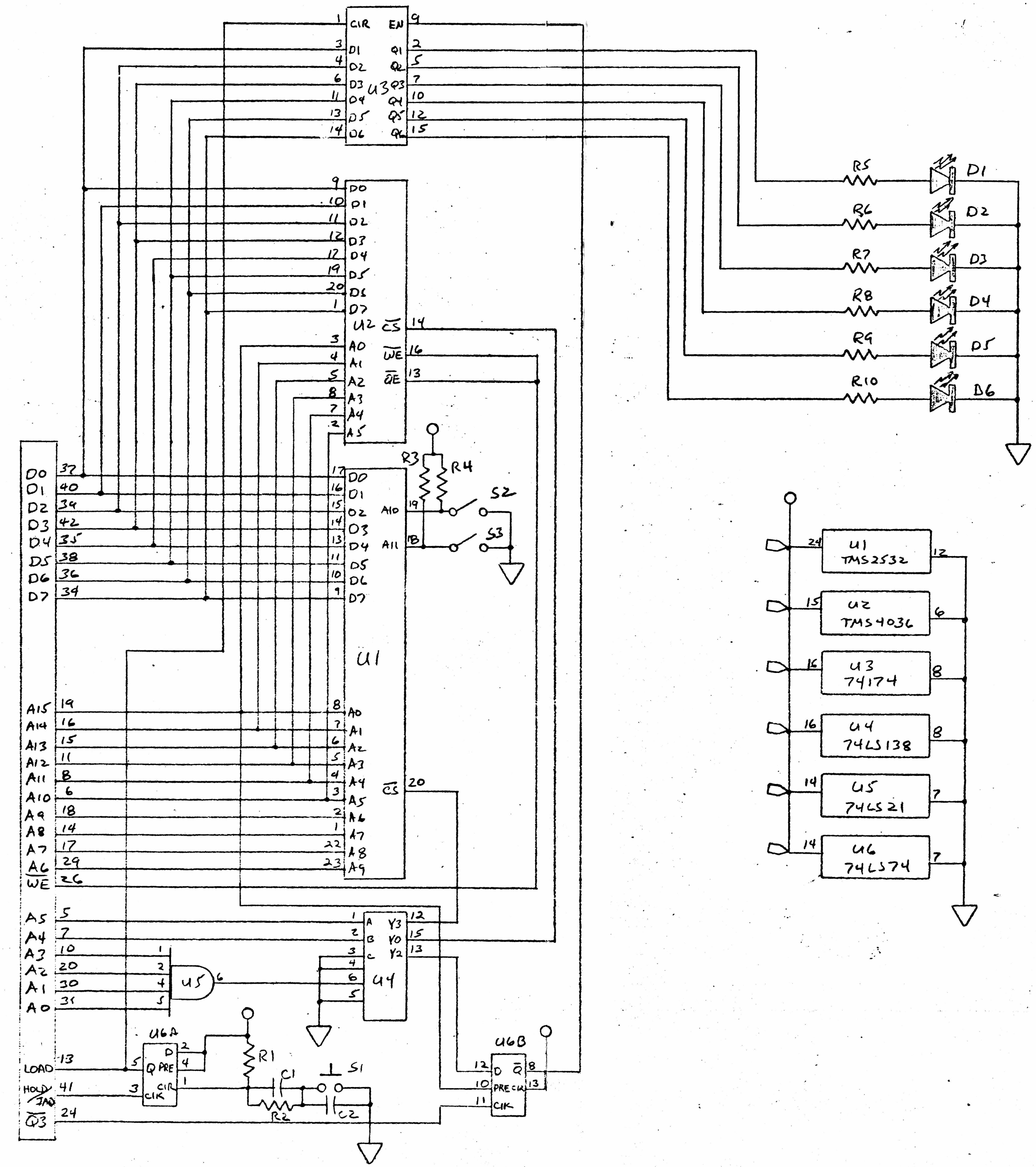
CODE IDENTITY NUMBER 01295

TITLE: I/O PARAMETER Tester
Logic Board, Display Board

TI PROPRIETARY INFORMATION

SCALE | SIZE C | SHEET 1/1

TI-4514G



D1-D6 - Red LED
R1 - 470Ω
R2 - 100KΩ
R3,R4 - 2.7KΩ
R5-R10 - 180Ω
C1,C2 - .1µf polarized cap
▷ = .1µf bypass capacitor

MATERIAL:	
PROCESSES:	NEXT ASSEMBLY
DRAFTSMAN [signature] DATE 12-6-79	TOLERANCES UNLESS SPECIFIED .XXX ± .010 .XX ± .020 .X ± .030
DESIGNER [signature] DATE 12-6-79	
CHECKER DATE	
ENGINEER DATE	
APPROVED DATE	HOLE TOLERANCES / SURFACE
RELEASED DATE	

TEXAS INSTRUMENTS INCORPORATED SEMICONDUCTOR OPERATIONS — 01295

TITLE: Bug Box Schematic

TI PROPRIETARY INFORMATION

SCALE — C — SHEET 1/1