# United States Patent [19]

**Walther**

[11] **3,766,370**

[45] **Oct. 16, 1973**


[54] **ELEMENTARY FLOATING POINT CORDIC FUNCTION PROCESSOR AND SHIFTER**


[75] Inventor: **John S. Walther**, Sunnyvale, Calif.

[73] Assignee: **Hewlett-Packard Company**, Palo Alto, Calif.



[22] Filed: **May 14, 1971**

[21] Appl. No.: **143,578**

[52] **U.S. Cl.**..... **235/156**, 235/164, 340/172.5

[51] **Int. Cl.**..... **G06f 7/00**, G06f 7/38

[58] **Field of Search**..... 235/156, 159, 160, 235/164, 197; 444/1; 340/172.5



[56] **References Cited**

UNITED STATES PATENTS

| | | | |
|---|---|---|---|
| 3,022,006 | 2/1962 | Alrich et al. | 235/160 |
| 3,134,091 | 5/1964 | Shugart | 235/156 X |
| 3,553,652 | 1/1971 | Hanson | 340/172.5 |

OTHER PUBLICATIONS

J. Volder, "The Cordic Trigonometric Computing Technique," IRE Trans. on Electronic Computers, Sept. 1959, pp. 330–334.



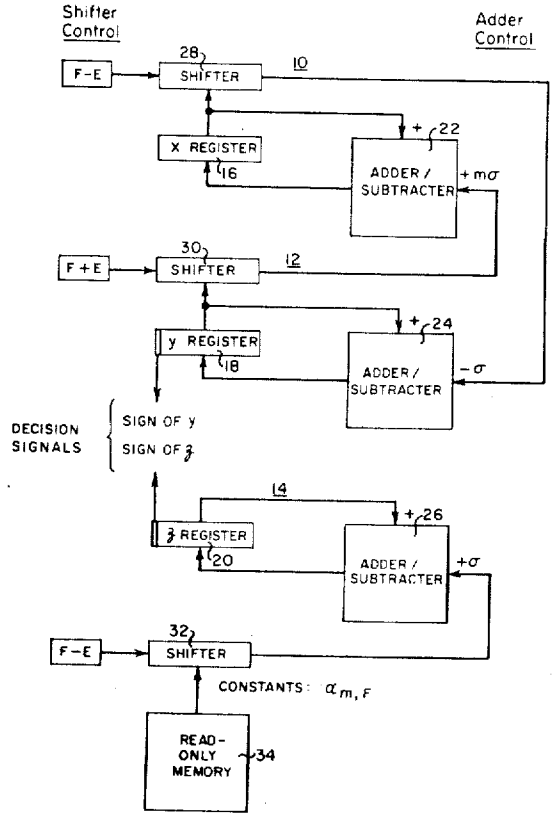
*Primary Examiner*—Charles E. Atkinson
*Assistant Examiner*—David H. Malzahn
*Attorney*—Roland I. Griffin



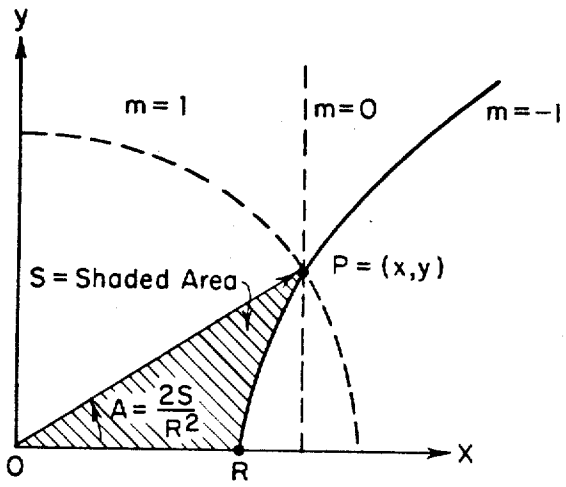
[57] **ABSTRACT**

Three arithmetic units including three shifters are operated in parallel and controlled by a microprogram stored in a read-only memory to provide an improved elementary function floating-point processor. The microprogram includes a set of routines for calculating 20 elementary functions including arithmetic, exponential, hyperbolic, logarithmic, square root, and trigonometric functions. Each shifter is capable of reading a fixed plural number of consecutive bits, beginning with any bit position, from an associated data storage register.


**5 Claims, 234 Drawing Figures**



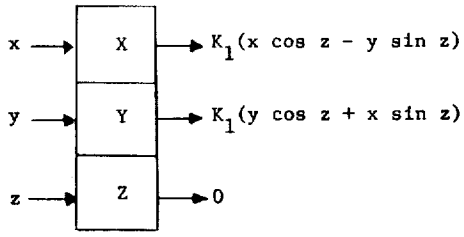

HARDWARE BLOCK DIAGRAM



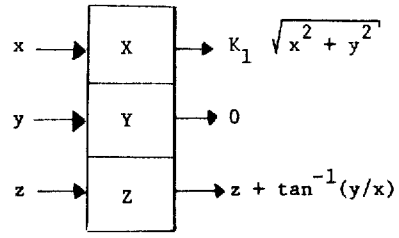

ANGLE A AND RADIUS R OF THE VECTOR P = (x,y)

FIG. 1

INPUT-OUTPUT FUNCTIONS FOR CORDIC MODES

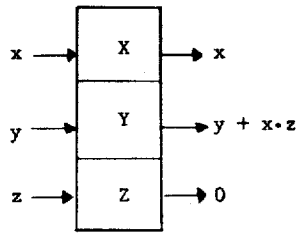



CIRCULAR (m=1), $z \to 0$

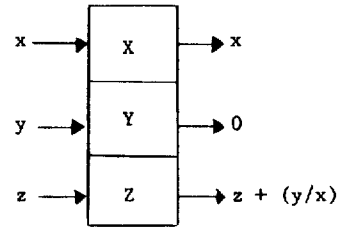



CIRCULAR (m=1), $A \to 0$

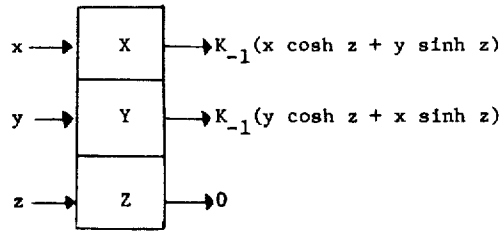
$$K_1 = \prod_{j=0}^{n-1} (1 + \delta_j^2)^{1/2} \quad \text{for n iterations}$$



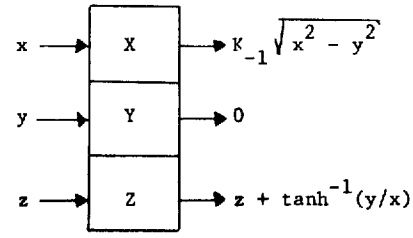

LINEAR (m=0), $z \to 0$




LINEAR (m=0), $A \to 0$



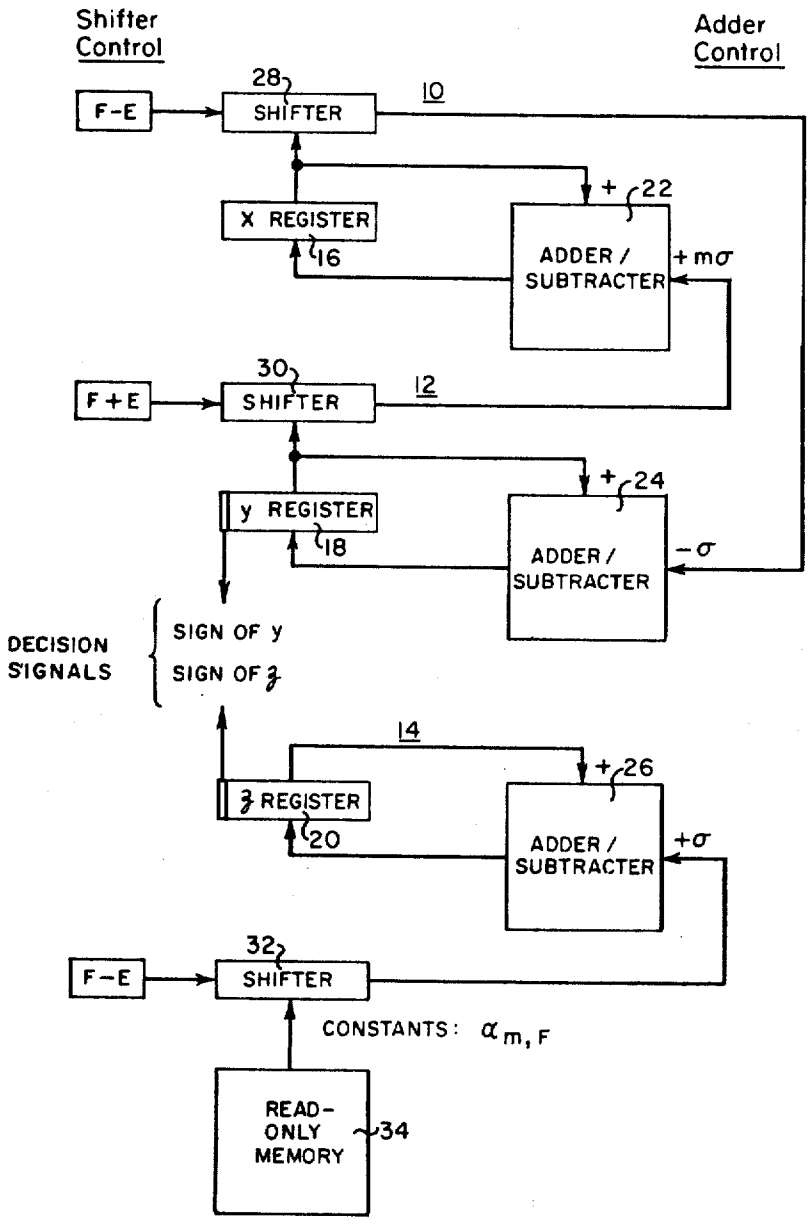

HYPERBOLIC (m = -1), $z \to 0$




HYPERBOLIC (m = -1), $A \to 0$

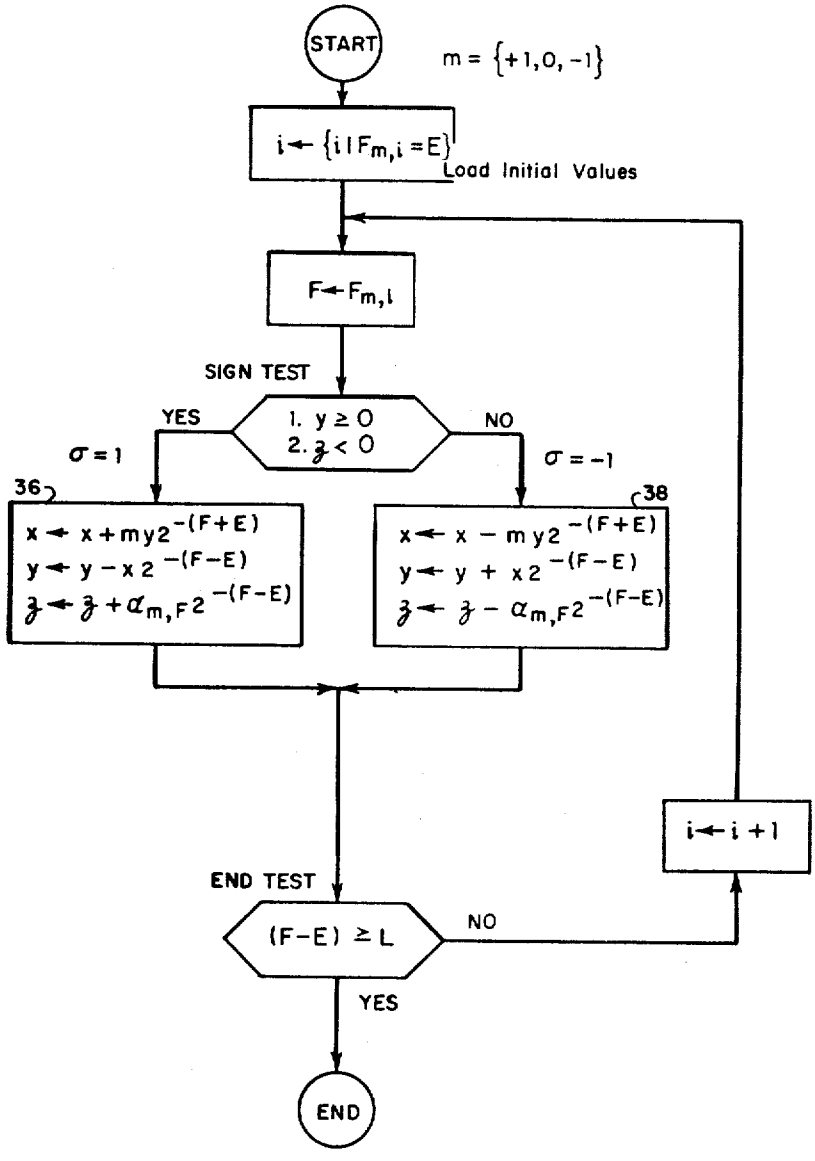
$$K_{-1} = \prod_{j=0}^{n-1} (1 - \delta_j^2)^{1/2} \quad \text{for n iterations}$$

FIG. 2




HARDWARE BLOCK DIAGRAM

FIG.3




FLOWCHART OF THE MICROPROGRAM CONTROL

FIG.4

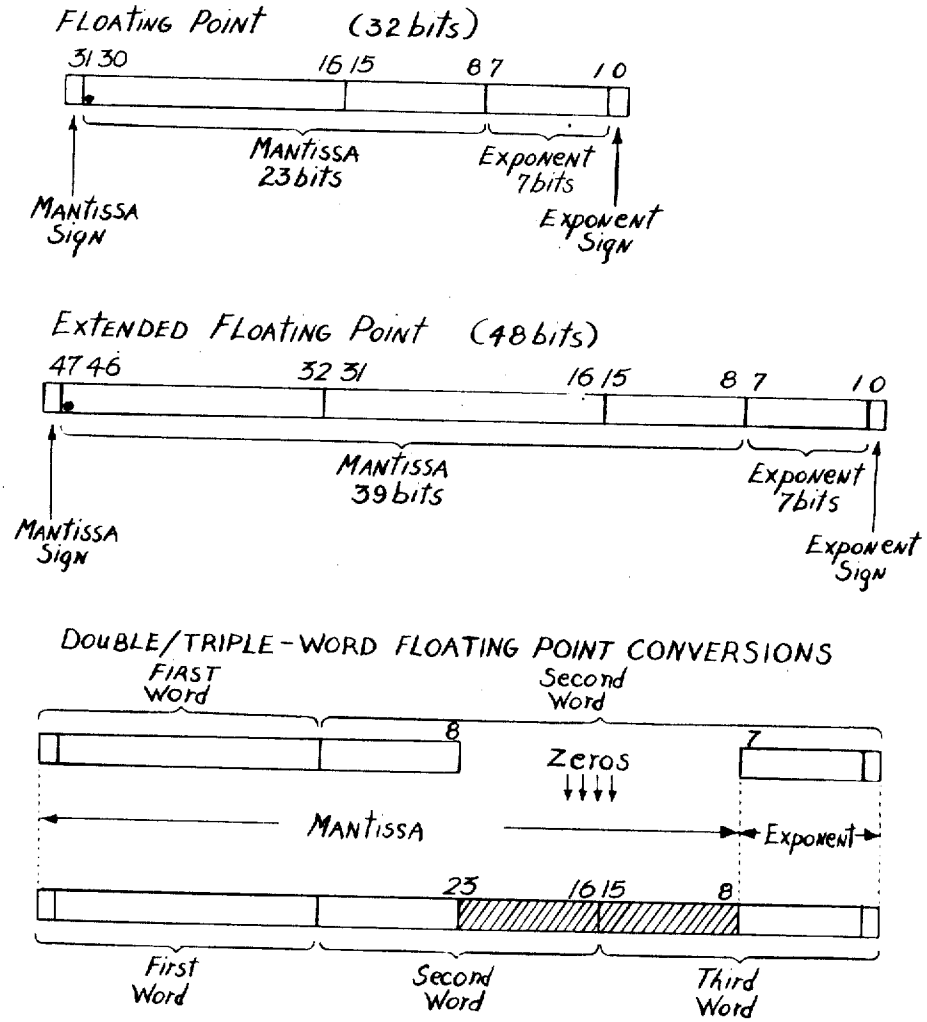

FLOATING POINT (32 bits)
31 30
16 15
8 7
1 0
MANTISSA 23 bits
EXPONENT 7 bits
MANTISSA SIGN
EXPONENT SIGN
EXTENDED FLOATING POINT (48 bits)
47 46
32 31
16 15
8 7
1 0
MANTISSA 39 bits
EXPONENT 7 bits
MANTISSA SIGN
EXPONENT SIGN
DOUBLE/TRIPLE-WORD FLOATING POINT CONVERSIONS
FIRST WORD
SECOND WORD
8
7
ZEROS
MANTISSA
EXPONENT
23
16 15
8
FIRST WORD
SECOND WORD
THIRD WORD

FIG. 5

RANGES OF EXTENDED FLOATING POINT NUMBERS

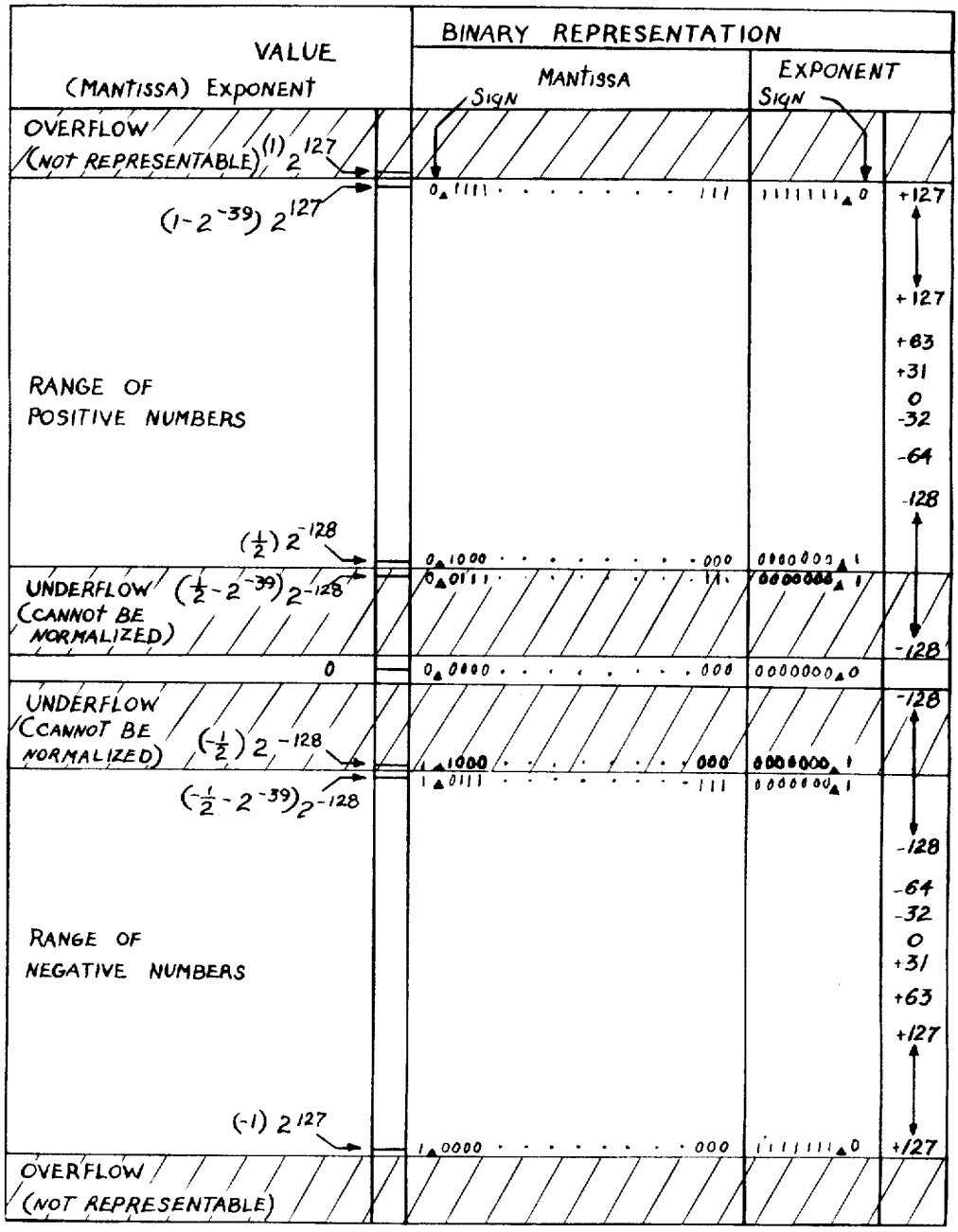



FIG. 6

BASIC FPP OPERATIONS

A. UNARY FUNCTION ROUTINES

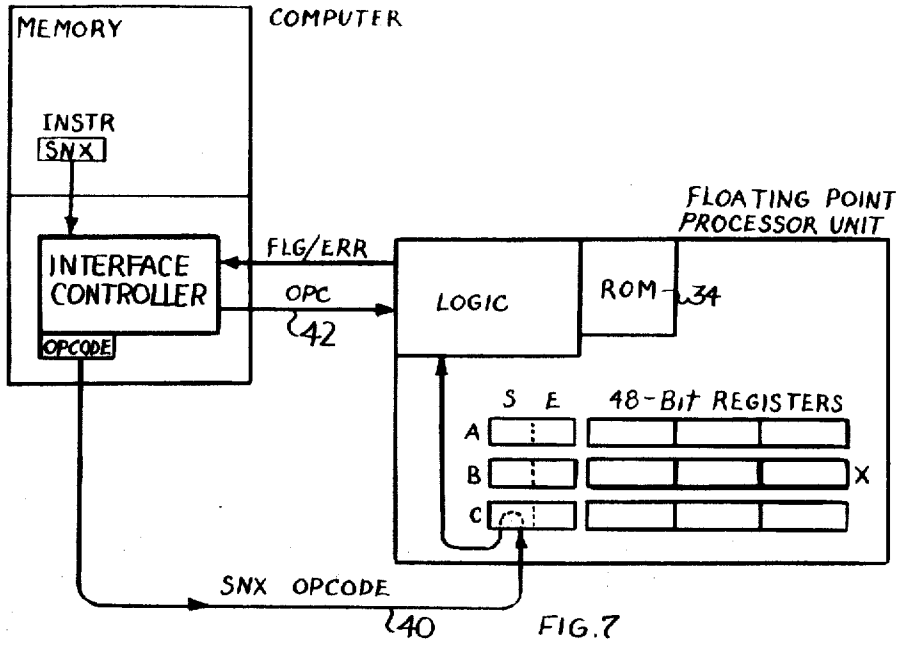



FIG. 7

B. BINARY FUNCTION ROUTINES

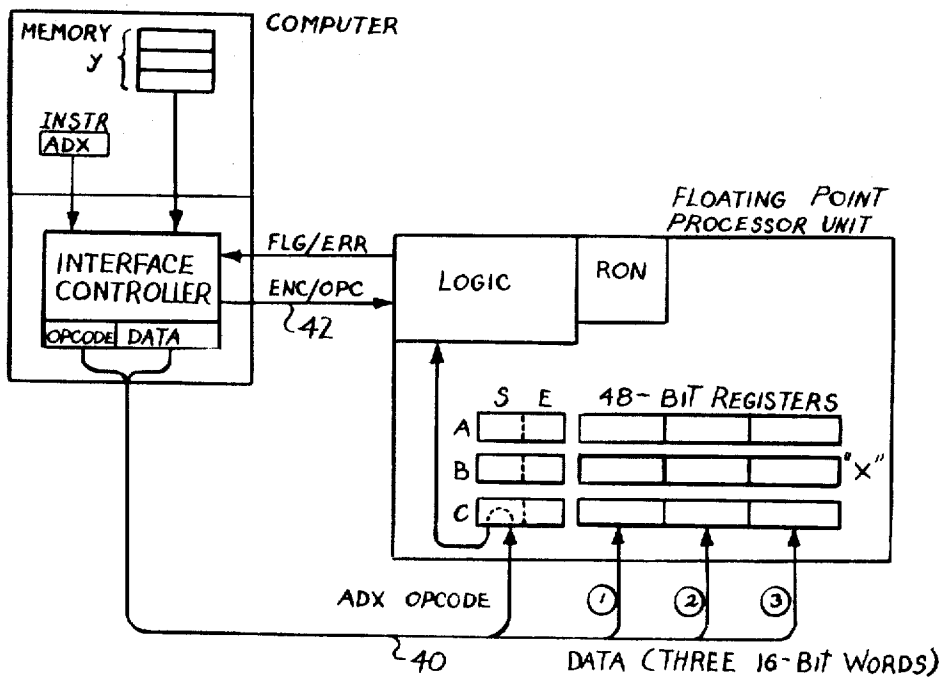



FIG. 8

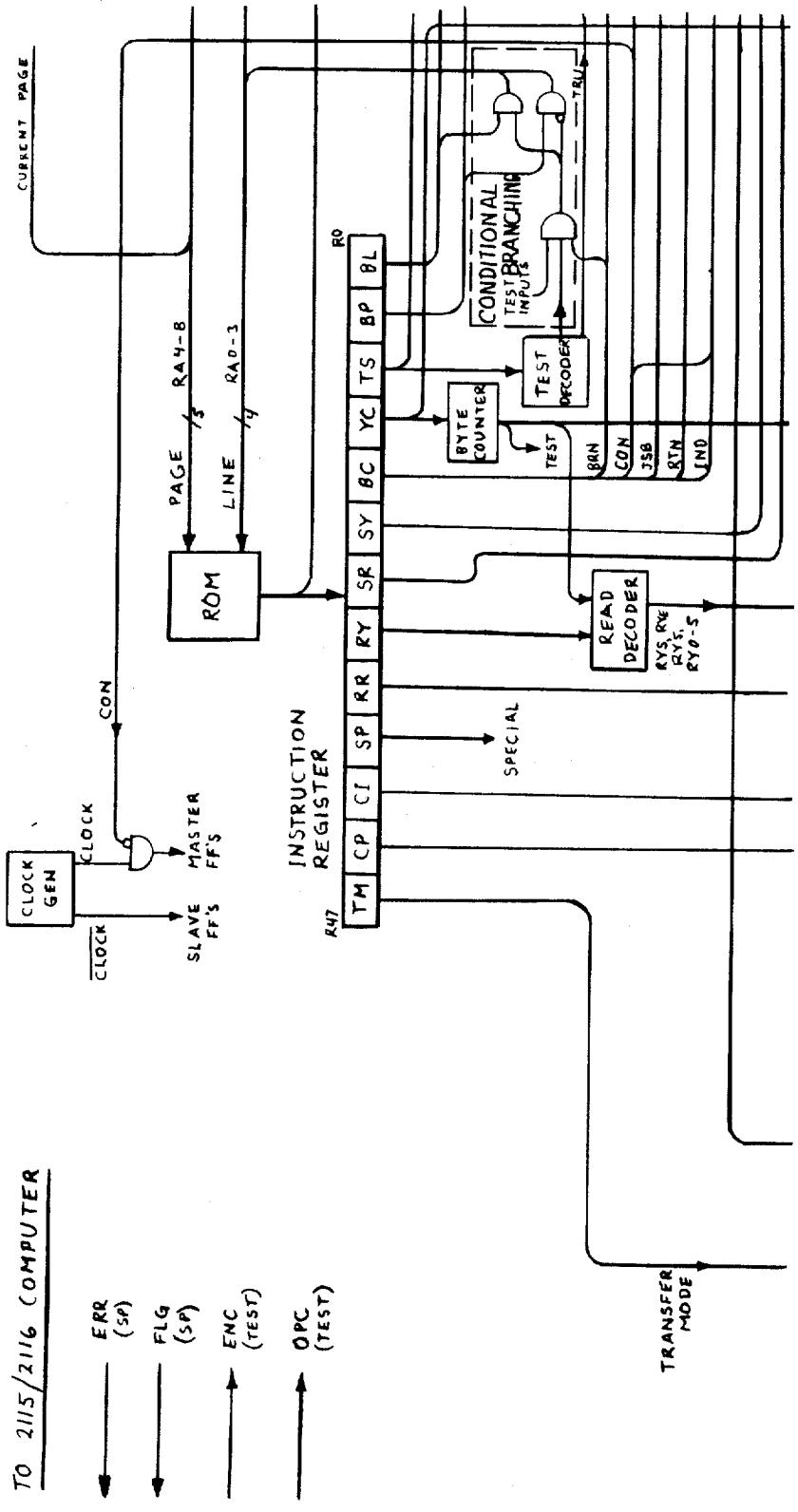

CURRENT PAGE
RA4-8
RA0-3
PAGE
LINE
ROM
CONDITIONAL BRANCHING
TEST INPUTS
TRU
TEST DECODER
BYTE COUNTER
TEST
BRN
CON
JSB
RTN
IND
READ DECODER
RY5, RY6, RY7, RY0-5
INSTRUCTION REGISTER
R0
R47
BL
BP
TS
YC
BC
SY
SR
RY
RR
SP
CI
CP
TM
SPECIAL
CON
CLOCK
CLOCK GEN
MASTER FF'S
SLAVE FF'S
TRANSFER MODE
TO 2115/2116 COMPUTER
ERR (SP)
FLG (SP)
ENC (TEST)
OPC (TEST)

FIG. 9A

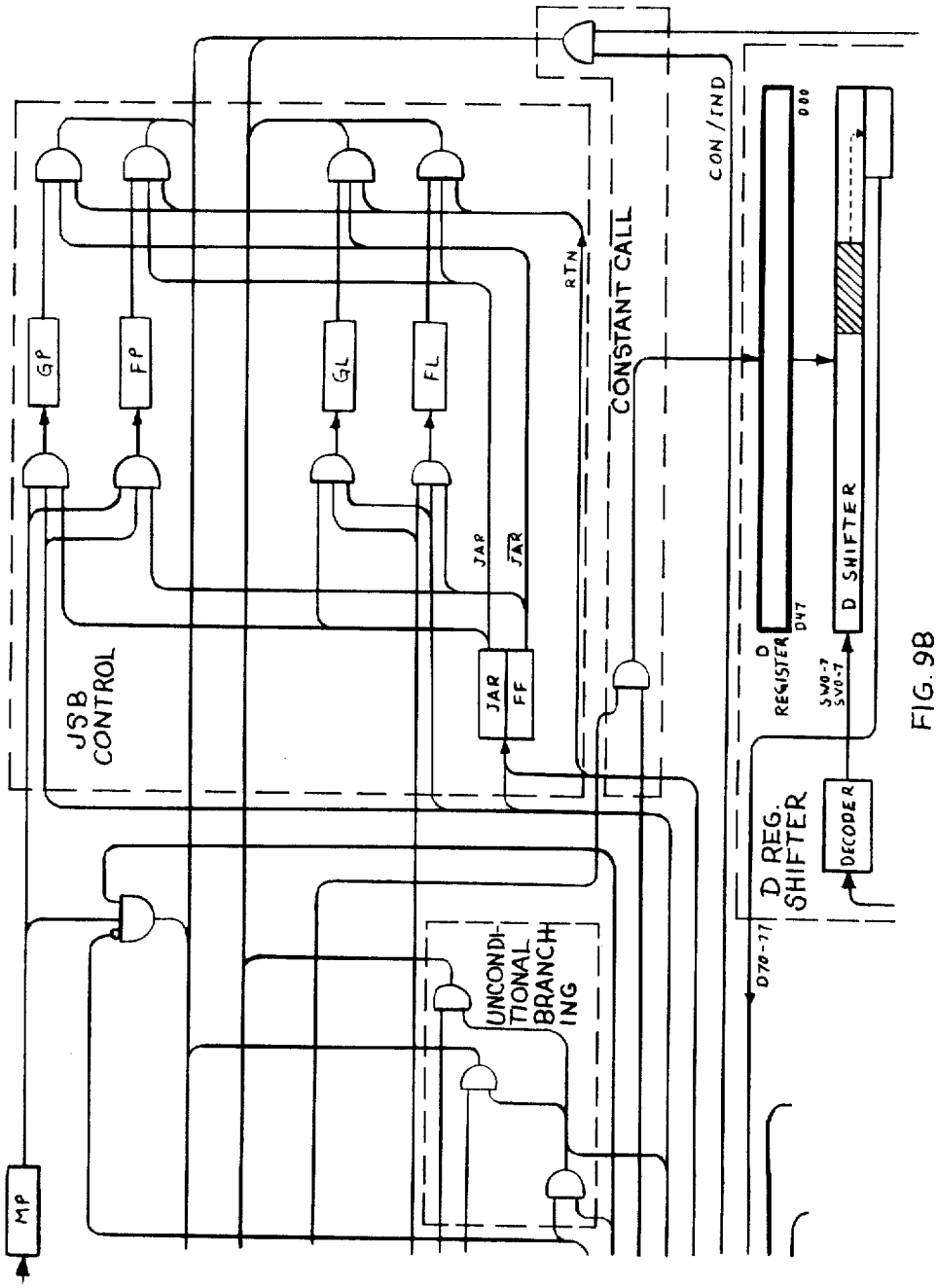

GP
FP
GL
FL
JAP
JAR
RTN
CONSTANT CALL
CON/IND
000
D REGISTER
D17
D SHIFTER
SW0-7
SV0-7
JSB CONTROL
JAR FF
UNCONDI-TIONAL BRANCH ING
D REG. SHIFTER
DECODER
D70-77
MP

FIG. 98

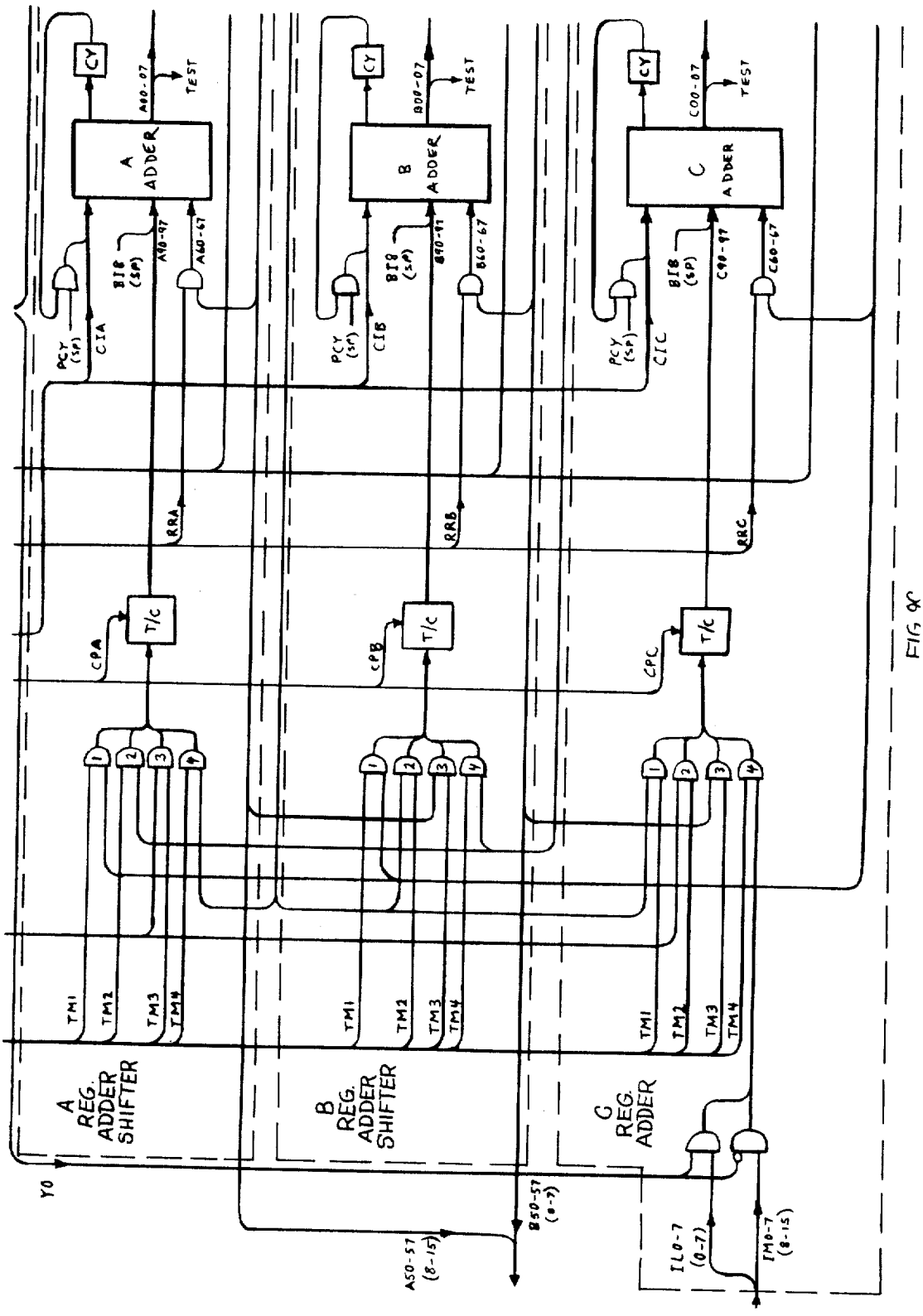

A ADDER
B ADDER
C ADDER
CY
TEST
A00-07
B00-07
C00-07
BIB (SP)
PCY (SP)
CIA
CIB
CIC
A10-47
A60-67
B10-47
B60-67
C00-47
C60-67
RRA
RRB
RRC
T/C
CPA
CPB
CPC
TM1
TM2
TM3
TM4
A REG. ADDER SHIFTER
B REG. ADDER SHIFTER
C REG. ADDER
Y0
AS0-57 (8-15)
BS0-57 (8-15)
IL0-7 (0-7)
IM0-7 (8-15)

FIG 9C

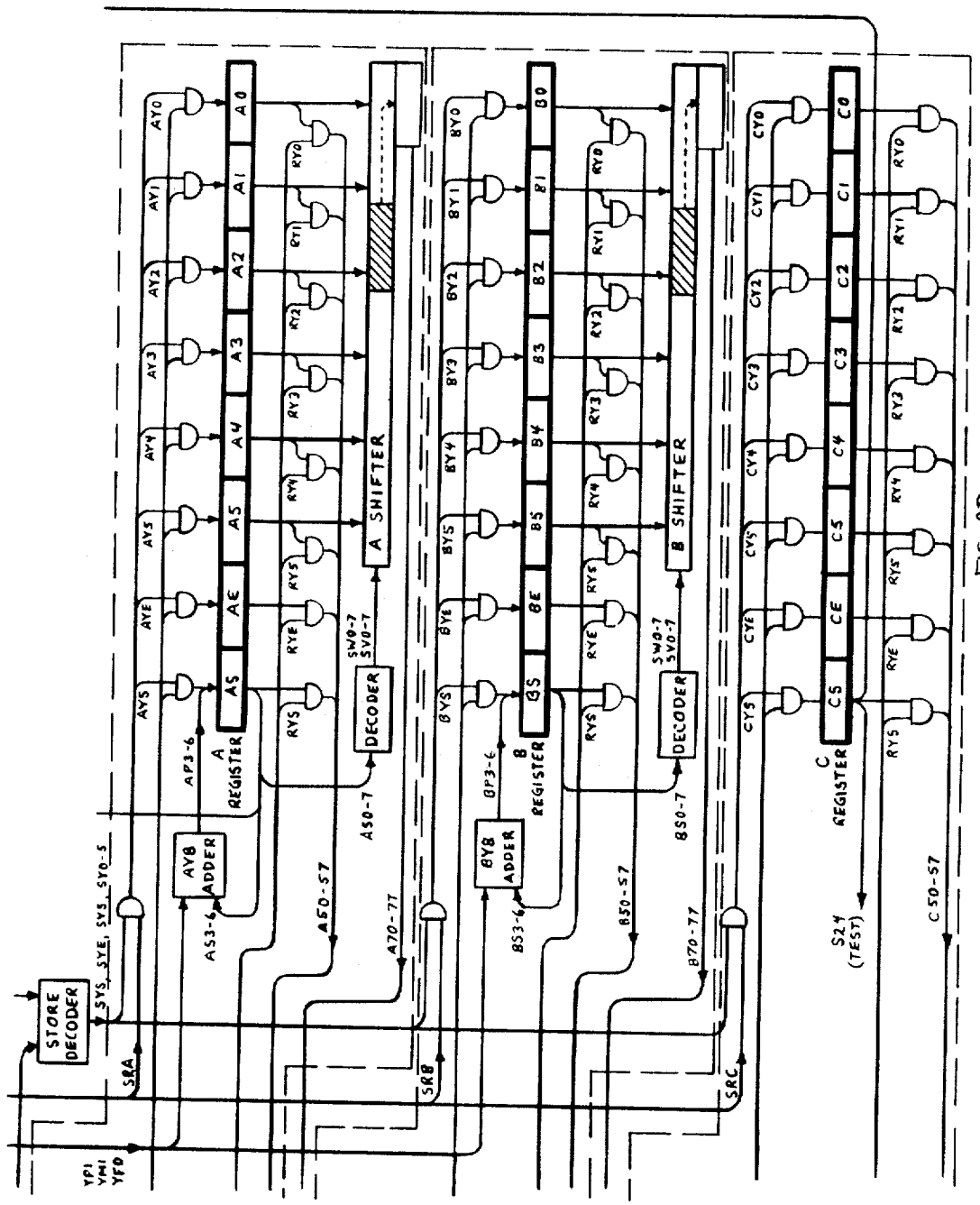

STORE DECODER
SYS, SYE, SV3, SV0-5
SRA
SRB
SRC
AY8 ADDER
AP3-6
AS3-6
A REGISTER
AS
AE
AS
A0
A1
A2
A3
A4
A5
A SHIFTER
DECODER
SW0-7
SV0-7
AS0-7
AS0-S7
A70-77
BY8 ADDER
BP3-6
BS3-6
B REGISTER
BS
BE
BS
B0
B1
B2
B3
B4
B5
B SHIFTER
BS0-7
BS0-S7
B70-77
C REGISTER
CS
CE
CS
C0
C1
C2
C3
C4
C5
S24 (TEST)
CS0-S7
YP1
YM1
YF0

FIG. 9D

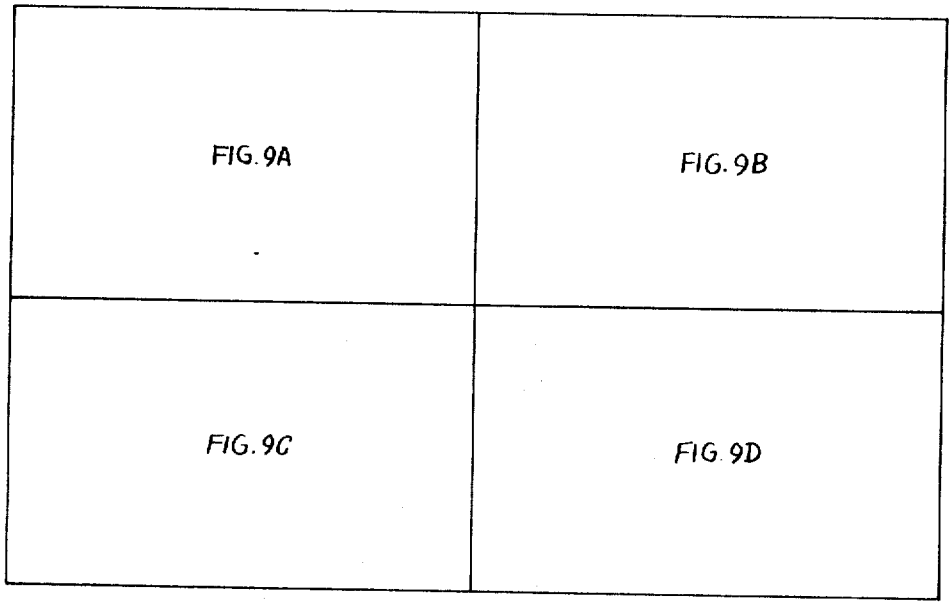



FIG. 10

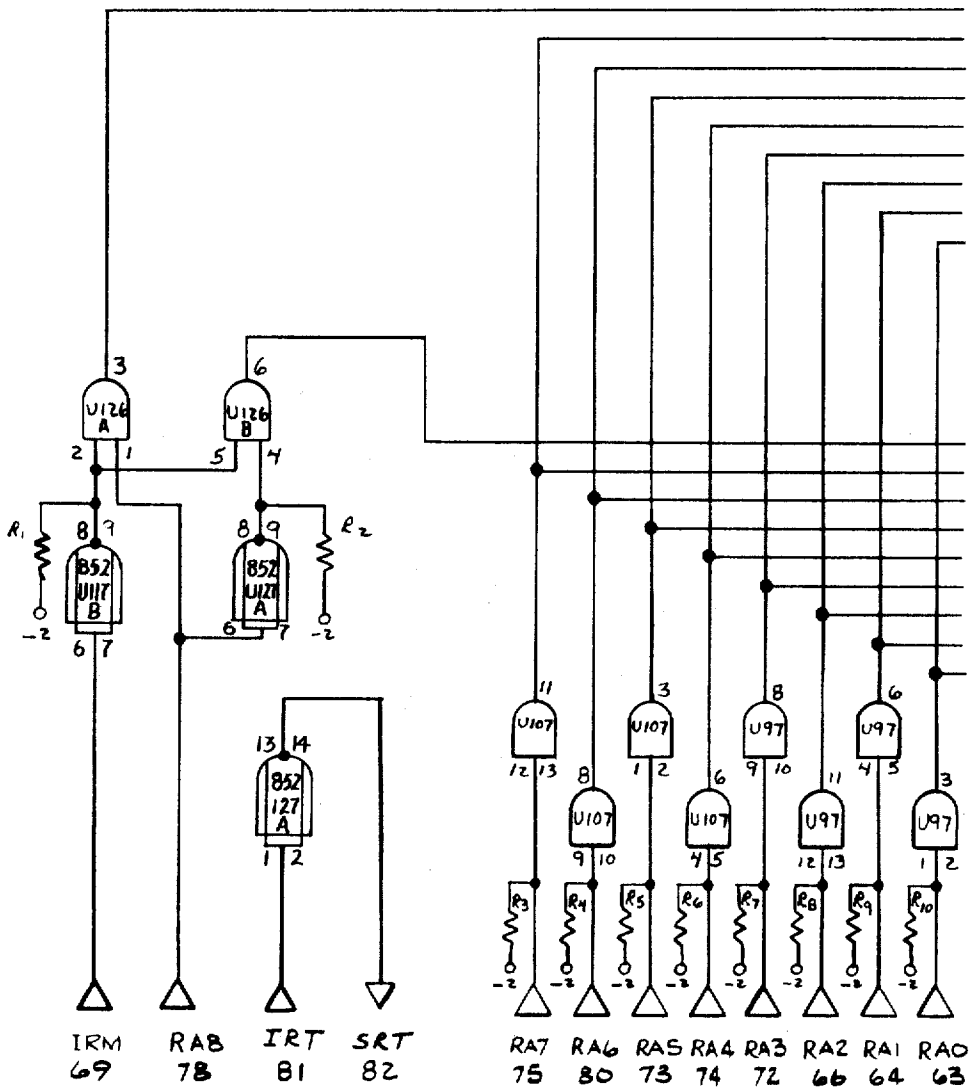

U126 A
U126 B
R1
R2
-2
852 U117 B
852 U127 A
852 127 A
U107
U97
R3
R4
R5
R6
R7
R8
R9
R10
IRM 69
RAB 78
IRT 81
SRT 82
RA7 75
RA6 80
RA5 73
RA4 74
RA3 72
RA2 66
RA1 64
RA0 63

FIG. 11A

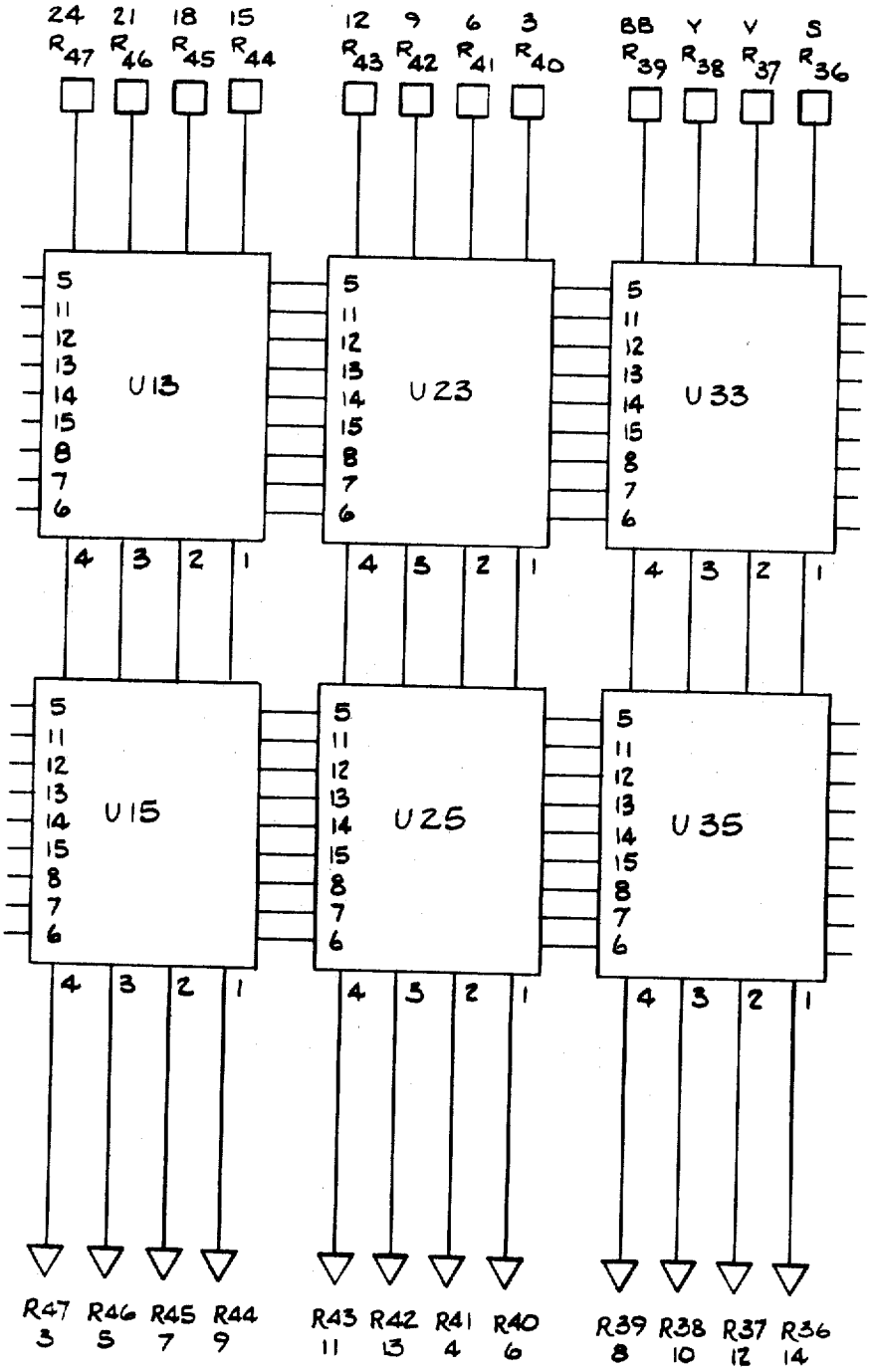

24 21 18 15
R47 R46 R45 R44
12 9 6 3
R43 R42 R41 R40
BB Y V S
R39 R38 R37 R36
U13
U23
U33
U15
U25
U35
5 11 12 13 14 15 8 7 6
4 3 2 1
R47 R46 R45 R44
3 5 7 9
R43 R42 R41 R40
11 13 4 6
R39 R38 R37 R36
8 10 12 14

FIG. 11B

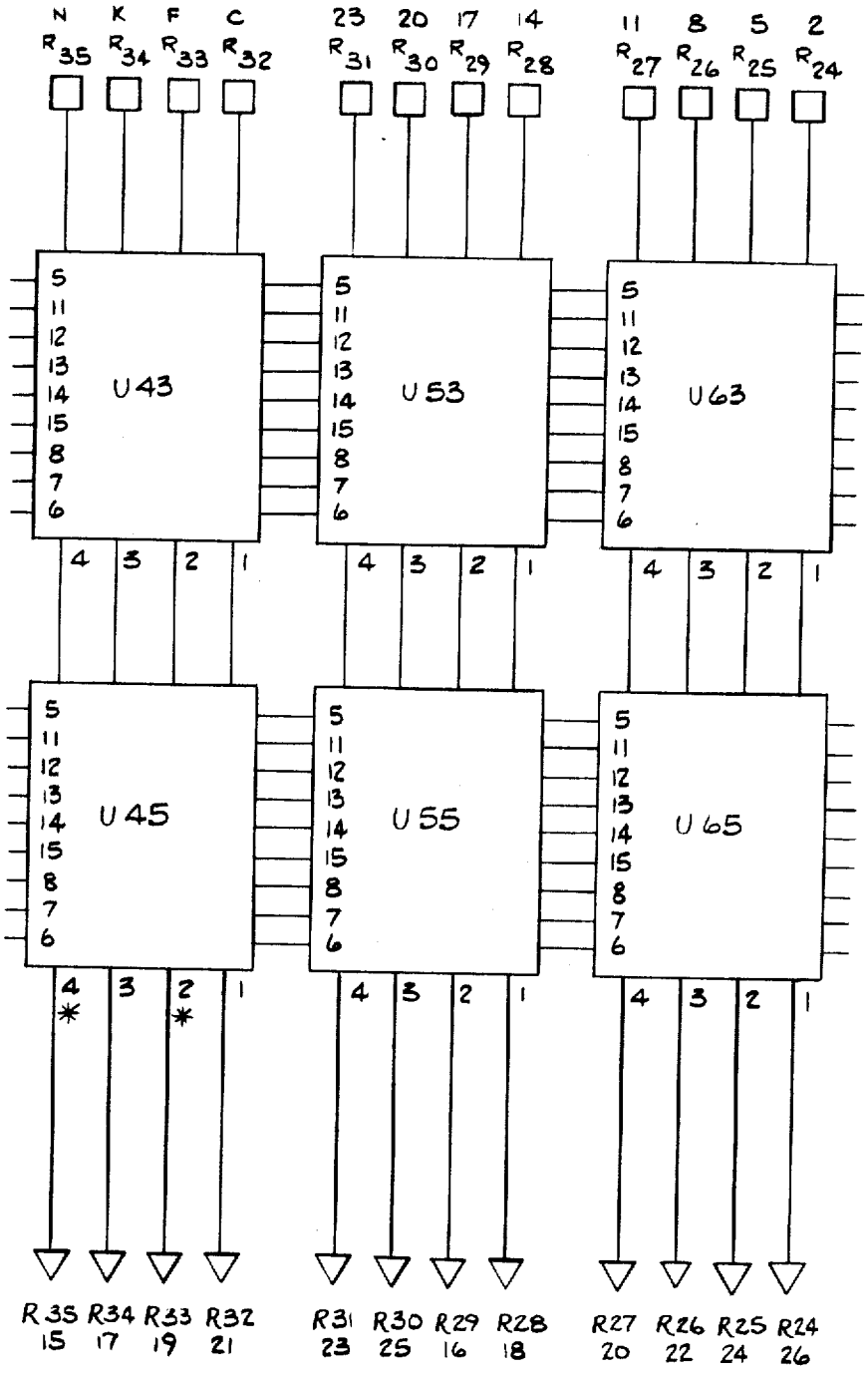

N K F C
R35 R34 R33 R32
23 20 17 14
R31 R30 R29 R28
11 8 5 2
R27 R26 R25 R24
U43
U53
U63
U45
U55
U65
5 11 12 13 14 15 8 7 6
4 3 2 1
R35 R34 R33 R32
15 17 19 21
R31 R30 R29 R28
23 25 16 18
R27 R26 R25 R24
20 22 24 26

FIG. IIC

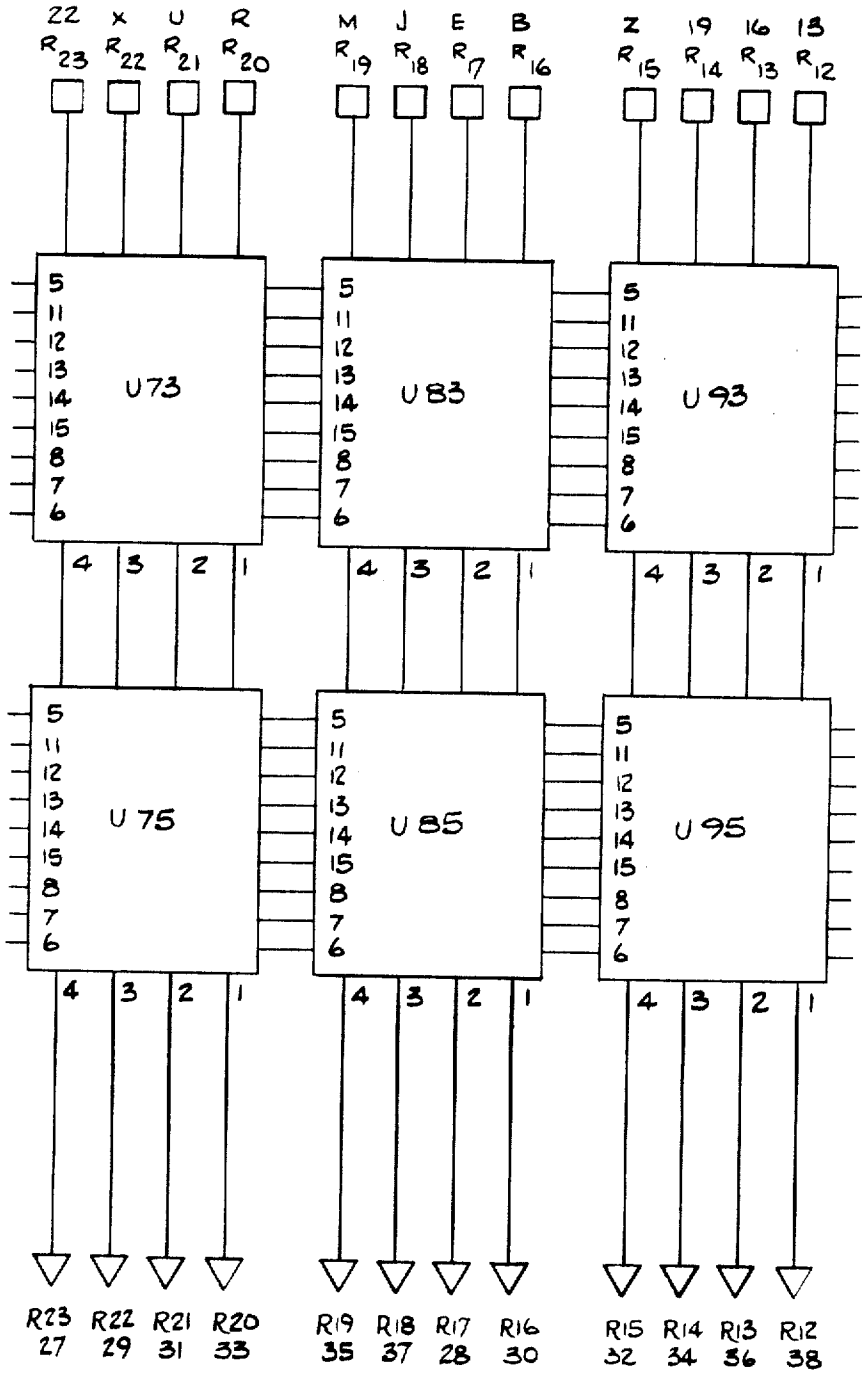

22 X U R
R23 R22 R21 R20
M J E B
R19 R18 R17 R16
Z 19 16 13
R15 R14 R13 R12
U73
U83
U93
U75
U85
U95
5 11 12 13 14 15 8 7 6
4 3 2 1
R23 27
R22 29
R21 31
R20 33
R19 35
R18 37
R17 28
R16 30
R15 32
R14 34
R13 36
R12 38

FIG. 11D

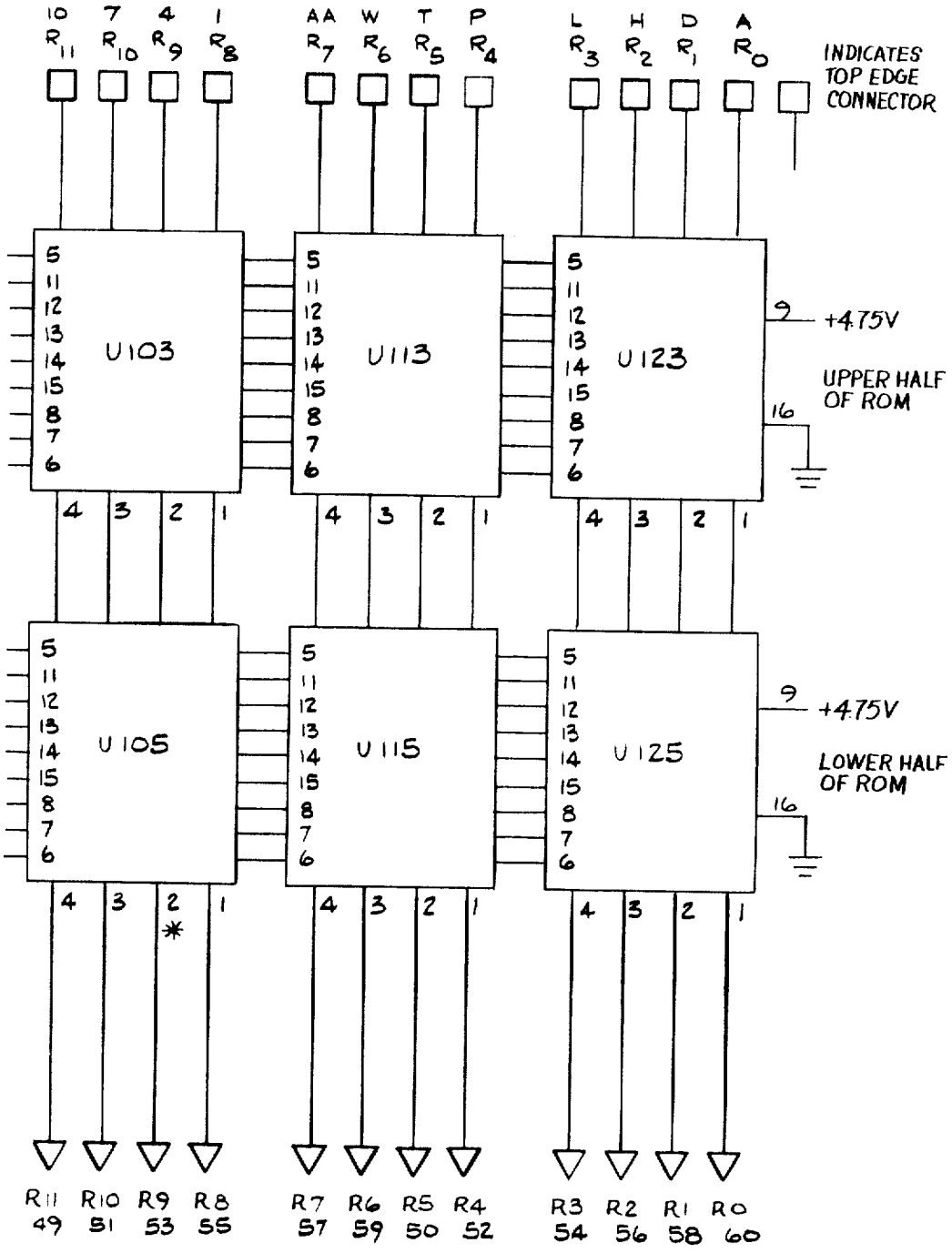

10 R11
7 R10
4 R9
1 R8
AA R7
W R6
T R5
P R4
L R3
H R2
D R1
A R0
INDICATES TOP EDGE CONNECTOR
U103
U113
U123
U105
U115
U125
+4.75V
UPPER HALF OF ROM
LOWER HALF OF ROM
R11 49
R10 51
R9 53
R8 55
R7 57
R6 59
R5 50
R4 52
R3 54
R2 56
R1 58
R0 60

FIG. 11E

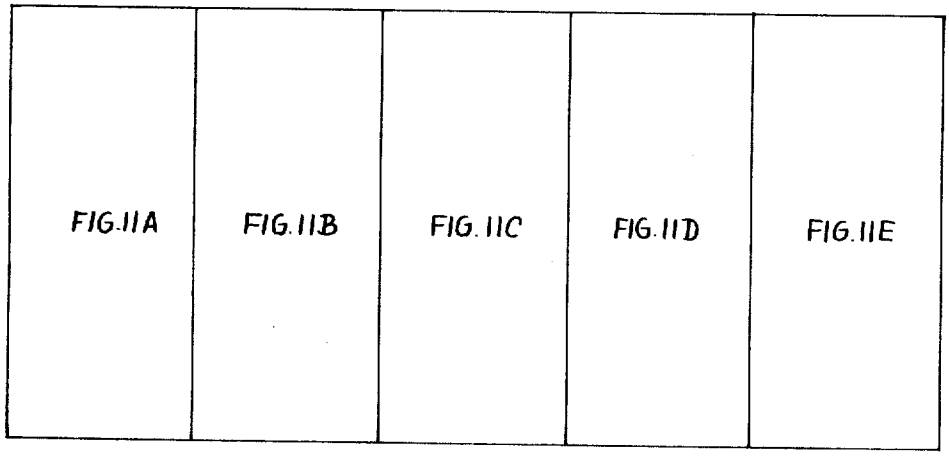



FIG. 12

NOTES:

1. *INDICATES PIN NOT CONNECTED ON THIS MICROCIRCUIT PACKAGE.
2. ALL RESISTORS ARE 560 OHMS.
3. ROM PC WIRING SHOWN IN SIMPLIFIED FORM ABOVE. ACTUAL WIRING OF EACH PACKAGE IS AS SHOWN BELOW.

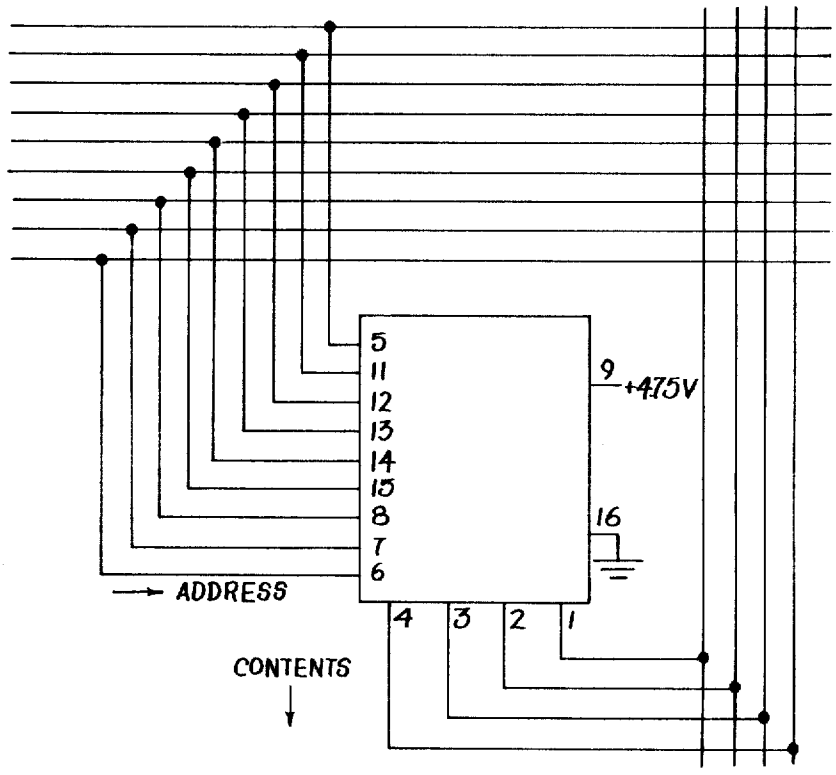



FIG. 13

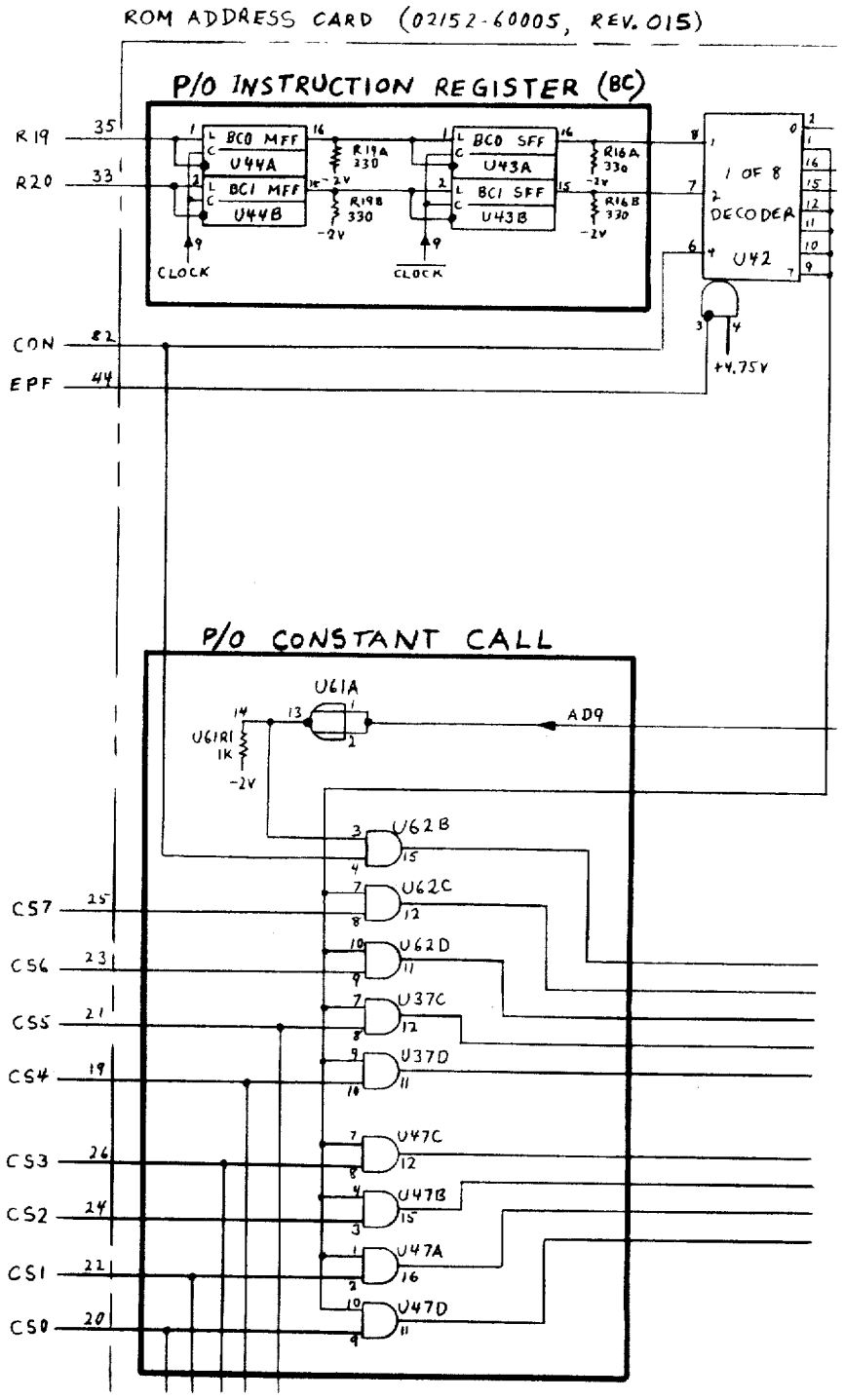

ROM ADDRESS CARD (02152-60005, REV. 015)
P/O INSTRUCTION REGISTER (BC)
R19
R20
35
33
BC0 MFF
U44A
BC1 MFF
U44B
R19A 330
R19B 330
-2V
CLOCK
BC0 SFF
U43A
BC1 SFF
U43B
R16A 330
R16B 330
1 OF 8 DECODER U42
+4.75V
CON
EPF
82
44
P/O CONSTANT CALL
U61A
U61R1 1K
AD9
U62B
U62C
U62D
U37C
U37D
U47C
U47B
U47A
U47D
CS7
CS6
CS5
CS4
CS3
CS2
CS1
CS0
25
23
21
19
26
24
22
20

FIG. 14A

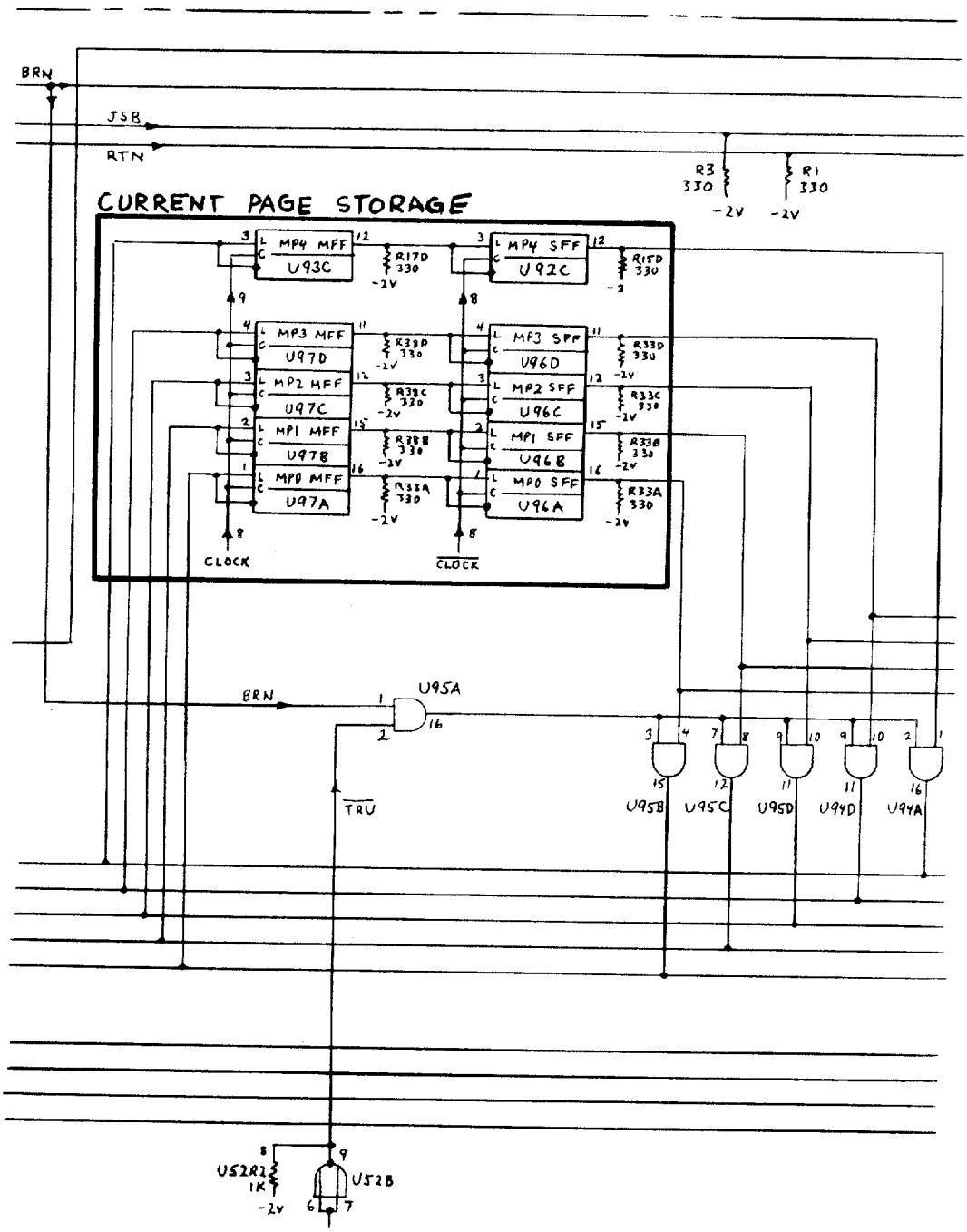

BRN
JSB
RTN
R3 330 -2V
R1 330 -2V
CURRENT PAGE STORAGE
MP4 MFF U93C
MP3 MFF U97D
MP2 MFF U97C
MP1 MFF U97B
MP0 MFF U97A
MP4 SFF U92C
MP3 SFF U96D
MP2 SFF U96C
MP1 SFF U96B
MP0 SFF U96A
R17D 330 -2V
R15D 330 -2
R23D 330 -2V
R33D 330 -1V
R23C 330 -2V
R33C 330 -2V
R23B 330 -2V
R33B 330 -2V
R23A 330 -2V
R33A 330 -2V
CLOCK
CLOCK
BRN
U95A
TRU
U95B
U95C
U95D
U94D
U94A
U52R1 1K -2V
U52B

FIG. 14B

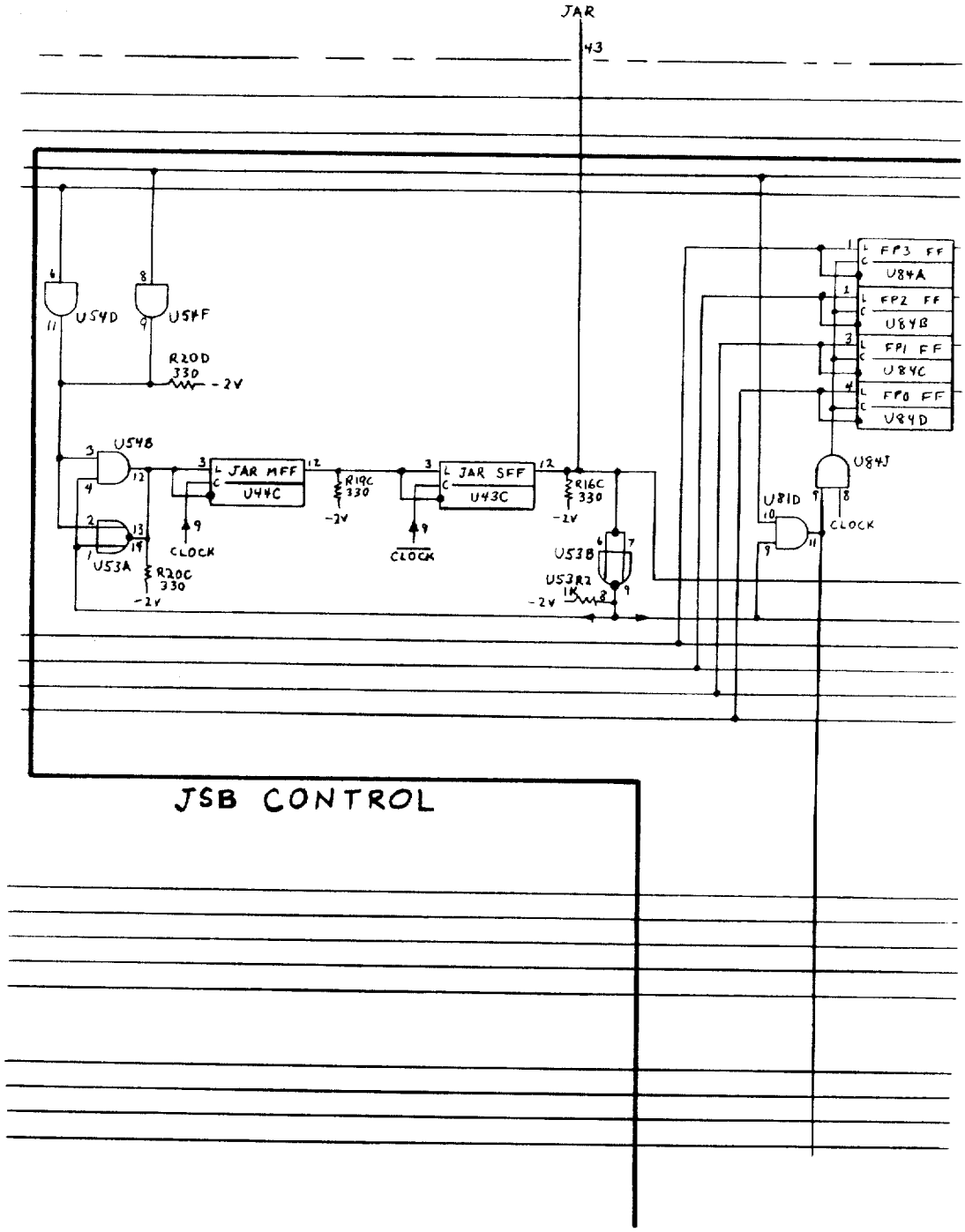

JAR
43
U54D
U54E
R20D
330
-2V
U54B
U53A
R20C
330
-2V
JAR MFF
U44C
R19C
330
-2V
CLOCK
JAR SFF
U43C
R16C
330
-2V
CLOCK
U53B
U53B2
1K
-2V
FF3 FF
U84A
FF2 FF
U84B
FF1 FF
U84C
FF0 FF
U84D
U84J
U81D
CLOCK
JSB CONTROL

FIG. 14C

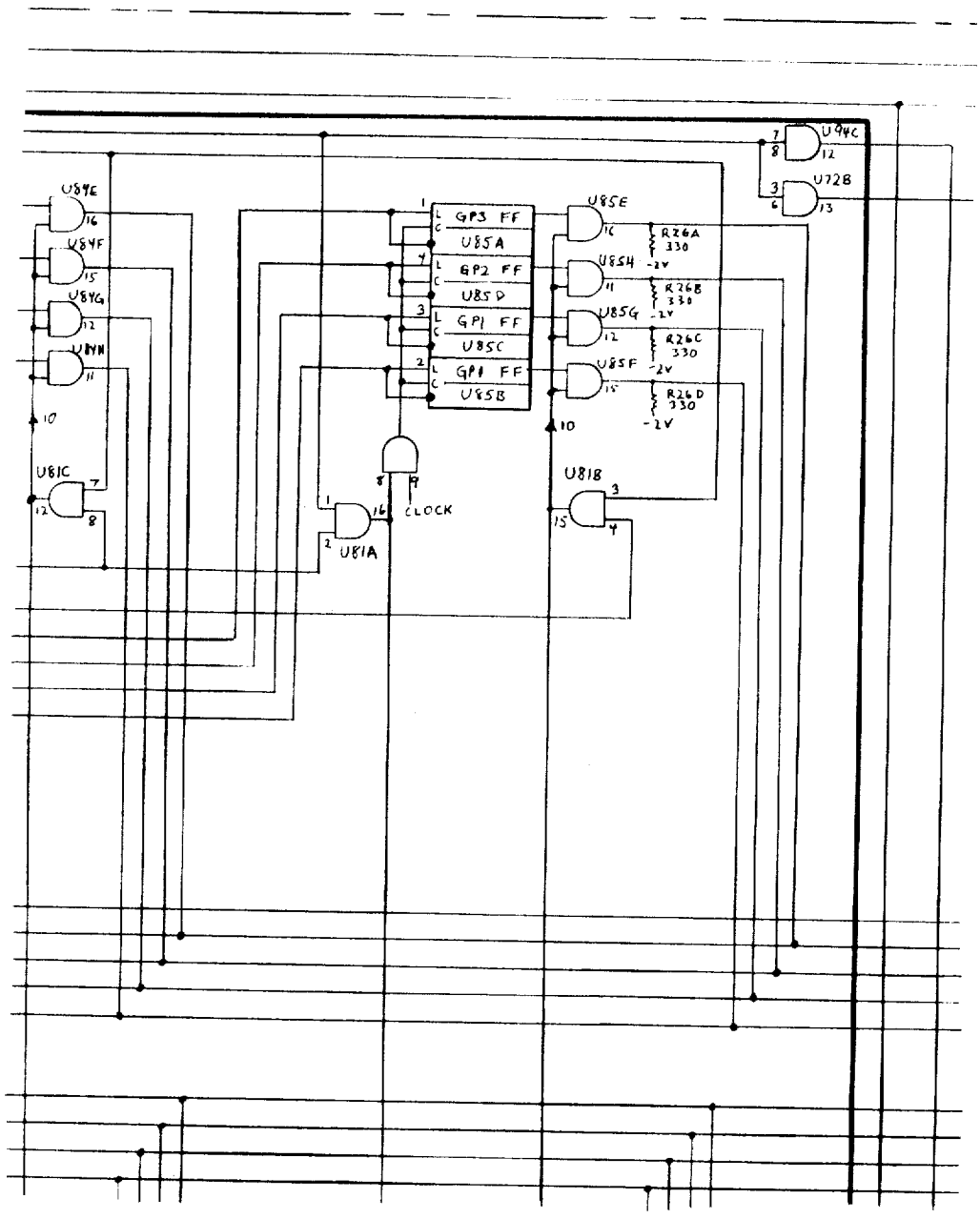

U94C
U72B
U84E
U84F
U84G
U84H
U81C
U81A
U81B
U85E
U85H
U85G
U85F
GP3 FF
U85A
GP2 FF
U85D
GP1 FF
U85C
GP1 FF
U85B
R26A 330 -2V
R26B 330 -2V
R26C 330 -2V
R26D 330 -2V
CLOCK

FIG. 14D

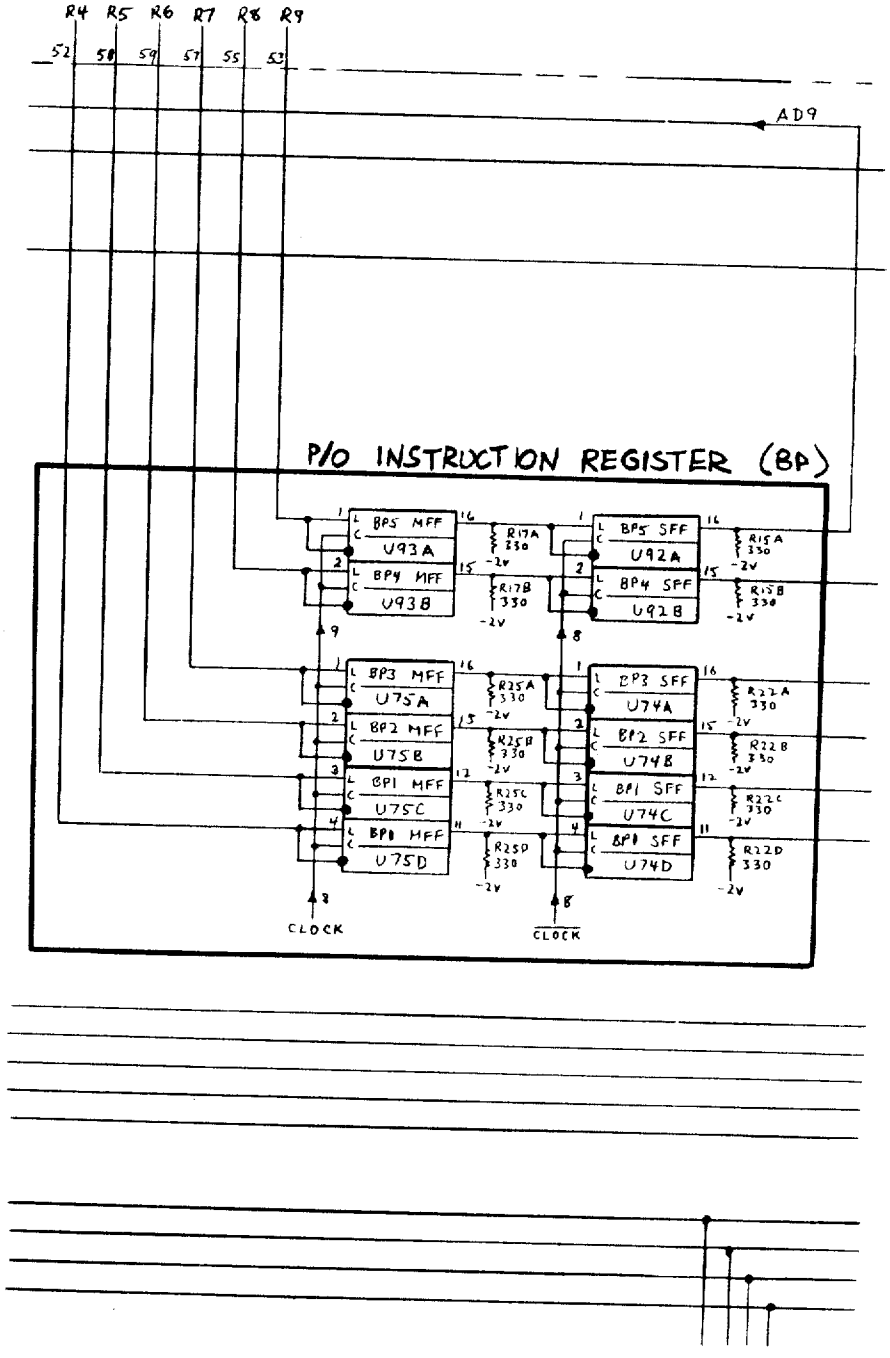

R4 R5 R6 R7 R8 R9
52 50 59 57 55 53
AD9
P/O INSTRUCTION REGISTER (BP)
BP5 MFF U93A
BP4 MFF U93B
BP3 MFF U75A
BP2 MFF U75B
BP1 MFF U75C
BP0 MFF U75D
BP5 SFF U92A
BP4 SFF U92B
BP3 SFF U74A
BP2 SFF U74B
BP1 SFF U74C
BP0 SFF U74D
R17A 330 -2V
R17B 330 -2V
R25A 330 -2V
R25B 330 -2V
R25C 330 -2V
R25D 330 -2V
R15A 330 -2V
R15B 330 -2V
R22A 330 -2V
R22B 330 -2V
R22C 330 -2V
R22D 330 -2V
CLOCK
CLOCK

FIG. 14-E

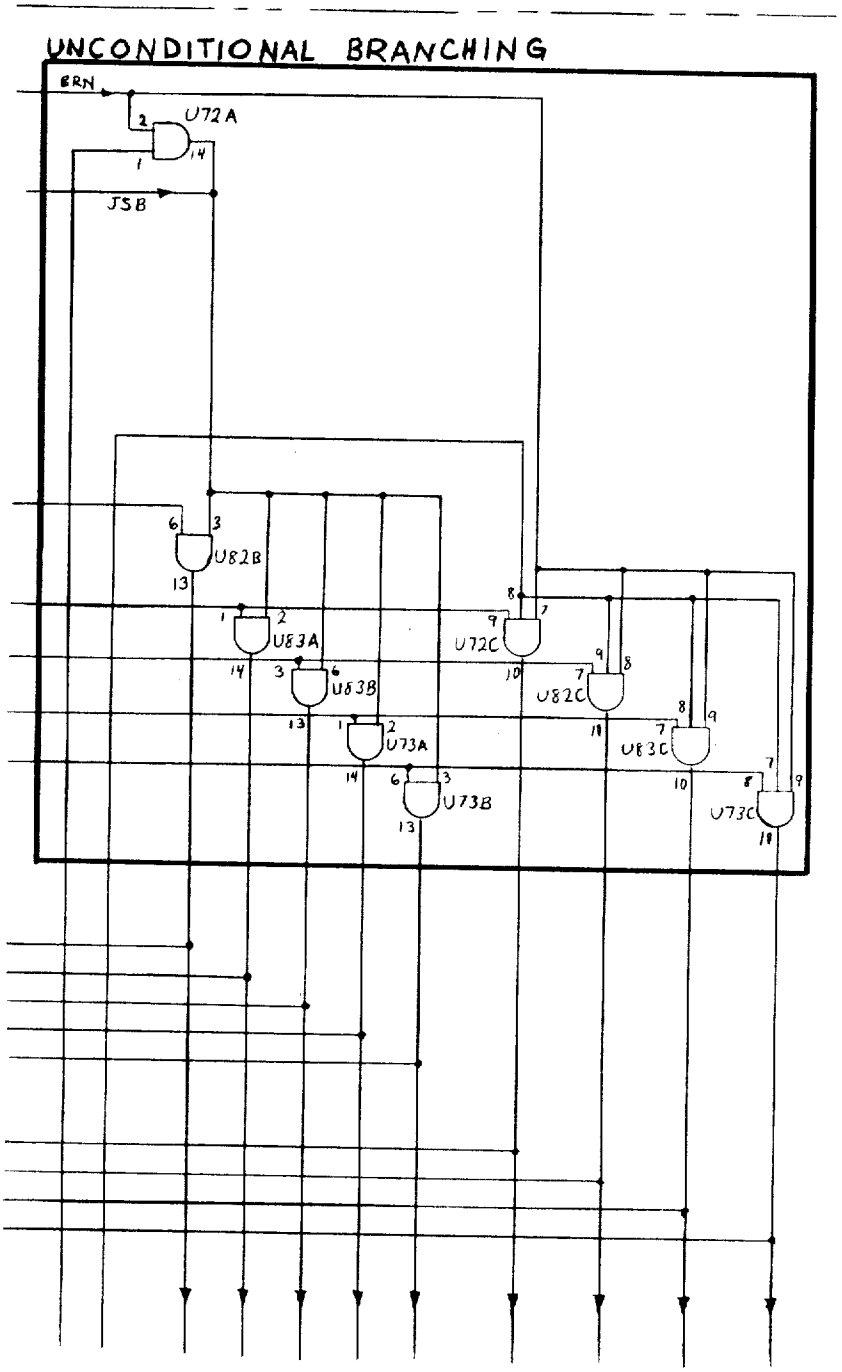

UNCONDITIONAL BRANCHING
BRN
JSB
U72A
U82B
U83A
U83B
U73A
U73B
U72C
U82C
U83C
U73C

FIG. 14F

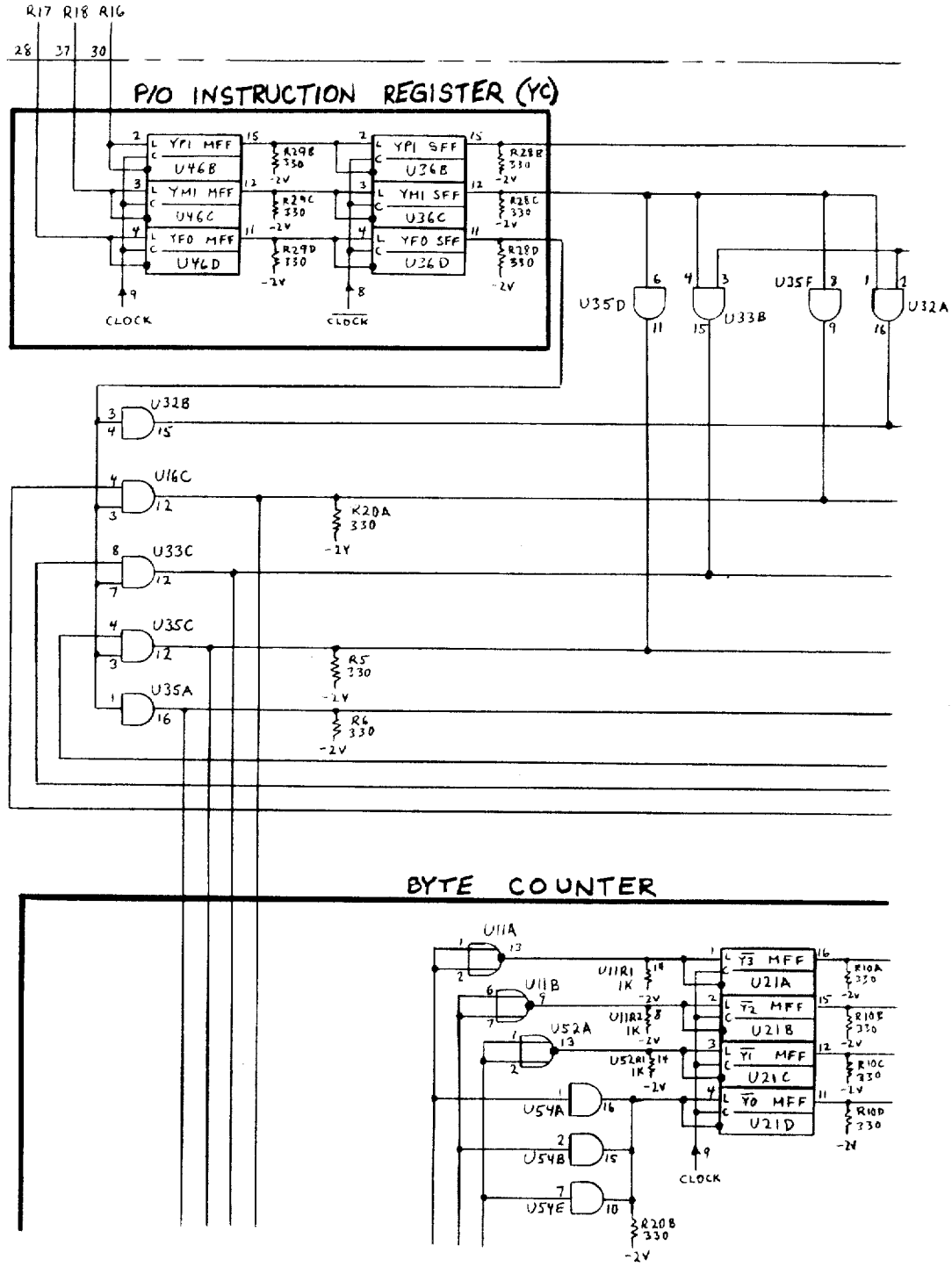

R17 R18 R16
28 37 30
P/O INSTRUCTION REGISTER (YC)
YP1 MFF
U46B
YM1 MFF
U46C
YF0 MFF
U46D
YP1 SFF
U36B
YM1 SFF
U36C
YF0 SFF
U36D
R29B 330 -2V
R29C 330 -2V
R29D 330 -2V
R28B 330 -2V
R28C 330 -2V
R28D 330 -2V
CLOCK
U35D
U33B
U35F
U32A
U32B
U16C
U33C
U35C
U35A
R20A 330 -2V
R5 330 -2V
R6 330 -2V
BYTE COUNTER
U11A
U11B
U52A
U54A
U54B
U54E
U11R1 1K -2V
U11R2 1K -2V
U52R1 1K -2V
R20B 330 -2V
Y3 MFF
U21A
Y2 MFF
U21B
Y1 MFF
U21C
Y0 MFF
U21D
R10A 330 -2V
R10B 330 -2V
R10C 330 -2V
R10D 330 -2V
CLOCK

FIG. 14-G

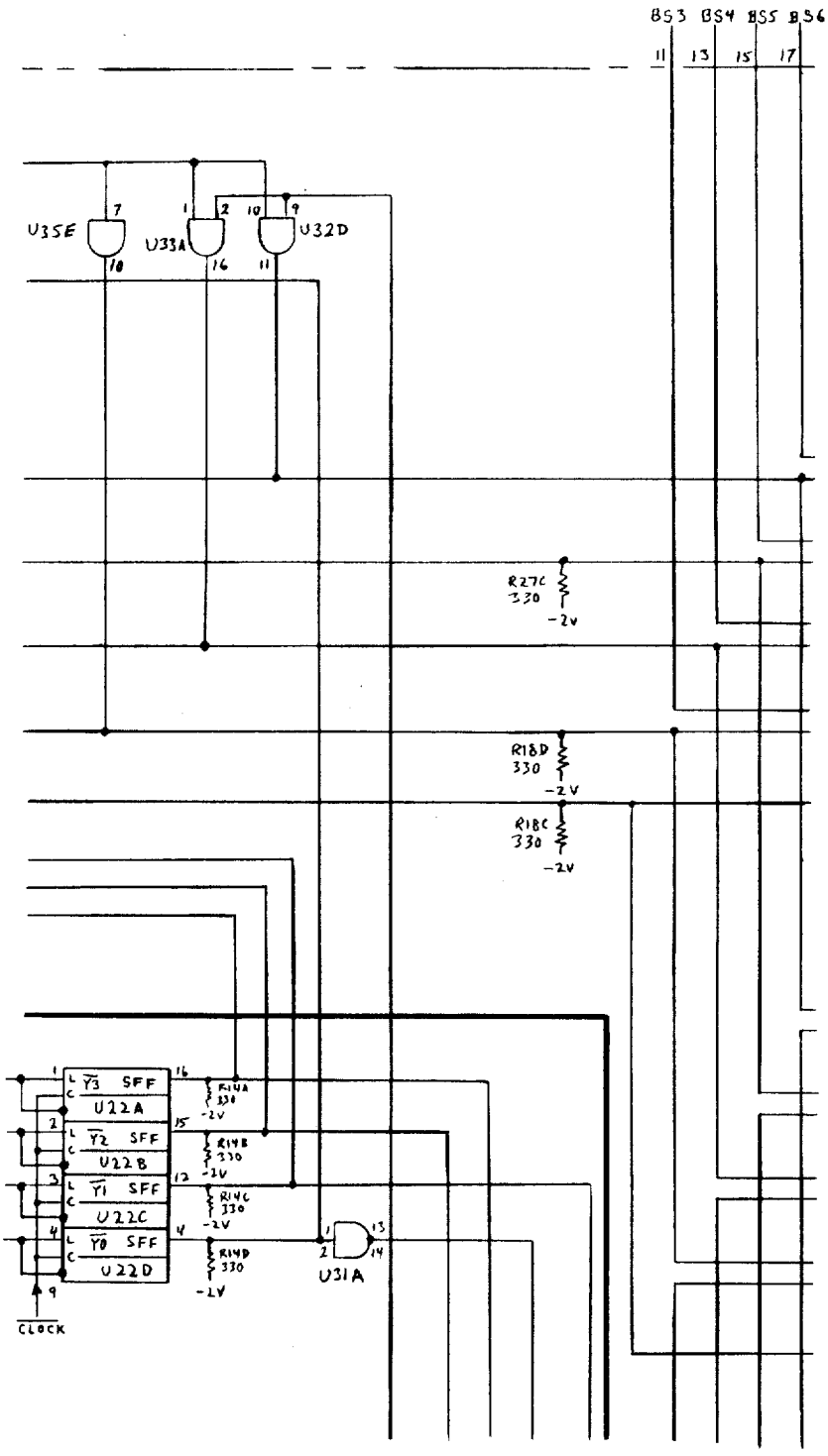

BS3 BS4 BS5 BS6
11 13 15 17
U35E
U33A
U32D
R27C 330 -2V
R18D 330 -2V
R18C 330 -2V
U22A
U22B
U22C
U22D
U31A
CLOCK

FIG. 14H

LOGIC DIAGRAM NOTES

1. UNLESS OTHERWISE INDICATED, pins 6 and 7 of quad latches (input control), pin 10 (read strobe), and unused clock input (pin 8 or 9) are tied to +4.75V.
2. Pin number of an input common to all four elements of the quad latches is shown adjacent to an arrowhead leading to the logic symbols.
3. MFF = Master flip-flop; SFF = Slave flip-flop.
4. P/O = "Part of..."

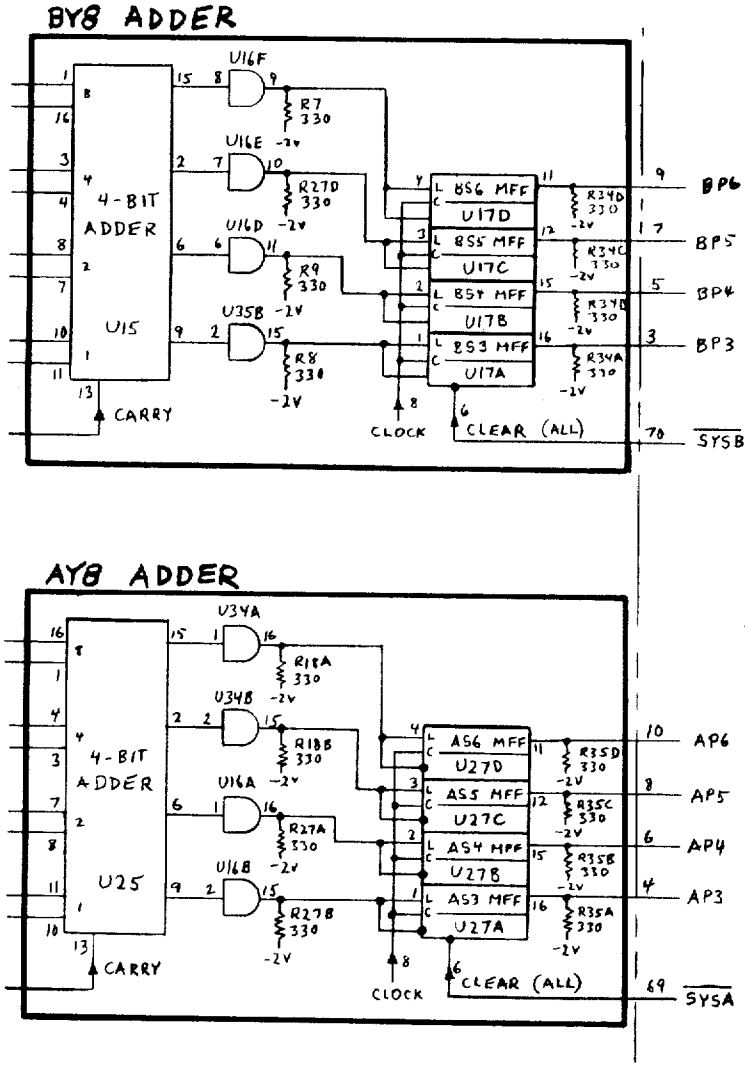



FIG. 14I

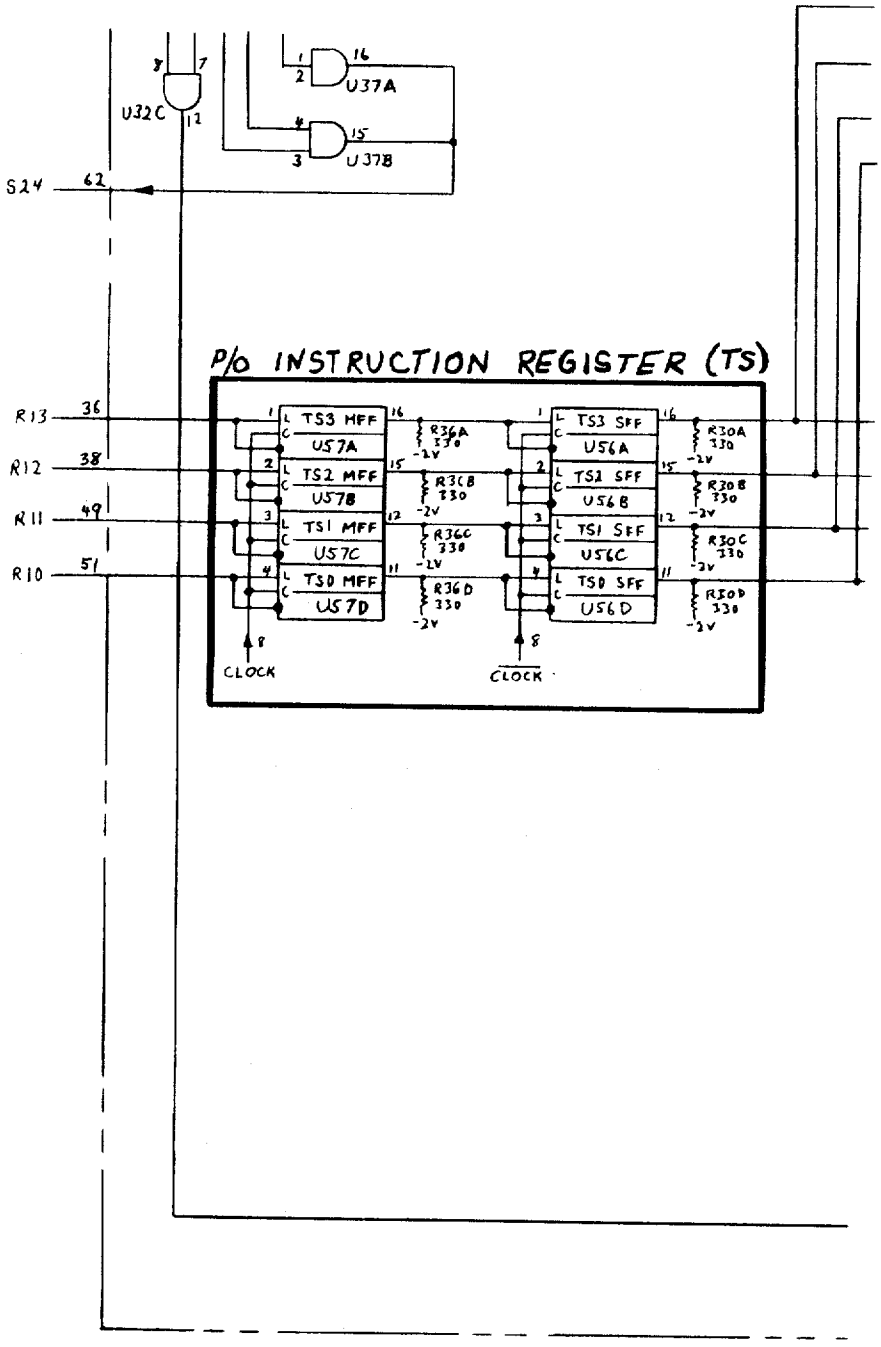

U32C
U37A
U37B
S24
62
P/O INSTRUCTION REGISTER (TS)
R13
36
R12
38
R11
49
R10
51
TS3 MFF
U57A
TS2 MFF
U57B
TS1 MFF
U57C
TS0 MFF
U57D
TS3 SFF
U56A
TS2 SFF
U56B
TS1 SFF
U56C
TS0 SFF
U56D
R36A 330
R36B 330
R36C 330
R36D 330
R30A 330
R30B 330
R30C 330
R30D 330
-2V
CLOCK
CLOCK

FIG. 14d

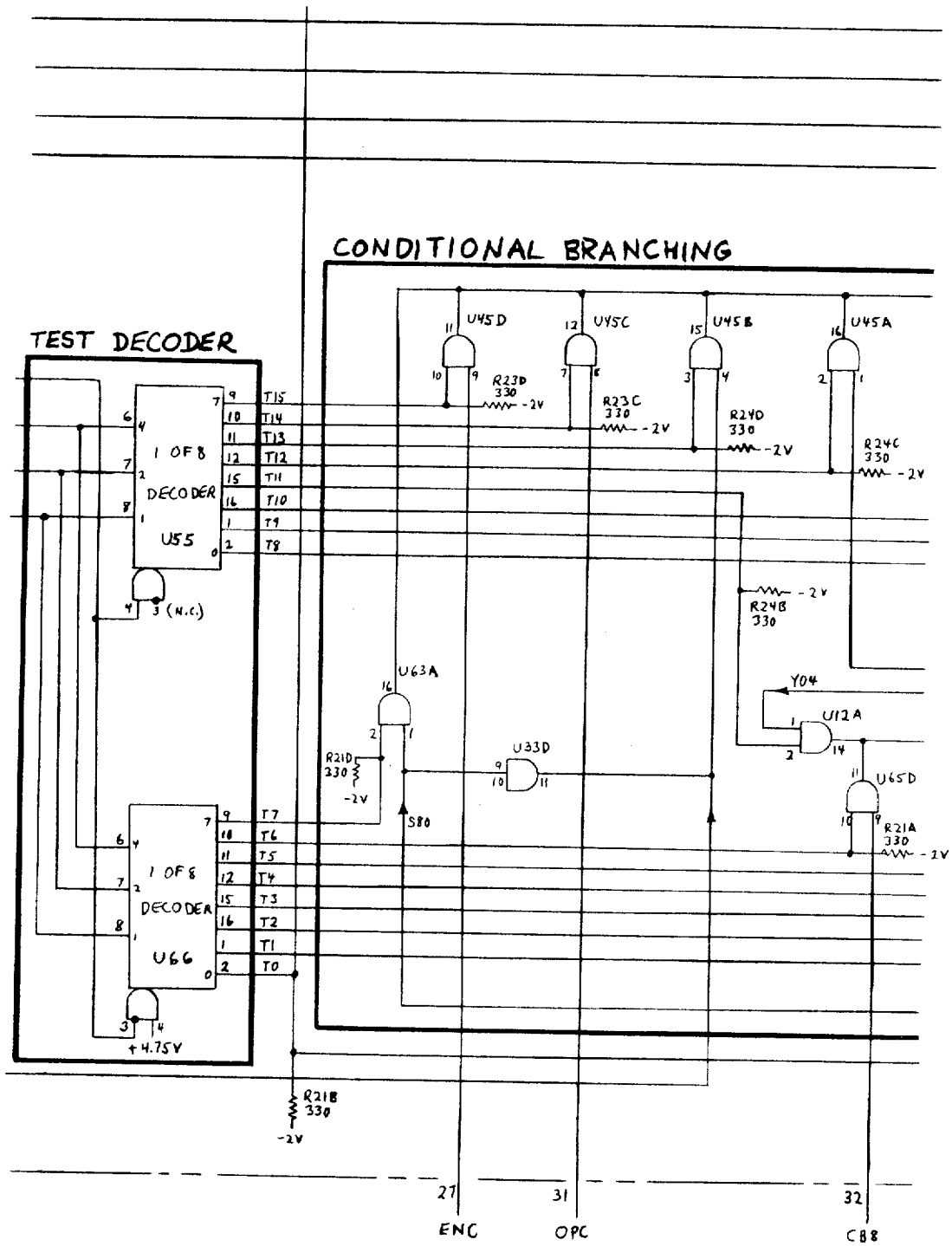

CONDITIONAL BRANCHING
TEST DECODER
1 OF 8 DECODER U55
1 OF 8 DECODER U66
U45D
U45C
U45B
U45A
R23D 330 -2V
R23C 330 -2V
R24D 330 -2V
R24C 330 -2V
R24B 330 -2V
U63A
U33D
U12A
U65D
Y04
R21D 330 -2V
R21A 330 -2V
R21B 330 -2V
S80
+4.75V
T15
T14
T13
T12
T11
T10
T9
T8
T7
T6
T5
T4
T3
T2
T1
T0
27 ENC
31 OPC
32 CBB

FIG. 14K

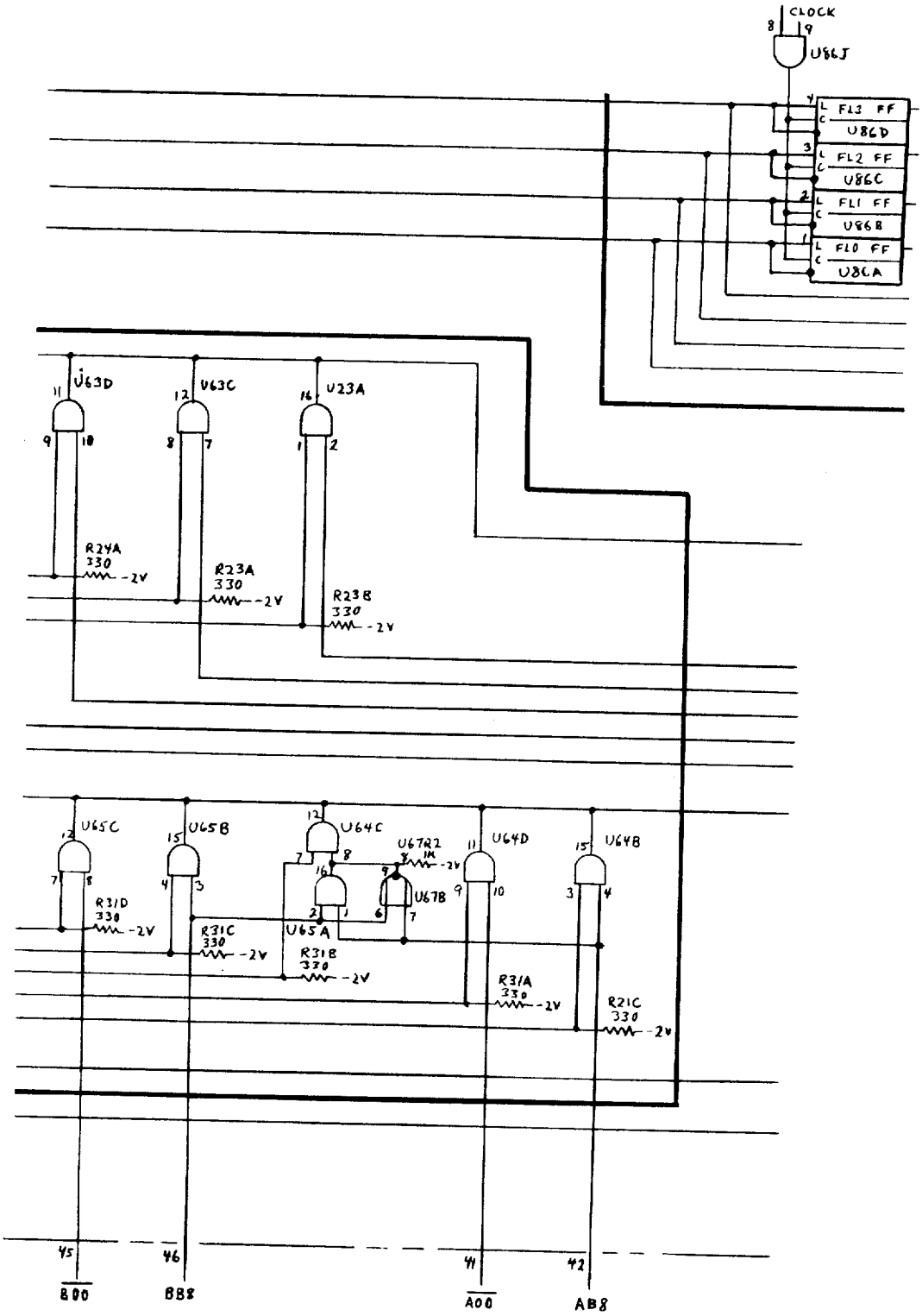

CLOCK
U86J
FL3 FF
U86D
FL2 FF
U86C
FL1 FF
U86B
FL0 FF
U86A
U63D
U63C
U23A
R24A 330 -2V
R23A 330 -2V
R23B 330 -2V
U65C
U65B
U64C
U67R2 1K -2V
U67B
U65A
U64D
U64B
R31D 330 -2V
R31C 330 -2V
R31B 330 -2V
R31A 330 -2V
R21C 330 -2V
45
46
41
42
B00
BB8
A00
AB8

FIG. 14L

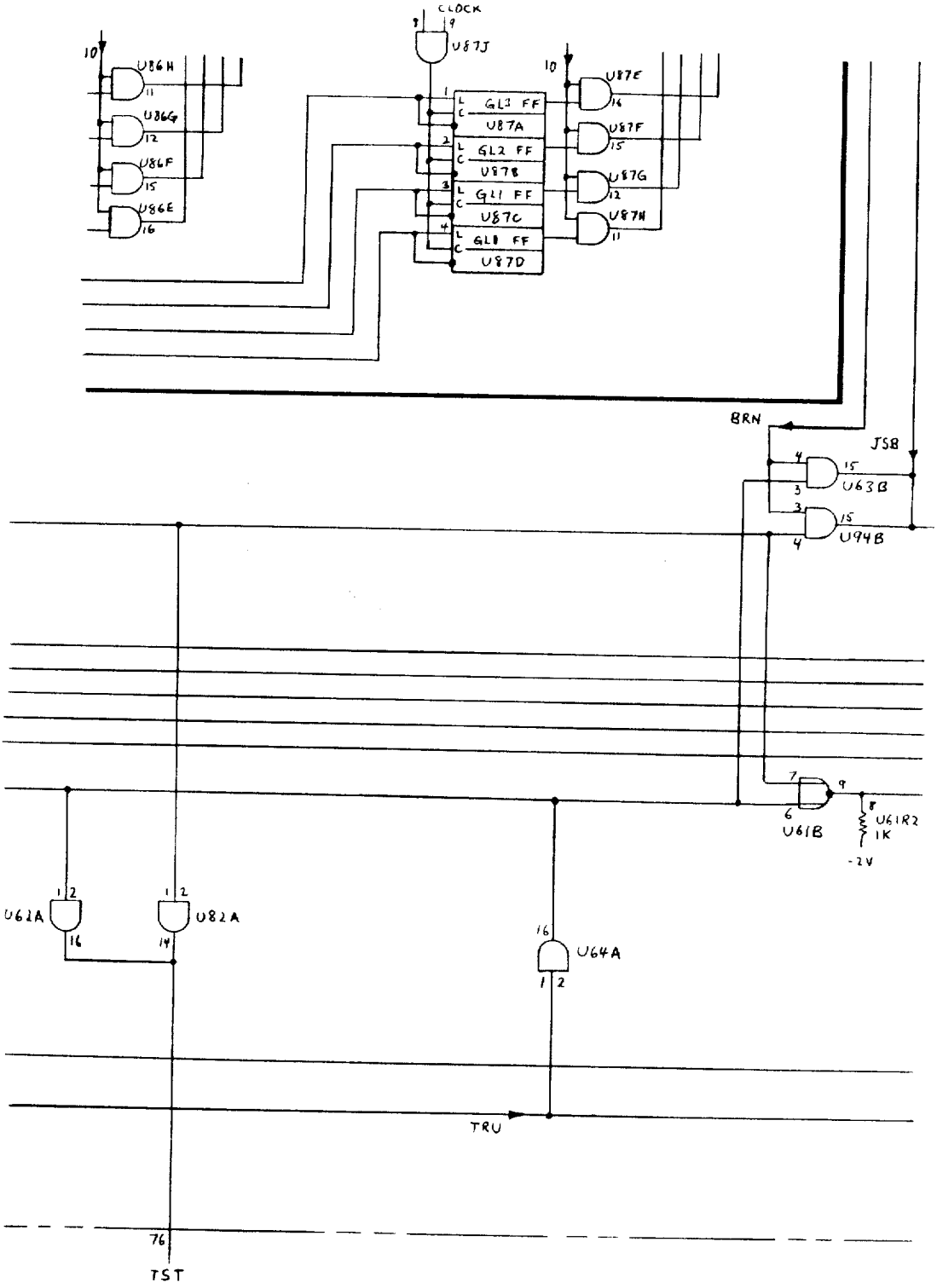

CLOCK
U87J
U86H
U86G
U86F
U86E
GL3 FF
U87A
GL2 FF
U87B
GL1 FF
U87C
GL0 FF
U87D
U87E
U87F
U87G
U87H
BRN
JS8
U63B
U94B
U61B
U61R2
1K
-2V
U62A
U82A
U64A
TRU
76
TST

FIG. 14M

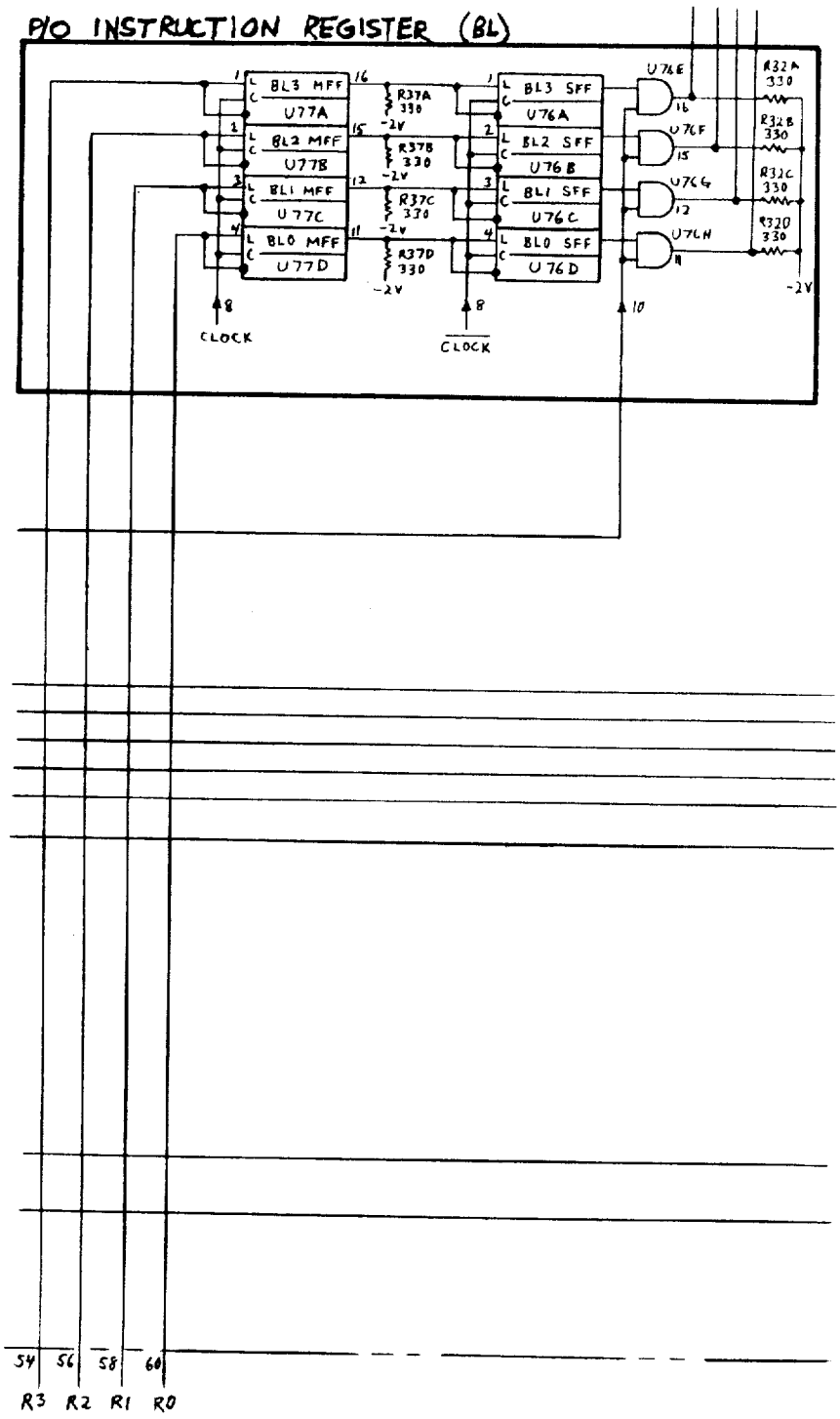

P/O INSTRUCTION REGISTER (BL)
BL3 MFF
U77A
BL2 MFF
U77B
BL1 MFF
U77C
BL0 MFF
U77D
R37A 330
R37B 330
R37C 330
R37D 330
-2V
CLOCK
BL3 SFF
U76A
BL2 SFF
U76B
BL1 SFF
U76C
BL0 SFF
U76D
U76E
U76F
U76G
U76H
R32A 330
R32B 330
R32C 330
R32D 330
54
56
58
60
R3
R2
R1
R0

FIG. 14N

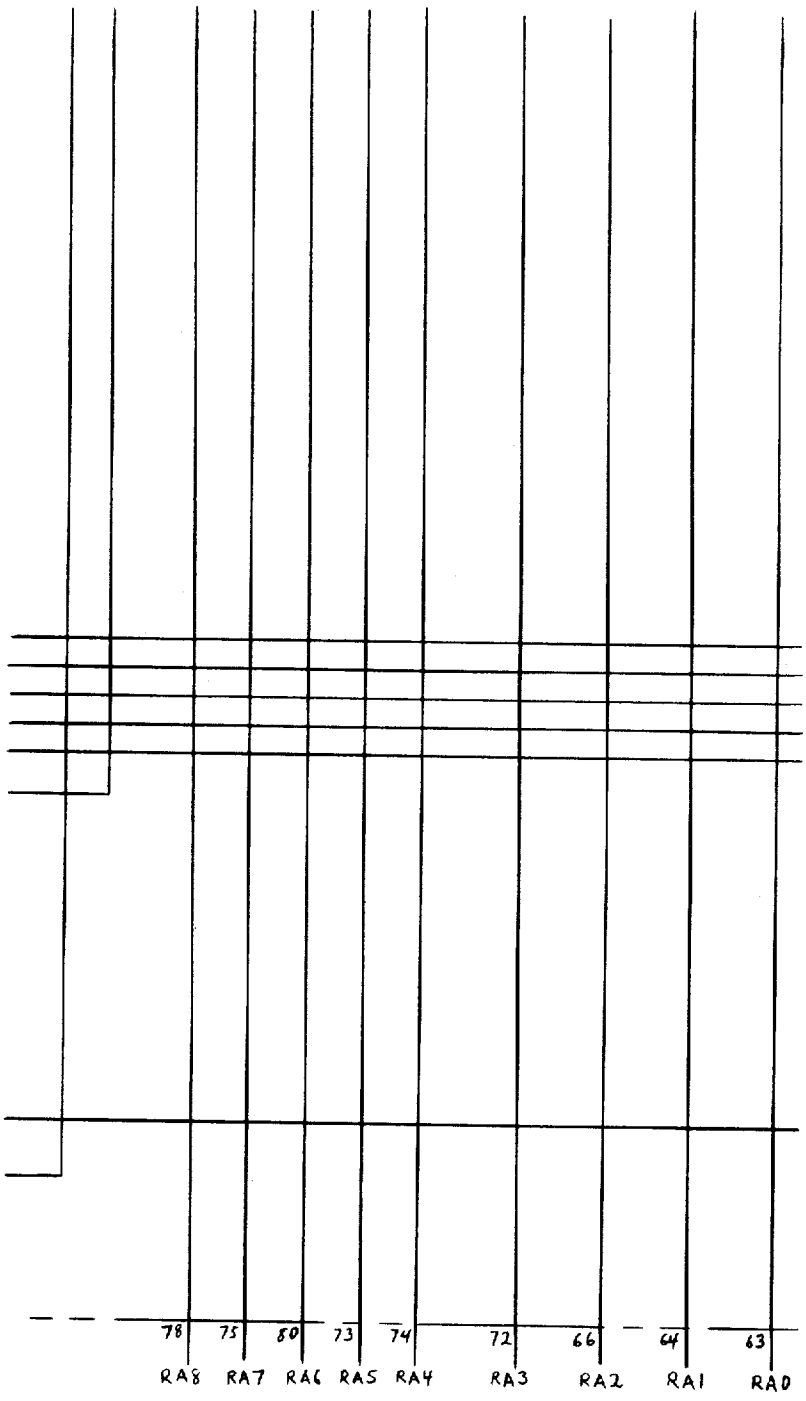



FIG. 140

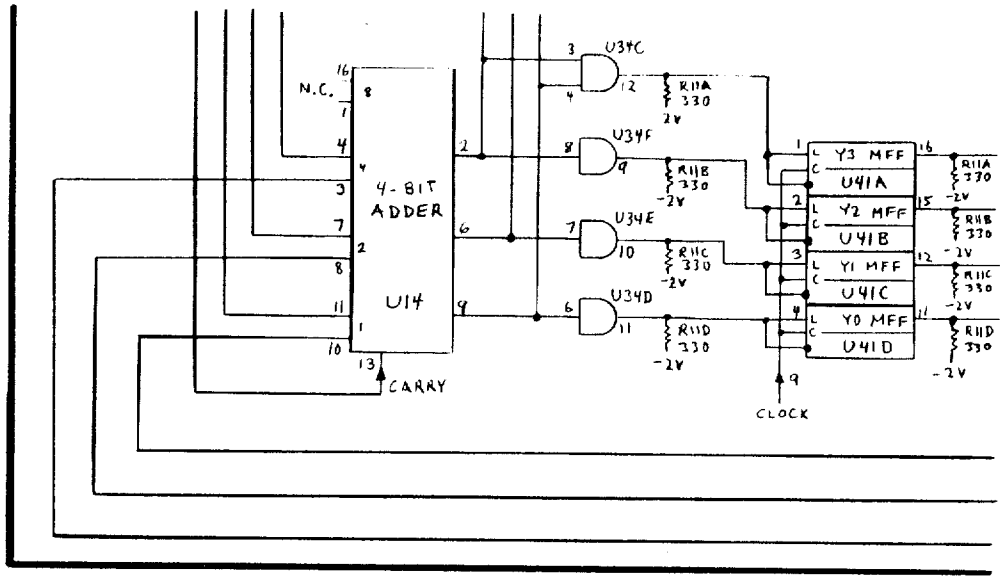

N.C.
4-BIT ADDER
U14
CARRY
U34C
U34F
U34E
U34D
R11A 330 2V
R11B 330 -2V
R11C 330 -2V
R11D 330 -2V
Y3 MFF U41A
Y2 MFF U41B
Y1 MFF U41C
Y0 MFF U41D
CLOCK

FIG. 14-P

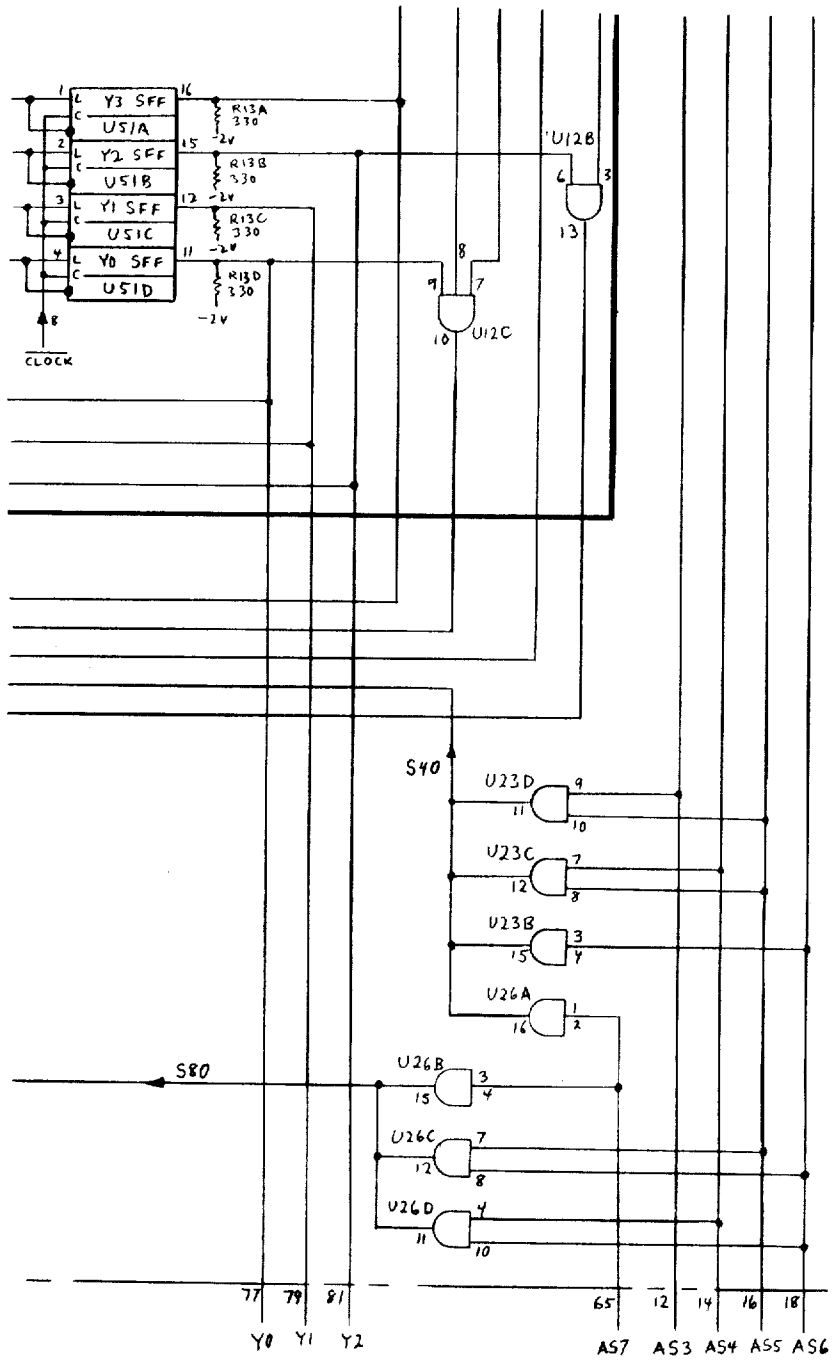

Y3 SFF
U51A
Y2 SFF
U51B
Y1 SFF
U51C
Y0 SFF
U51D
CLOCK
R13A 330
R13B 330
R13C 330
R13D 330
-2V
U12B
U12C
S40
U23D
U23C
U23B
U26A
S80
U26B
U26C
U26D
77
79
81
65
12
14
16
18
Y0
Y1
Y2
AS7
AS3
AS4
AS5
AS6

FIG. 14Q

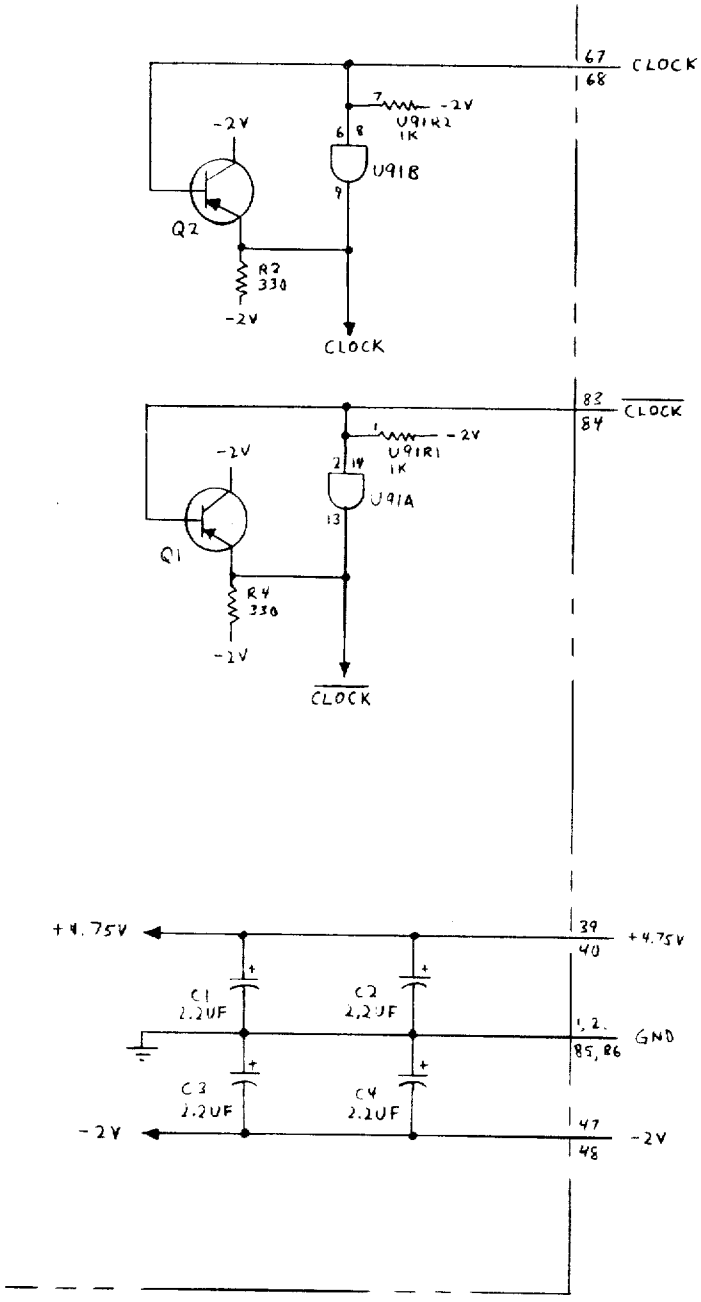

67
68
CLOCK
7
-2V
U91R2
1K
6
8
U91B
9
-2V
Q2
R2
330
-2V
CLOCK
83
84
CLOCK
1
-2V
U91R1
1K
2
14
U91A
13
-2V
Q1
R4
330
-2V
CLOCK
+4.75V
39
40
+4.75V
C1
2.2UF
C2
2.2UF
1, 2,
85, 86
GND
C3
2.2UF
C4
2.2UF
-2V
47
48
-2V

FIG. 14R

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| FIG 14A | FIG.14B | FIG.14C | FIG.14D | FIG.14E | FIG.14F | FIG.14G | FIG.14H | FIG.14I |
| FIG.14J | FIG.14K | FIG.14L | FIG.14M | FIG.14N | FIG.14O | FIG.14P | FIG.14Q | FIG.14R |

FIG. 15

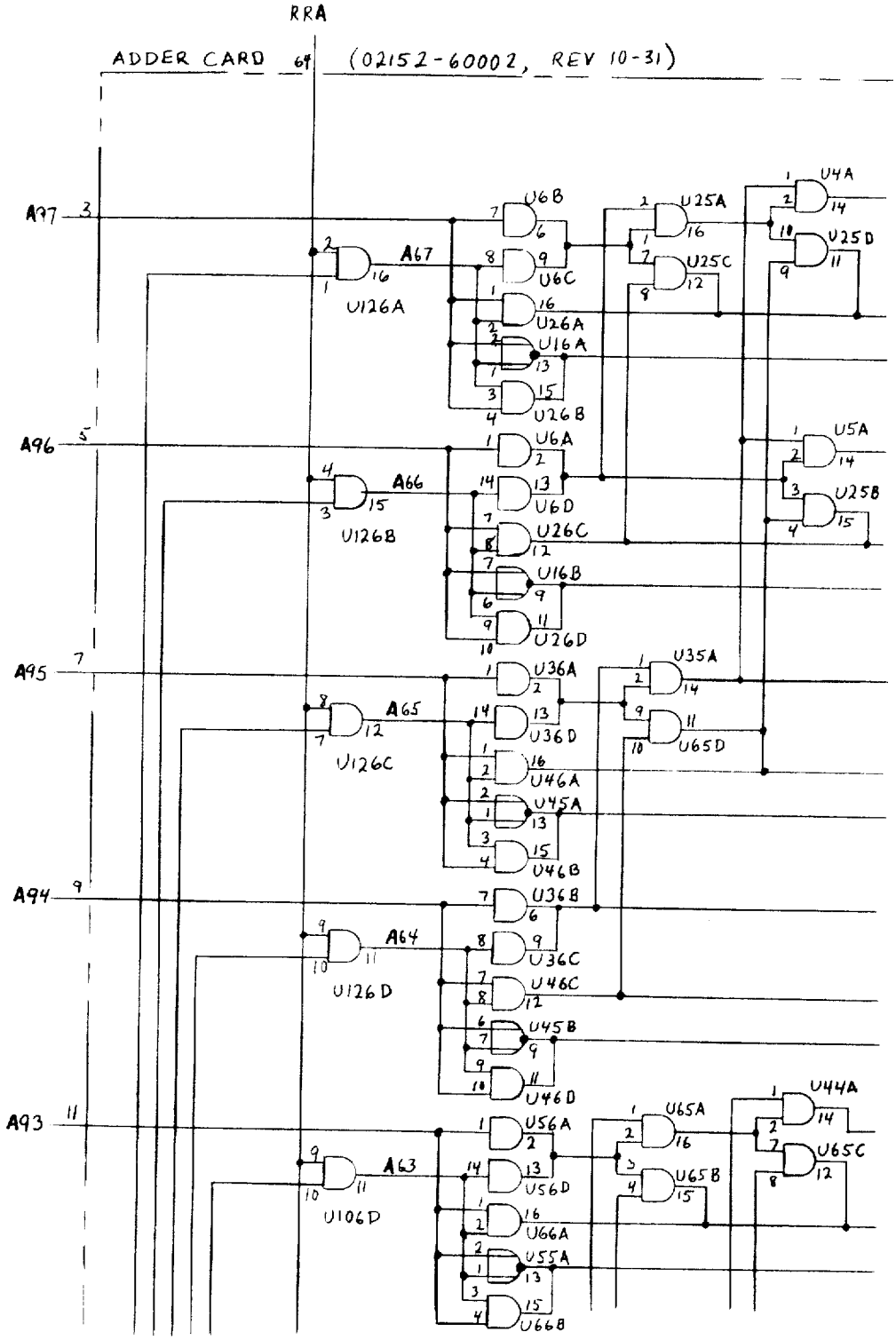

RRA
ADDER CARD 69 (02152-60002, REV 10-31)
A97
A96
A95
A94
A93
A67
A66
A65
A64
A63
U126A
U126B
U126C
U126D
U106D
U6B
U6C
U26A
U16A
U26B
U6A
U6D
U26C
U16B
U26D
U36A
U36D
U46A
U45A
U46B
U36B
U36C
U46C
U45B
U46D
U56A
U56D
U66A
U55A
U66B
U25A
U25C
U25D
U35A
U65D
U65A
U65B
U4A
U5A
U25B
U44A
U65C

FIG. 16A

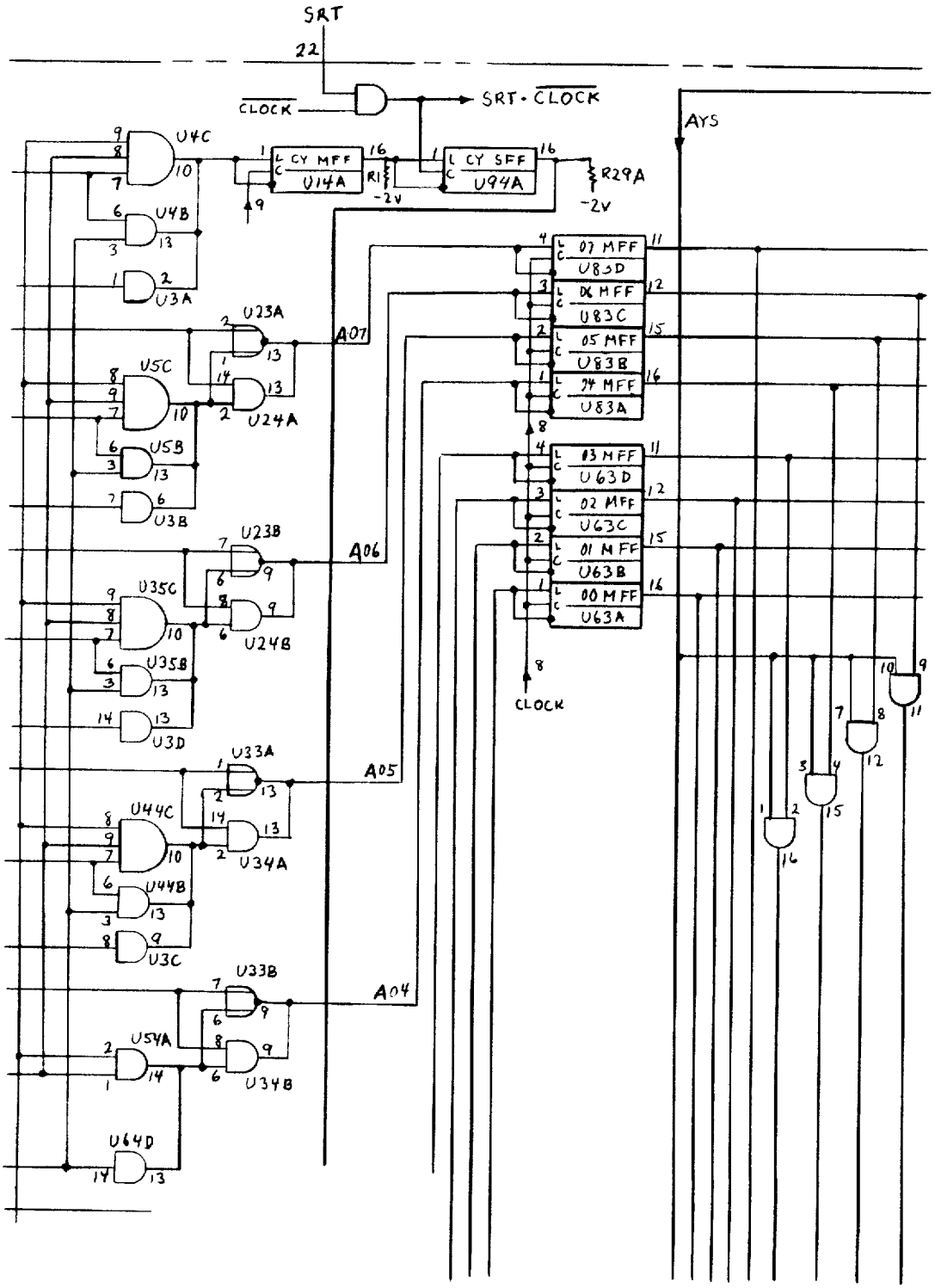

SRT
22
CLOCK
SRT · CLOCK
AYS
U4C
U4B
U3A
U23A
U5C
U24A
U5B
U3B
U23B
U35C
U24B
U35B
U3D
U33A
U44C
U34A
U44B
U3C
U33B
U54A
U34B
U64D
CY MFF U14A
CY SFF U94A
R1
-2V
R29A
-2V
07 MFF U83D
06 MFF U83C
05 MFF U83B
04 MFF U83A
03 MFF U63D
02 MFF U63C
01 MFF U63B
00 MFF U63A
A07
A06
A05
A04
CLOCK

FIG. 16B

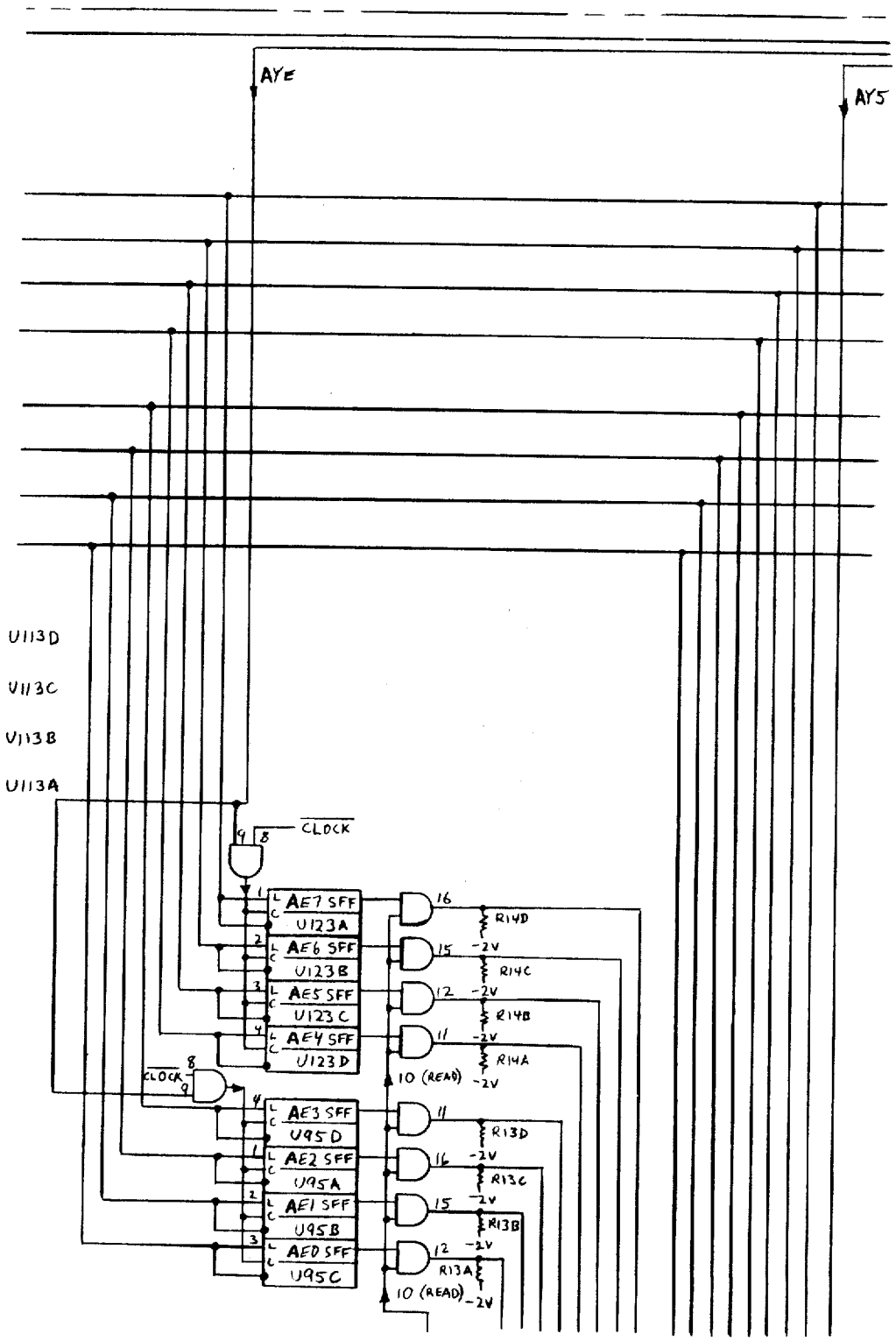

AYE
AY5
U113D
U113C
U113B
U113A
CLOCK
AE7 SFF
U123A
AE6 SFF
U123B
AE5 SFF
U123C
AE4 SFF
U123D
AE3 SFF
U95D
AE2 SFF
U95A
AE1 SFF
U95B
AE0 SFF
U95C
R14D
R14C
R14B
R14A
R13D
R13C
R13B
R13A
-2V
10 (READ)

FIG. 16C

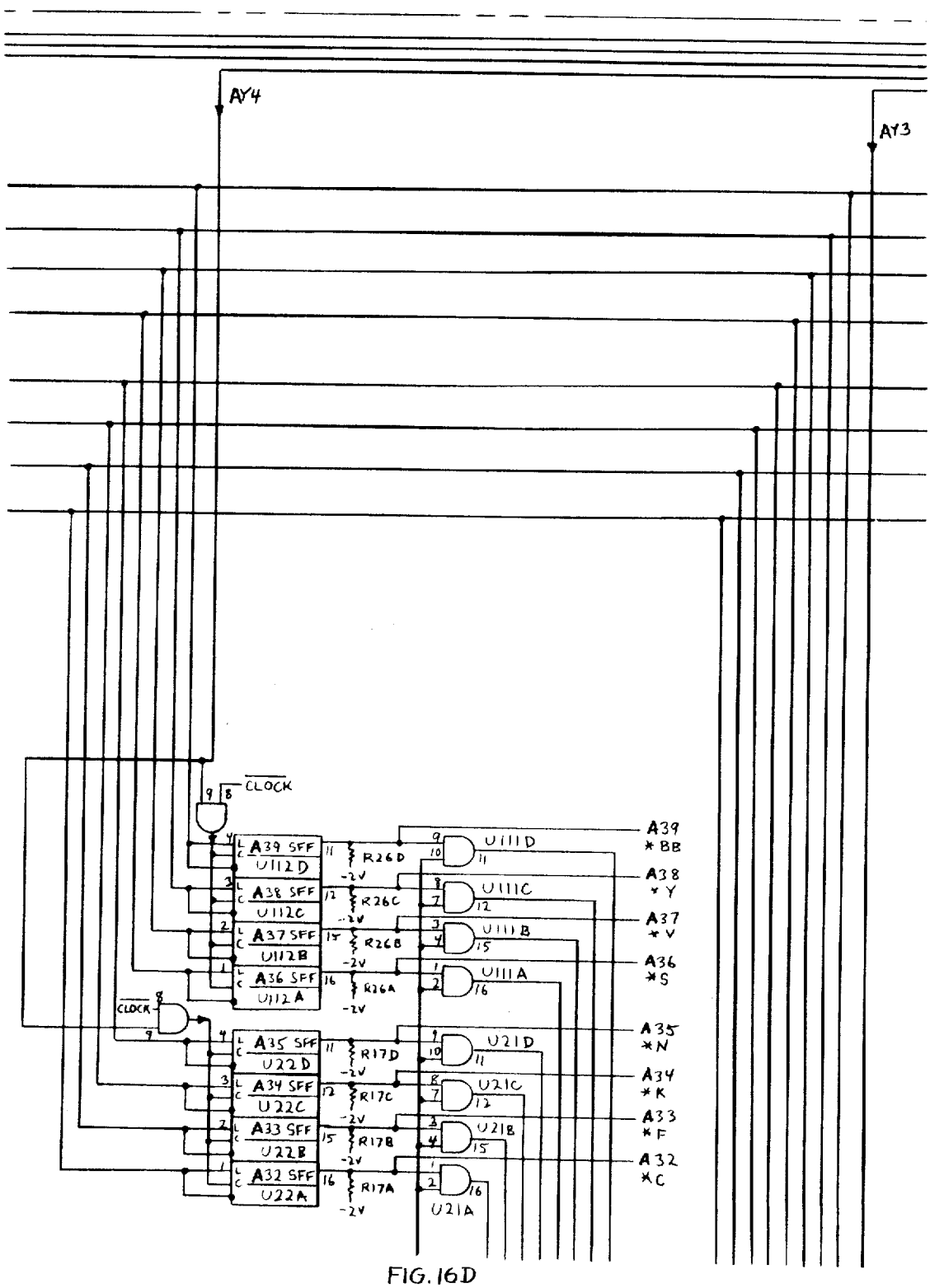

AY4
AY3
CLOCK
A39 SFF
U112D
A38 SFF
U112C
A37 SFF
U112B
A36 SFF
U112A
R26D
R26C
R26B
R26A
-2V
U111D
U111C
U111B
U111A
A39
*BB
A38
*Y
A37
*V
A36
*S
CLOCK
A35 SFF
U22D
A34 SFF
U22C
A33 SFF
U22B
A32 SFF
U22A
R17D
R17C
R17B
R17A
U21D
U21C
U21B
U21A
A35
*N
A34
*K
A33
*F
A32
*C

FIG. 16D

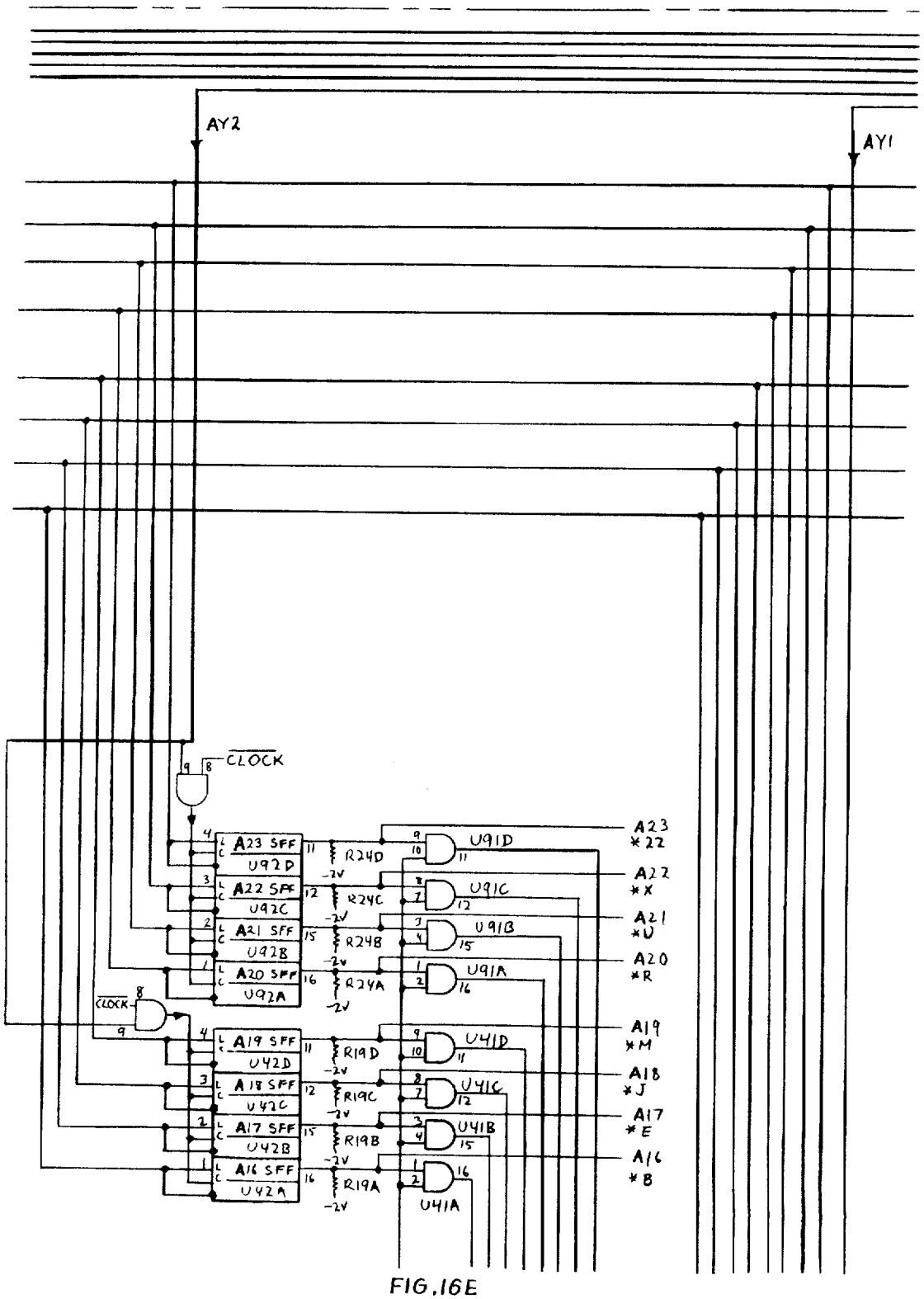

AY2
AY1
CLOCK
A23 SFF
U92D
A22 SFF
U92C
A21 SFF
U92B
A20 SFF
U92A
R24D
R24C
R24B
R24A
-2V
U91D
U91C
U91B
U91A
A23
*Z2
A22
*X
A21
*U
A20
*R
A19 SFF
U42D
A18 SFF
U42C
A17 SFF
U42B
A16 SFF
U42A
R19D
R19C
R19B
R19A
U41D
U41C
U41B
U41A
A19
*M
A18
*J
A17
*E
A16
*B

FIG. 16E

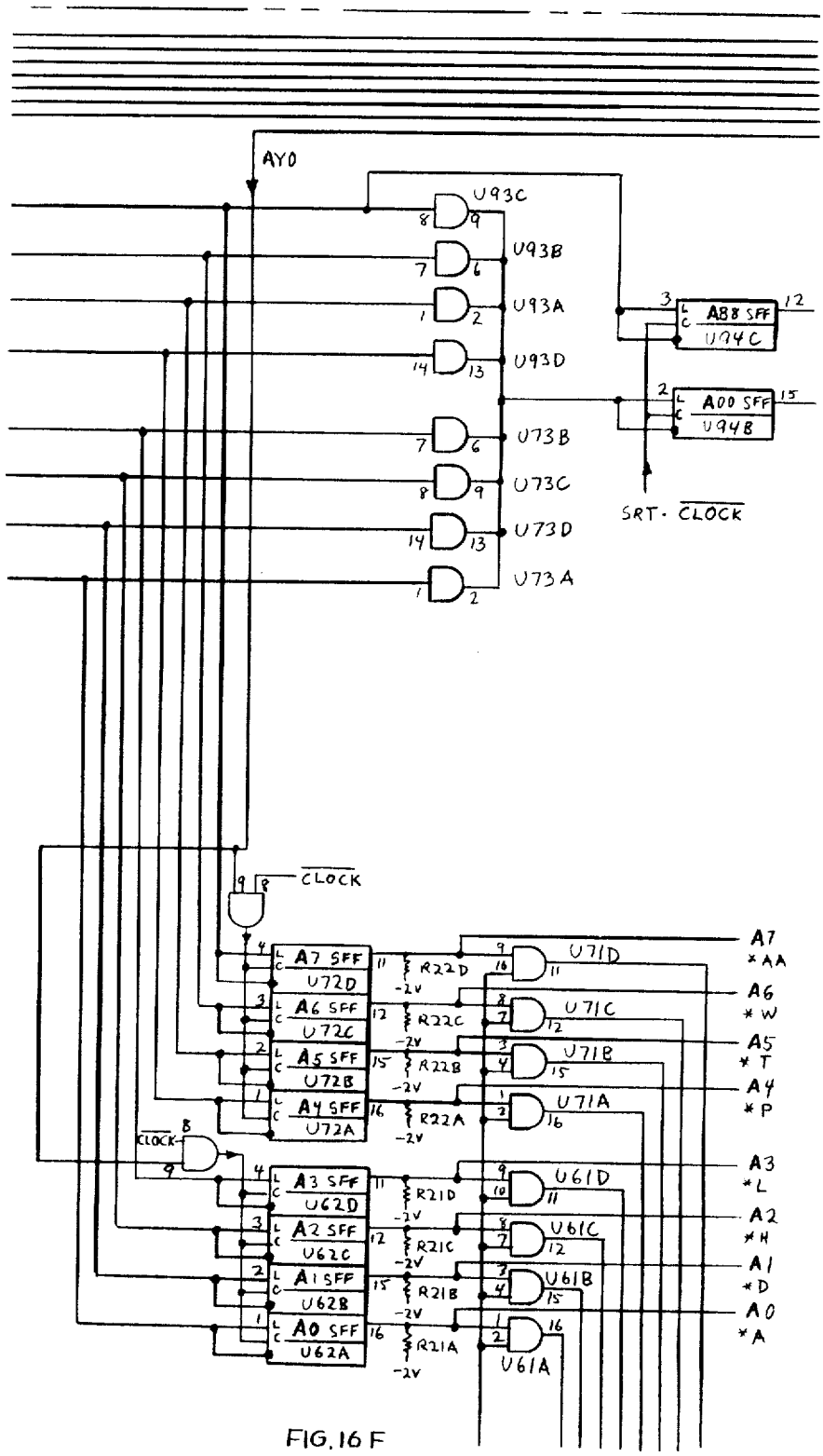

AY0
U93C
U93B
U93A
U93D
U73B
U73C
U73D
U73A
AB8 SFF
U94C
A00 SFF
U94B
SRT · CLOCK
CLOCK
A7 SFF
U72D
A6 SFF
U72C
A5 SFF
U72B
A4 SFF
U72A
A3 SFF
U62D
A2 SFF
U62C
A1 SFF
U62B
A0 SFF
U62A
R22D
R22C
R22B
R22A
R21D
R21C
R21B
R21A
-2V
U71D
U71C
U71B
U71A
U61D
U61C
U61B
U61A
A7 *AA
A6 *W
A5 *T
A4 *P
A3 *L
A2 *H
A1 *D
A0 *A

FIG. 16 F

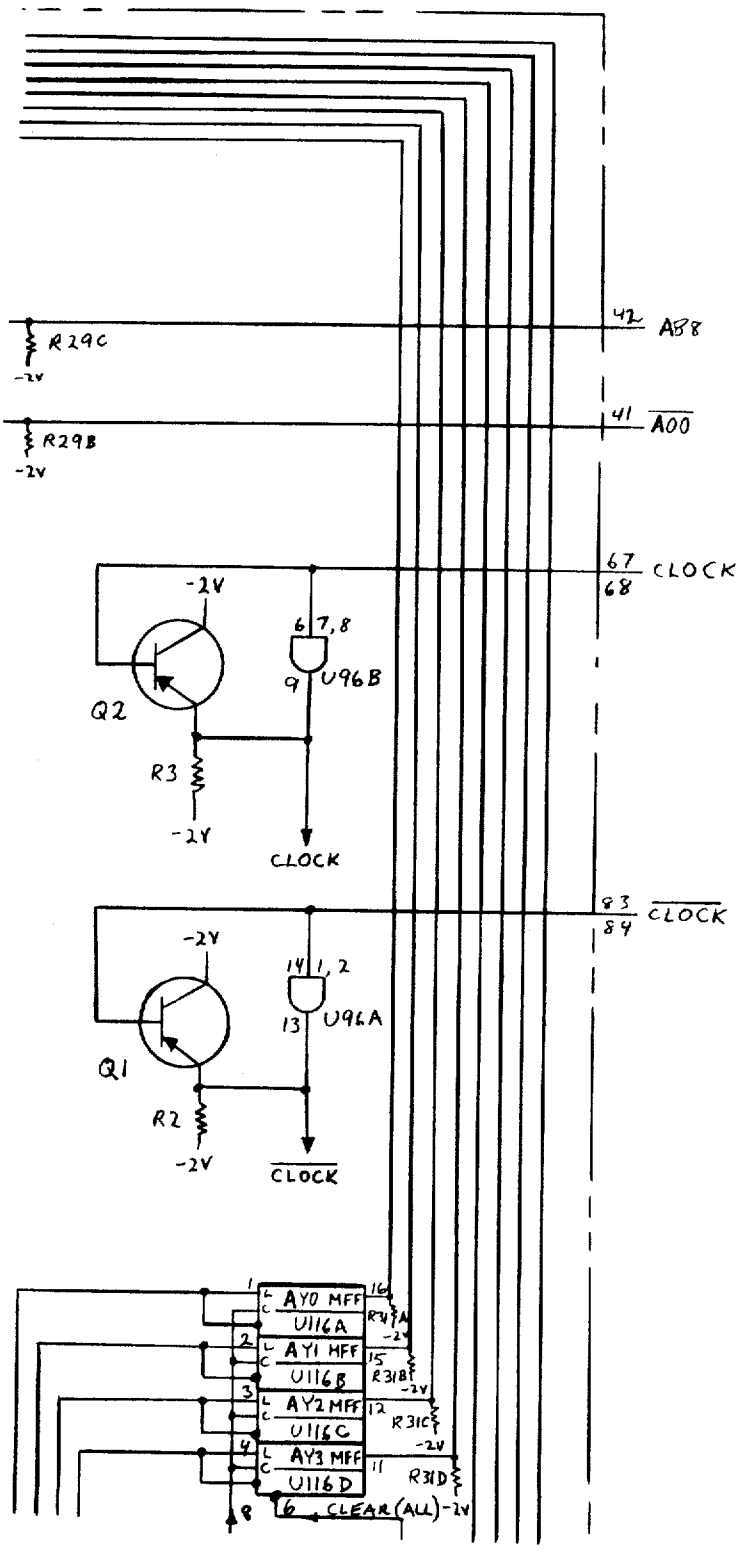

42 AB8
41 A00
67 CLOCK
68
83 CLOCK
84
R29C
-2V
R29B
-2V
-2V
Q2
R3
-2V
6 7,8
9 U96B
CLOCK
-2V
Q1
R2
-2V
14 1,2
13 U96A
CLOCK
AY0 MFF U116A
AY1 MFF U116B
AY2 MFF U116C
AY3 MFF U116D
R31A
R31B
R31C
R31D
-2V
CLEAR (ALL) -2V

FIG. 16G

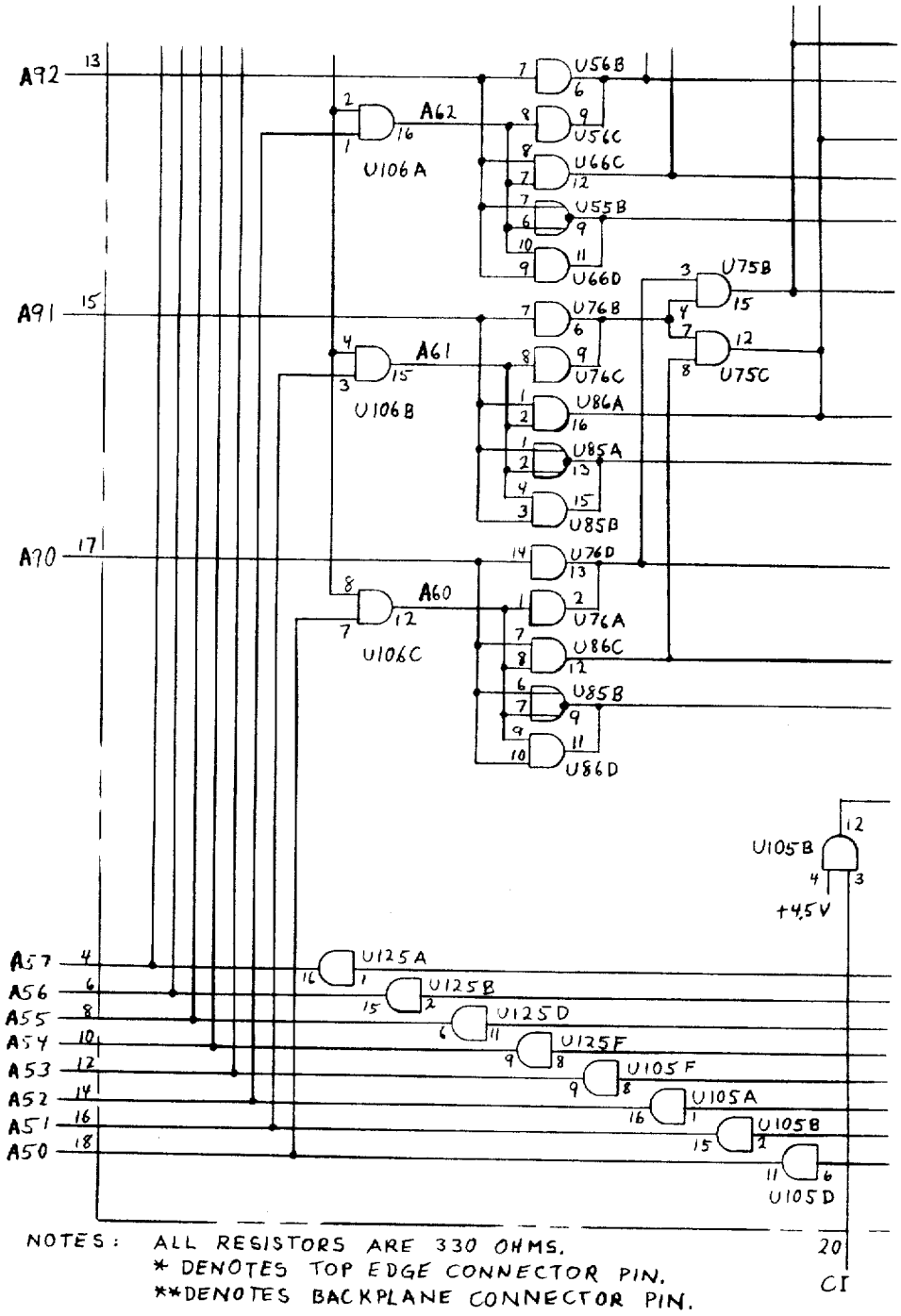

A92
A91
A90
A57
A56
A55
A54
A53
A52
A51
A50
A62
A61
A60
U106A
U106B
U106C
U56B
U56C
U66C
U55B
U66D
U76B
U76C
U86A
U85A
U85B
U76D
U76A
U86C
U85B
U86D
U75B
U75C
U105B
+4.5V
U125A
U125B
U125D
U125F
U105F
U105A
U105B
U105D
20
CI
NOTES: ALL RESISTORS ARE 330 OHMS.
* DENOTES TOP EDGE CONNECTOR PIN.
**DENOTES BACKPLANE CONNECTOR PIN.

FIG. 16 H

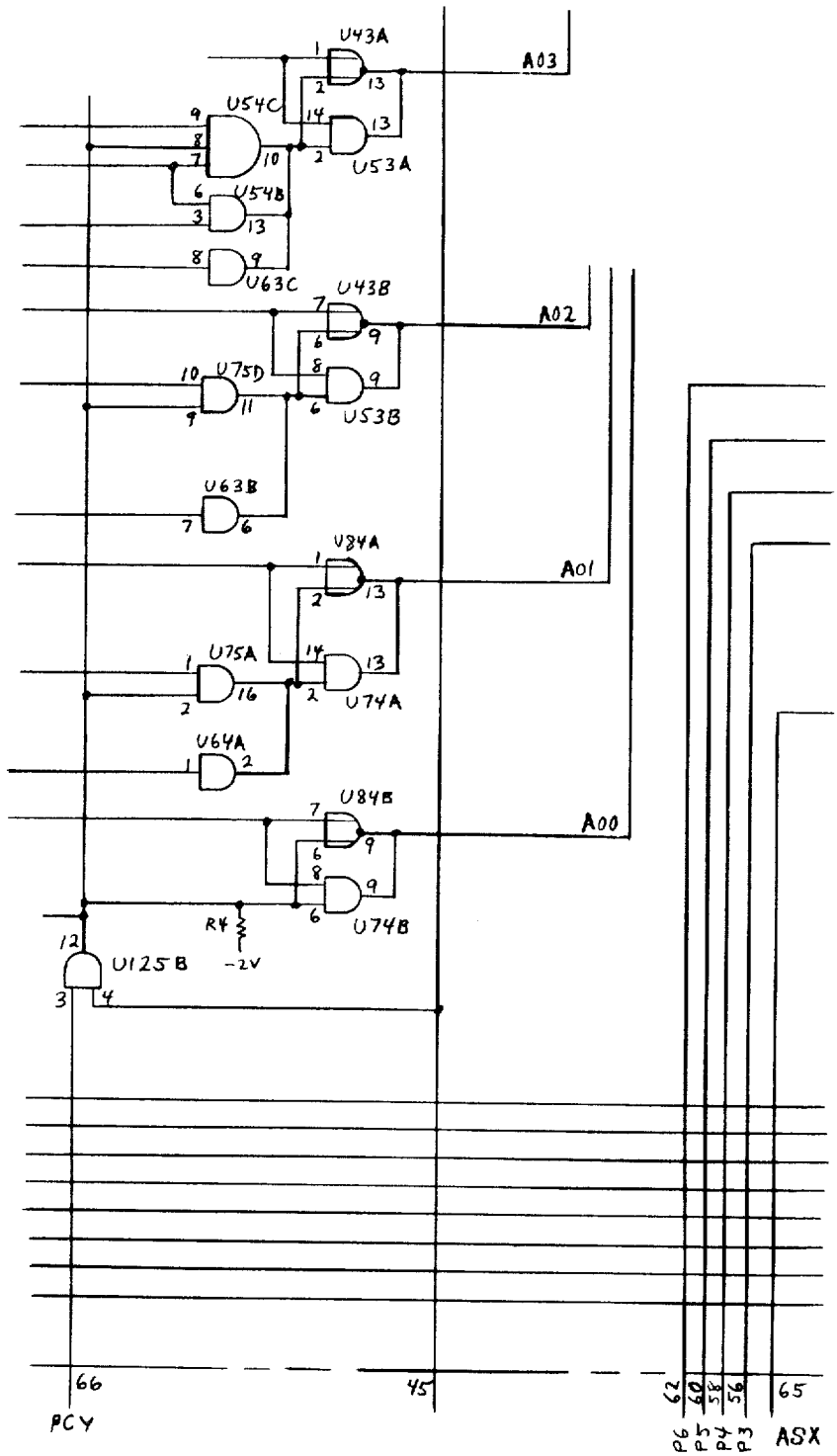

U43A
A03
U54C
U53A
U54B
U63C
U43B
A02
U75D
U53B
U63B
U84A
A01
U75A
U74A
U64A
U84B
A00
R4
-2V
U125B
U74B
66
PCY
45
62
60
58
56
65
P6
P5
P4
P3
ASX

FIG. 16 I

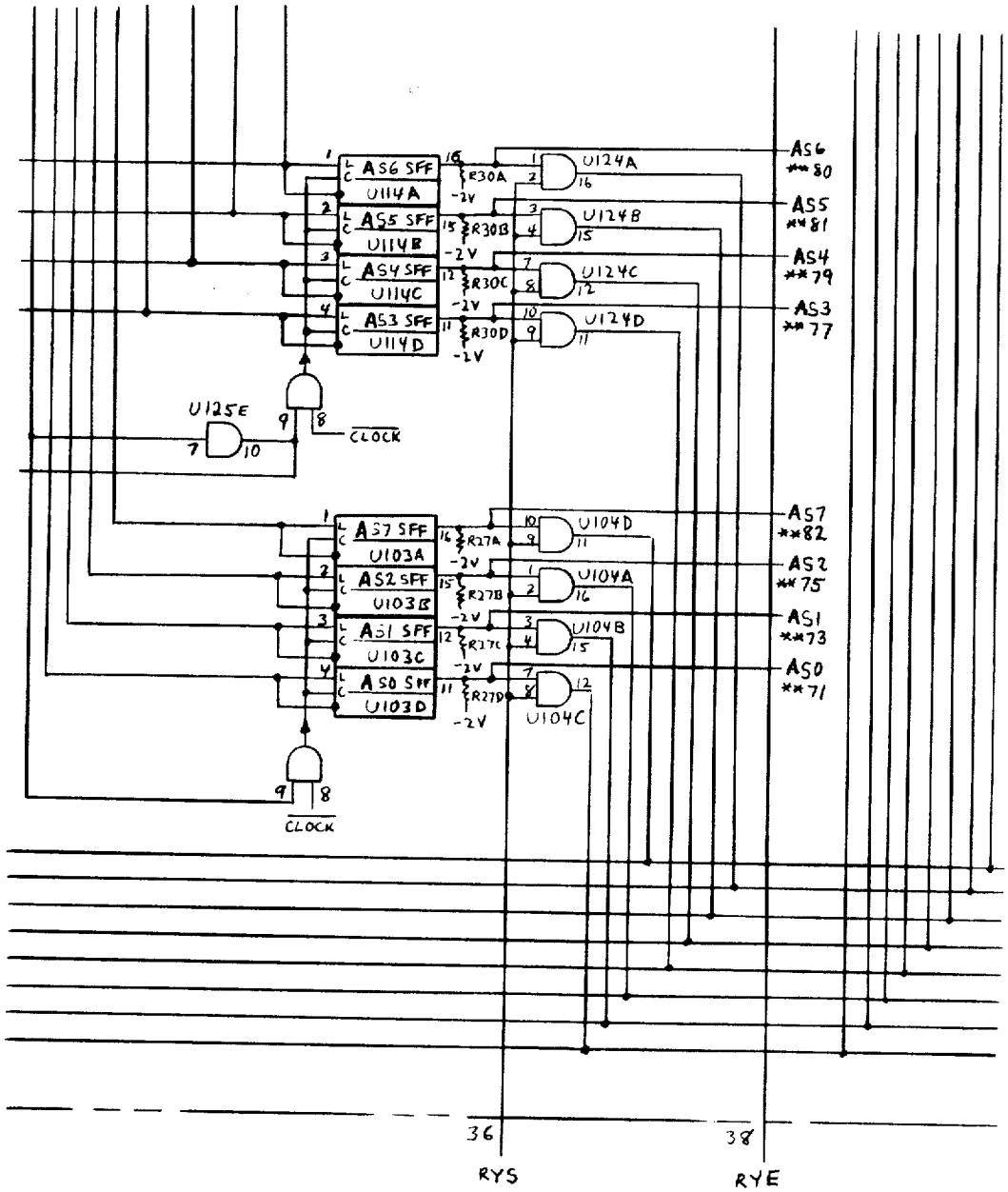

AS6 SFF
U114A
AS5 SFF
U114B
AS4 SFF
U114C
AS3 SFF
U114D
R30A
R30B
R30C
R30D
-2V
U124A
U124B
U124C
U124D
AS6 **80
AS5 **81
AS4 **79
AS3 **77
U125E
CLOCK
AS7 SFF
U103A
AS2 SFF
U103B
AS1 SFF
U103C
AS0 SFF
U103D
R27A
R27B
R27C
R27D
U104D
U104A
U104B
U104C
AS7 **82
AS2 **75
AS1 **73
AS0 **71
36
38
RYS
RYE

FIG. 16J

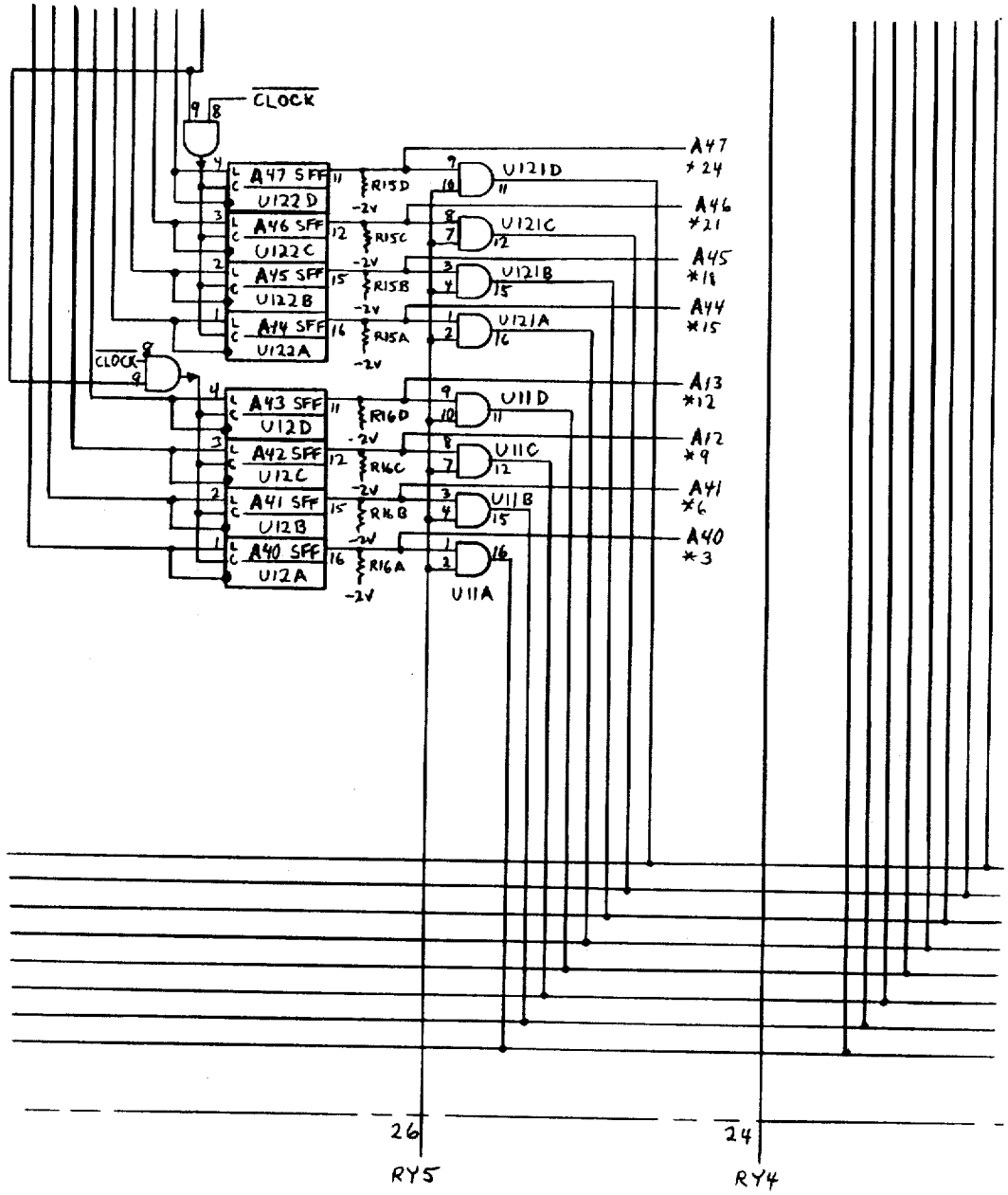

CLOCK
A47 SFF
U122D
A46 SFF
U122C
A45 SFF
U122B
A44 SFF
U122A
A43 SFF
U12D
A42 SFF
U12C
A41 SFF
U12B
A40 SFF
U12A
R15D
R15C
R15B
R15A
R16D
R16C
R16B
R16A
-2V
U121D
U121C
U121B
U121A
U11D
U11C
U11B
U11A
A47 *24
A46 *21
A45 *18
A44 *15
A13 *12
A12 *9
A41 *6
A40 *3
26
24
RY5
RY4

FIG. 16K

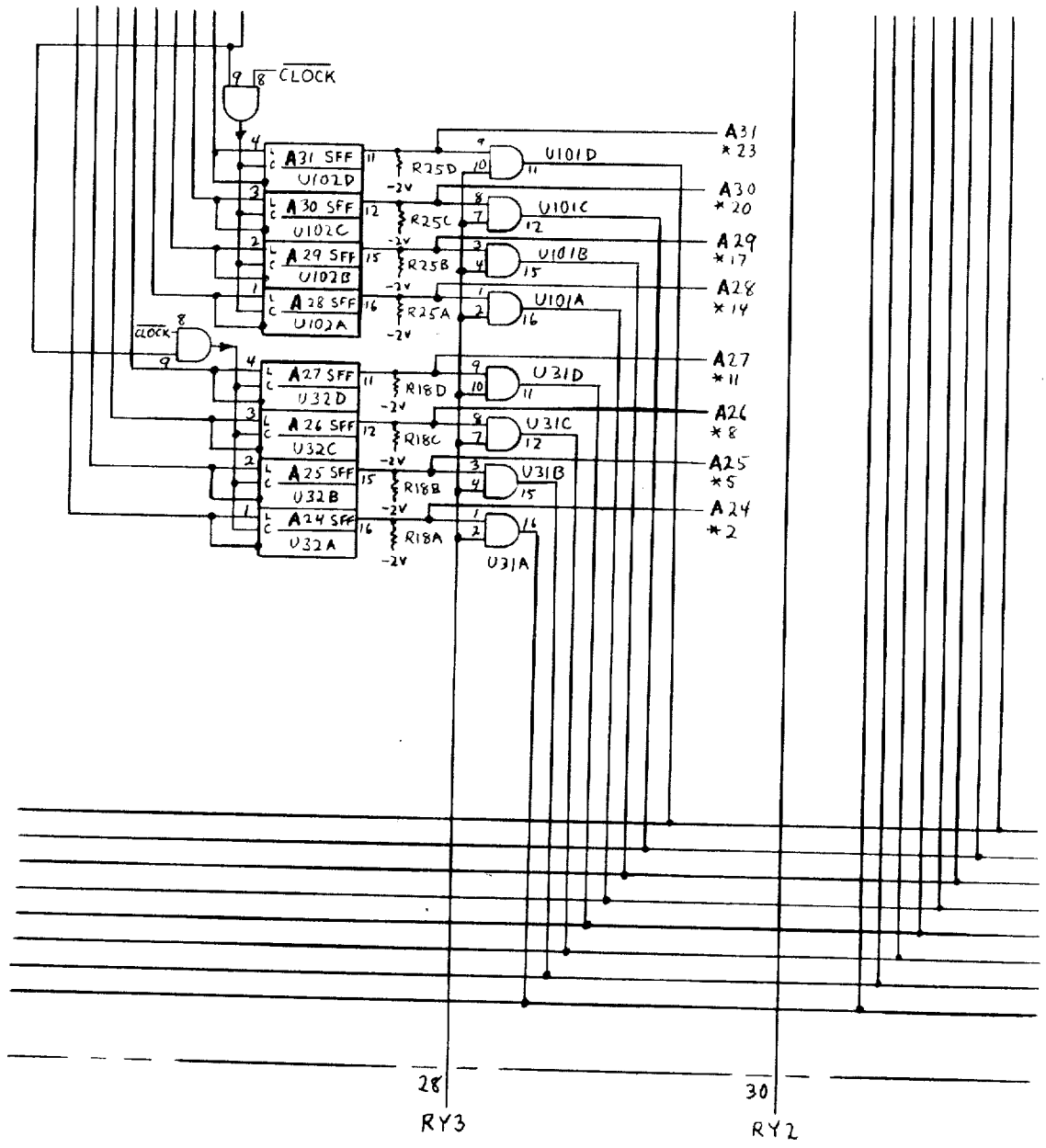

CLOCK
A31 SFF
U102D
A30 SFF
U102C
A29 SFF
U102B
A28 SFF
U102A
R25D
R25C
R25B
R25A
-2V
U101D
U101C
U101B
U101A
A31
*23
A30
*20
A29
*17
A28
*14
A27 SFF
U32D
A26 SFF
U32C
A25 SFF
U32B
A24 SFF
U32A
R18D
R18C
R18B
R18A
U31D
U31C
U31B
U31A
A27
*11
A26
*8
A25
*5
A24
*2
28
RY3
30
RY2

FIG. 16L

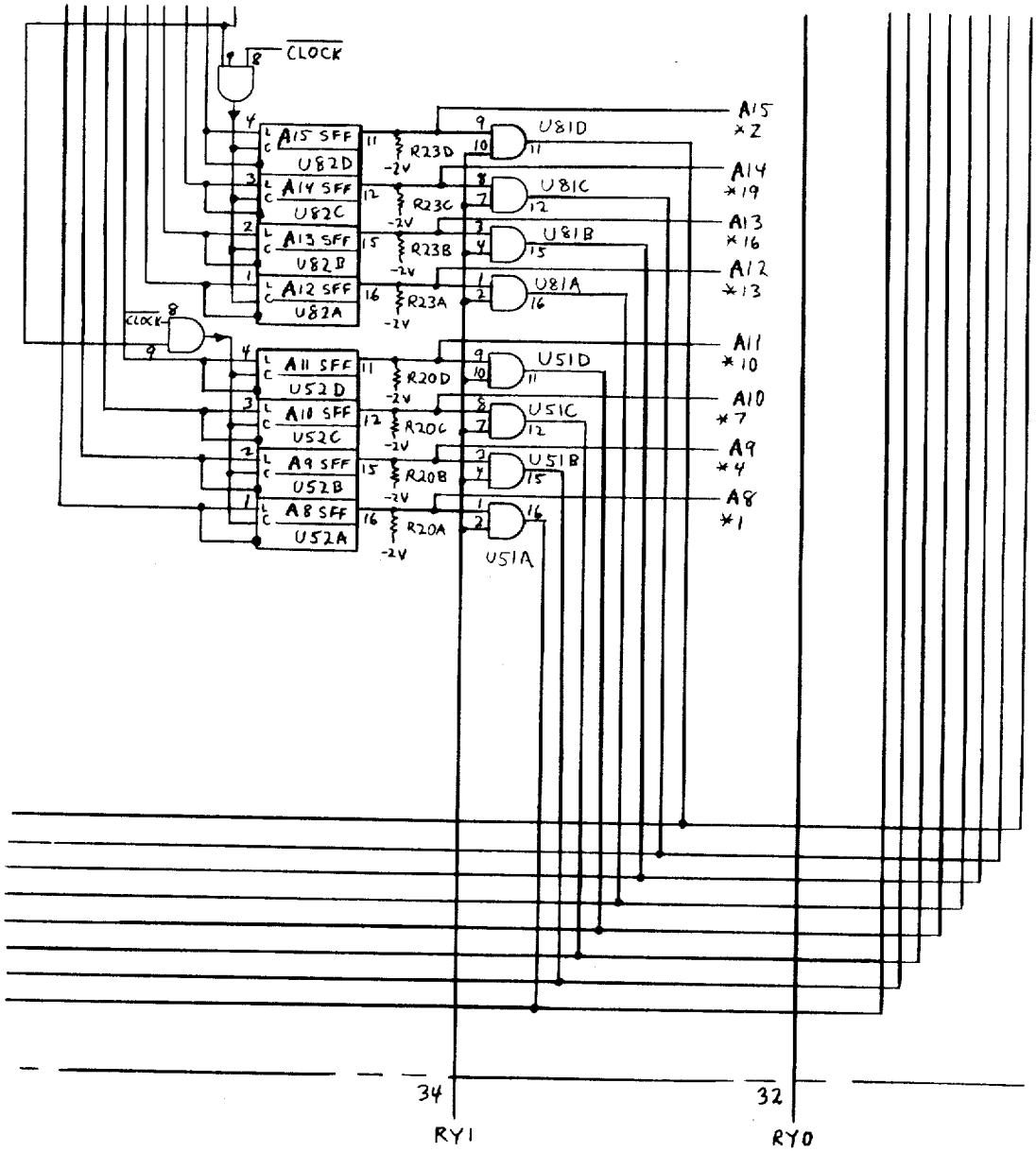

CLOCK
A15 SFF
U82D
A14 SFF
U82C
A13 SFF
U82B
A12 SFF
U82A
A11 SFF
U52D
A10 SFF
U52C
A9 SFF
U52B
A8 SFF
U52A
R23D
R23C
R23B
R23A
R20D
R20C
R20B
R20A
-2V
U81D
U81C
U81B
U81A
U51D
U51C
U51B
U51A
A15 *2
A14 *19
A13 *16
A12 *13
A11 *10
A10 *7
A9 *4
A8 *1
34
32
RY1
RY0

FIG. 16M

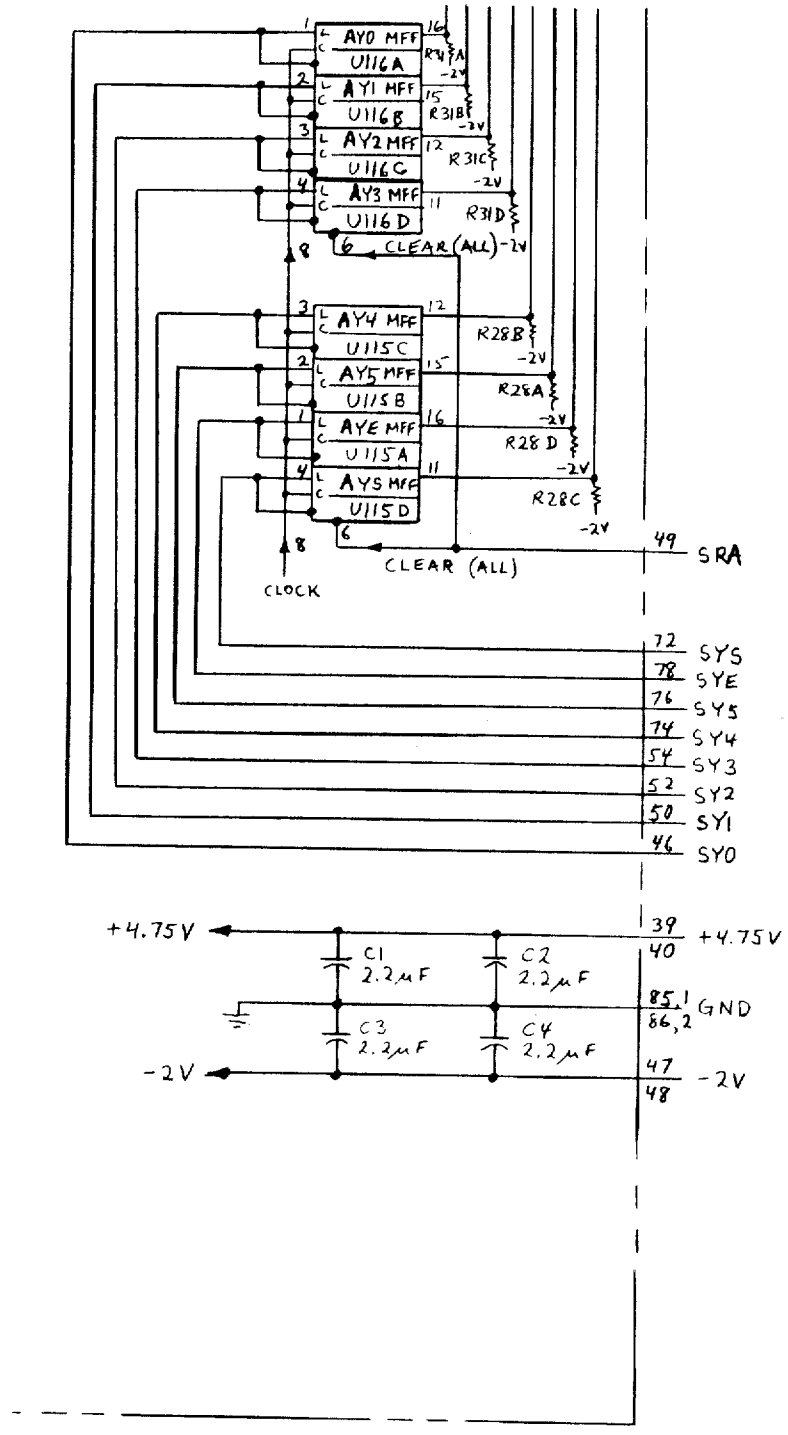

AY0 MFF
U116A
AY1 MFF
U116B
AY2 MFF
U116C
AY3 MFF
U116D
CLEAR (ALL) -2V
AY4 MFF
U115C
AY5 MFF
U115B
AYE MFF
U115A
AYS MFF
U115D
CLOCK
CLEAR (ALL)
R47A
R31B
R31C
R31D
R28B
R28A
R28D
R28C
-2V
49 SRA
72 SYS
78 SYE
76 SY5
74 SY4
54 SY3
52 SY2
50 SY1
46 SY0
+4.75V
C1 2.2μF
C2 2.2μF
C3 2.2μF
C4 2.2μF
39
40
+4.75V
85,1
86,2
GND
47
48
-2V

FIG. 16N

| | | | | | | |
|---|---|---|---|---|---|---|
| FIG.16A | FIG.16B | FIG.16C | FIG.16D | FIG.16E | FIG.16F | FIG.16G |
| FIG.16H | FIG.16I | FIG.16J | FIG.16K | FIG.16L | FIG.16M | FIG.16N |

FIG.17

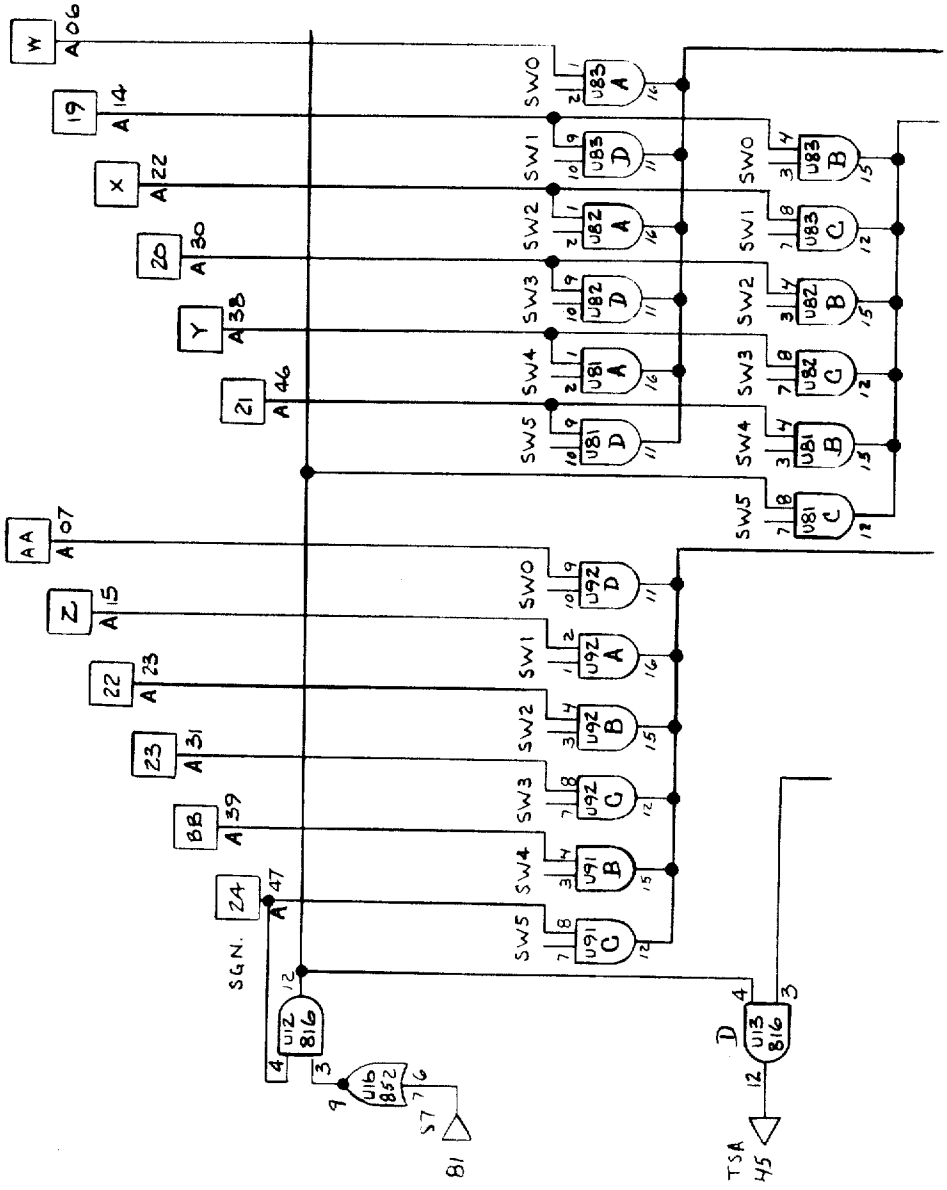

W A06
19 A14
X A22
20 A30
Y A38
21 A46
AA A07
Z A15
22 A23
23 A31
BB A39
24 A47
SGN.
SW0
SW1
SW2
SW3
SW4
SW5
U83 A
U83 D
U82 A
U82 D
U81 A
U81 D
U92 D
U92 A
U92 B
U92 C
U91 B
U91 C
U83 B
U83 C
U82 B
U82 C
U81 B
U81 C
U12 816
U13 816
U116 852
D
S7
81
TSA
45

FIG. 18A

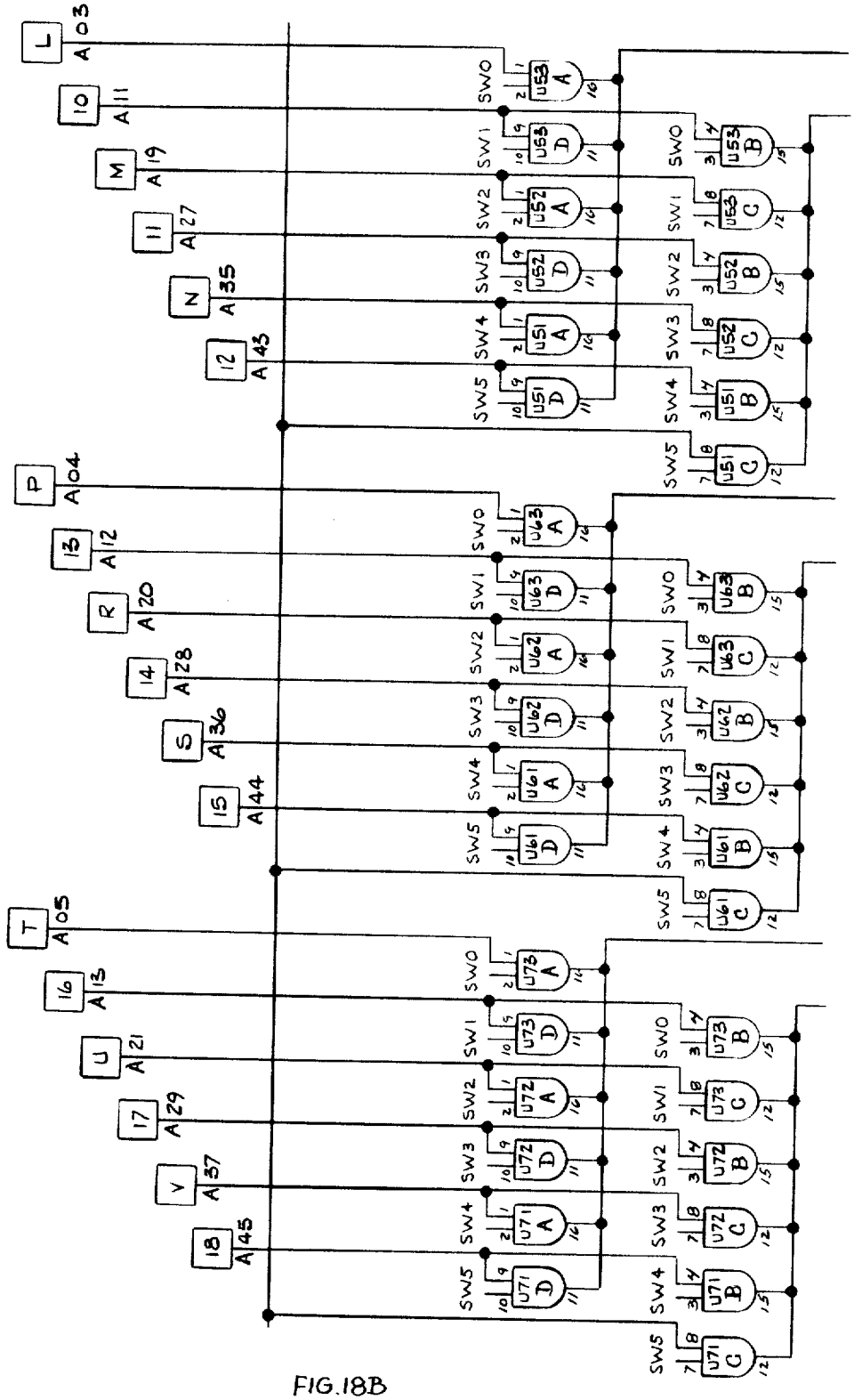

L A03
10 A11
M A19
11 A27
N A35
12 A43
P A04
13 A12
R A20
14 A28
S A36
15 A44
T A05
16 A13
U A21
17 A29
V A37
18 A45
SW0
SW1
SW2
SW3
SW4
SW5
U53 A
U53 B
U53 C
U53 D
U52 A
U52 B
U52 C
U52 D
U51 A
U51 B
U51 C
U51 D
U63 A
U63 B
U63 C
U63 D
U62 A
U62 B
U62 C
U62 D
U61 A
U61 B
U61 C
U61 D
U73 A
U73 B
U73 C
U73 D
U72 A
U72 B
U72 C
U72 D
U71 A
U71 B
U71 C
U71 D

FIG.18B

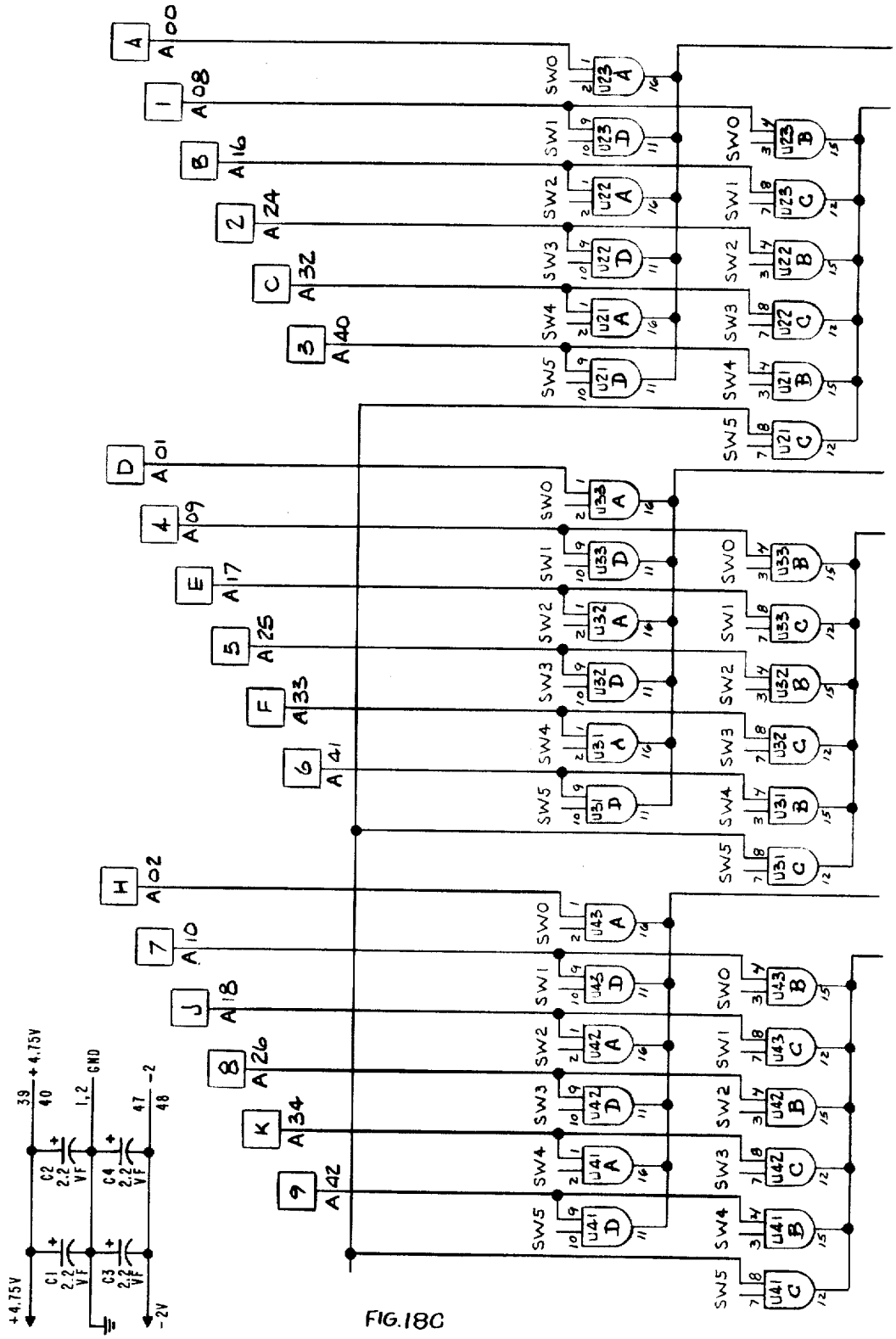
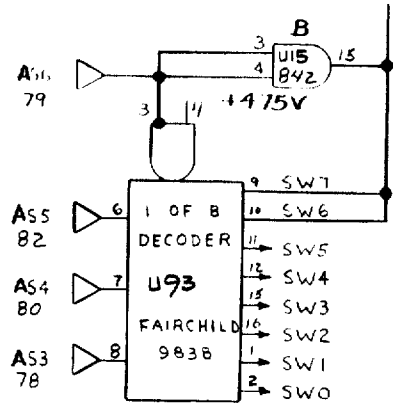

A 00
A 08
A 16
A 24
A 32
A 40
A 01
A 09
A 17
A 25
A 33
A 41
A 02
A 10
A 18
A 26
A 34
A 42
SW0
SW1
SW2
SW3
SW4
SW5
+4.75V
GND
-2V
C1 2.2 VF
C2 2.2 VF
C3 2.2 VF
C4 2.2 VF

FIG. 18C


B
U15
842
+4.75V
AS6
79
AS5
82
AS4
80
AS3
78
1 OF 8
DECODER
U93
FAIRCHILD
9838
SW7
SW6
SW5
SW4
SW3
SW2
SW1
SW0

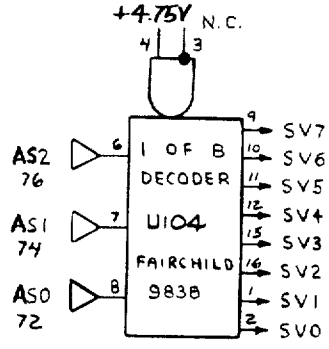

+4.75V
N.C.
AS2
76
AS1
74
AS0
72
1 OF 8
DECODER
U104
FAIRCHILD
9838
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0

FIG. 18D

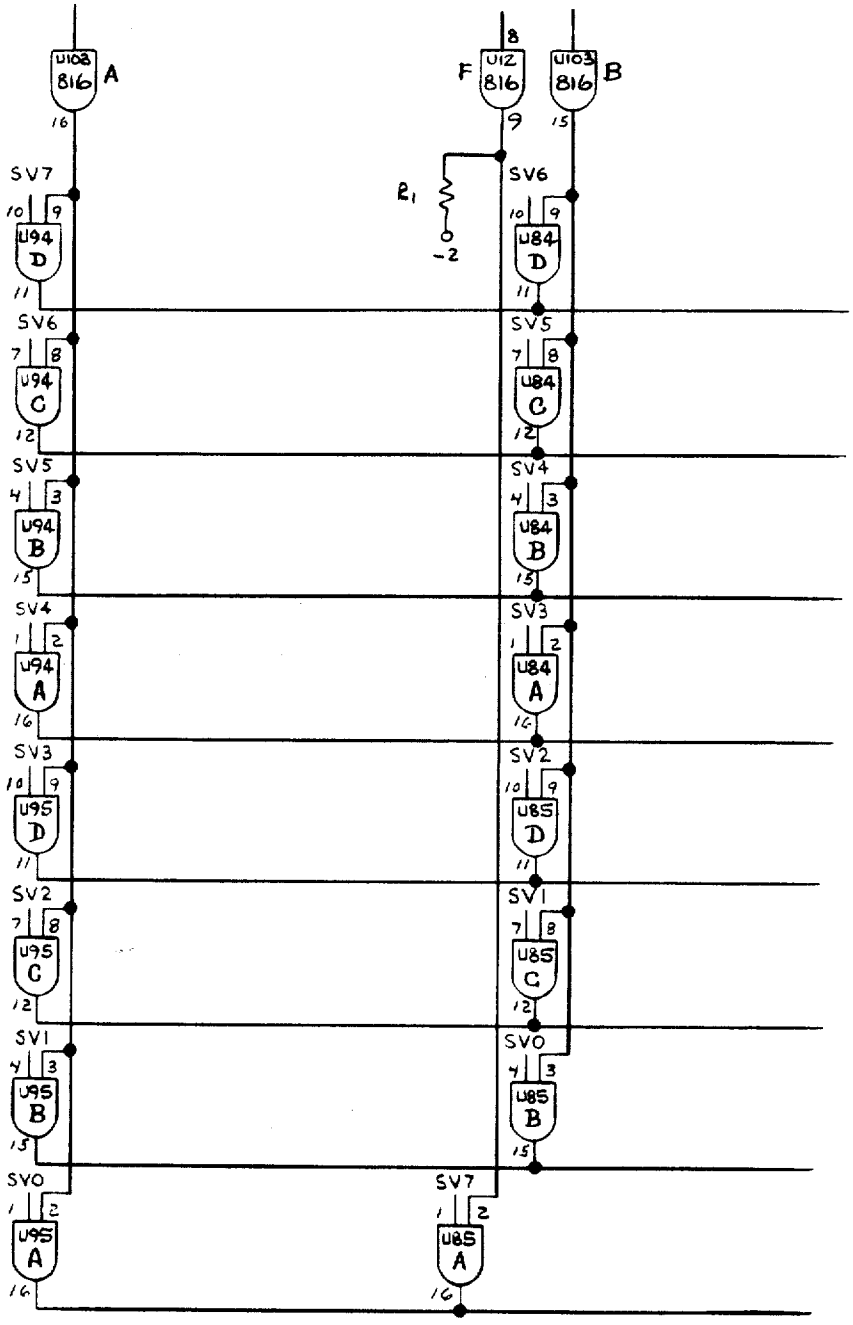

U108 816 A
U12 816 F
U103 816 B
R1
-2
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0
U94 D
U94 C
U94 B
U94 A
U95 D
U95 C
U95 B
U95 A
U84 D
U84 C
U84 B
U84 A
U85 D
U85 C
U85 B
U85 A

FIG. 18E

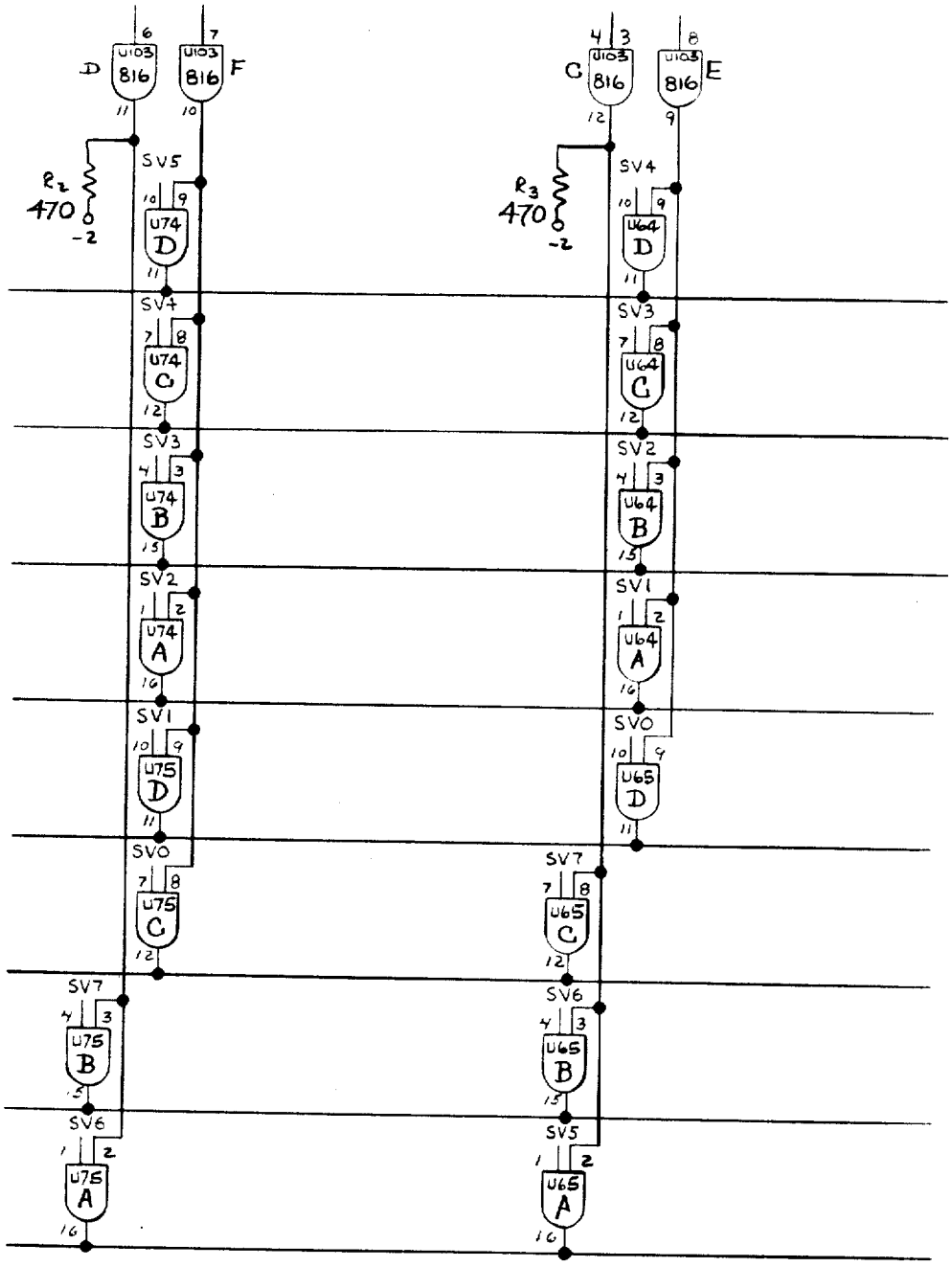

U103 816 D
U103 816 F
U103 816 G
U103 816 E
R2 470 -2
R3 470 -2
SV5 U74 D
SV4 U74 C
SV3 U74 B
SV2 U74 A
SV1 U75 D
SV0 U75 C
SV7 U75 B
SV6 U75 A
SV4 U64 D
SV3 U64 C
SV2 U64 B
SV1 U64 A
SV0 U65 D
SV7 U65 C
SV6 U65 B
SV5 U65 A

FIG. 18F

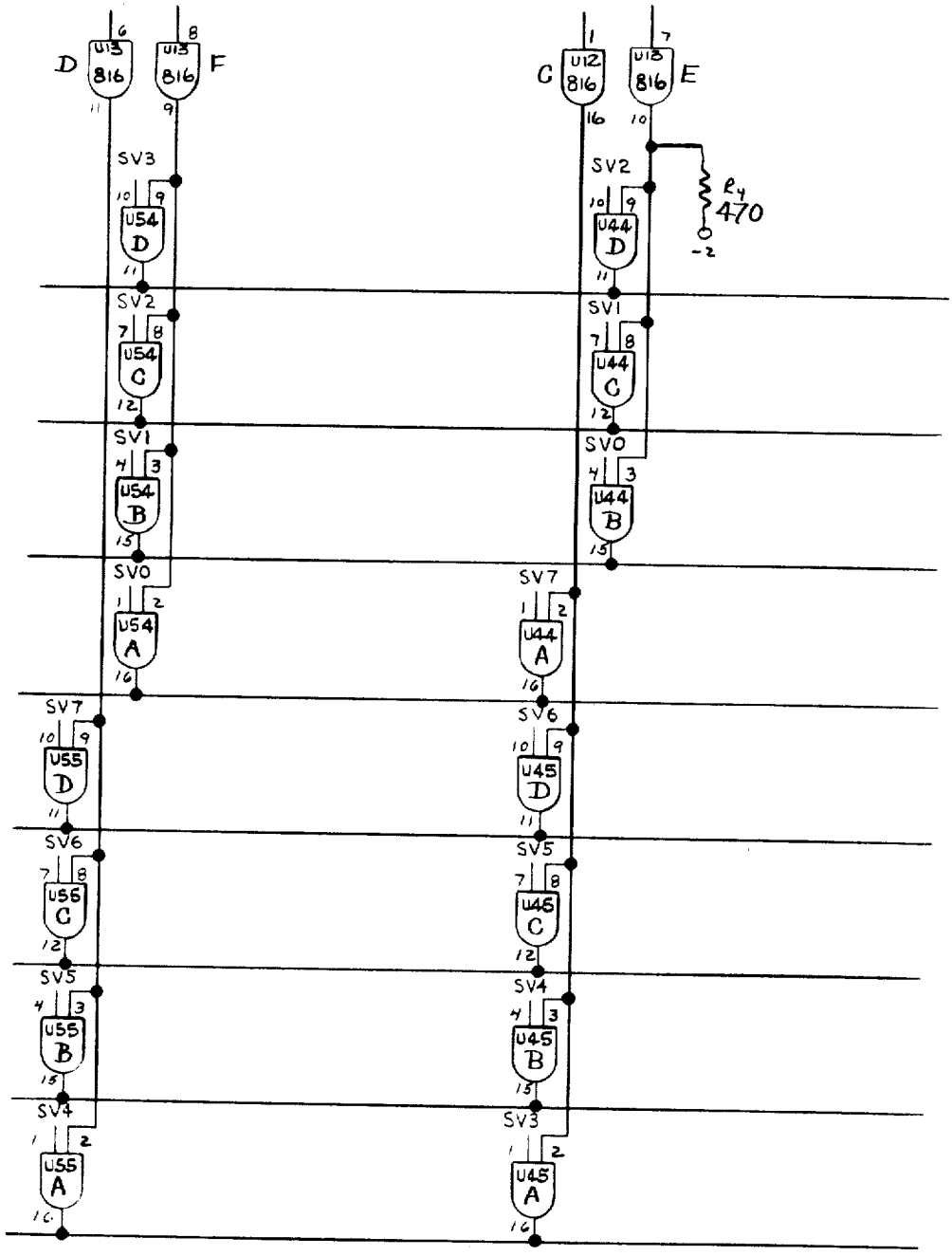

D
U13 816
F
C
U12 816
E
SV3
SV2
SV1
SV0
SV7
SV6
SV5
SV4
U54 D
U54 C
U54 B
U54 A
U55 D
U55 C
U55 B
U55 A
U44 D
U44 C
U44 B
U44 A
U45 D
U45 C
U45 B
U45 A
R4 470
-2

FIG. 18G

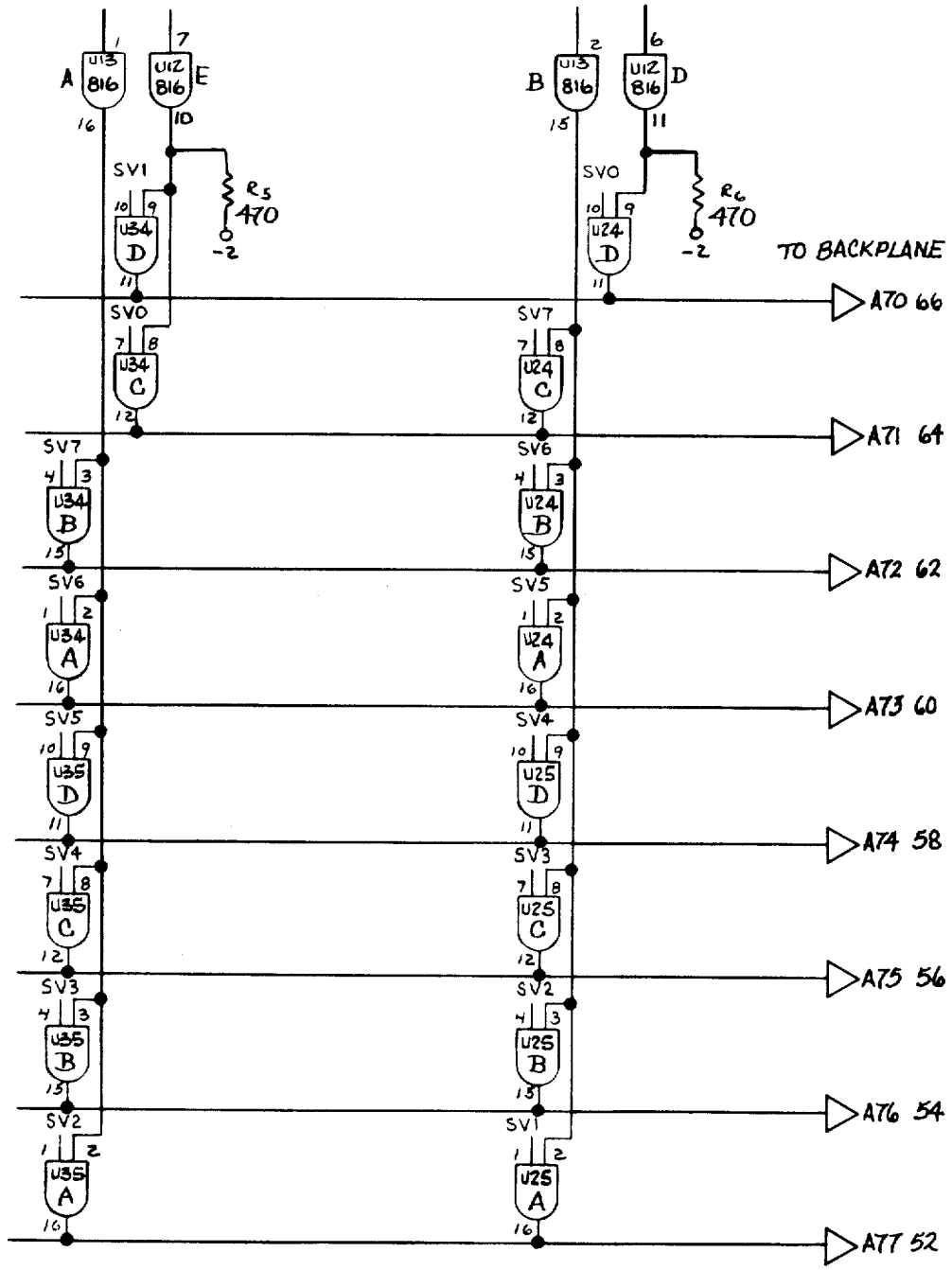

A
B
U13 816
U12 816
E
D
SV1
SV0
SV7
SV6
SV5
SV4
SV3
SV2
R5 470
R6 470
-2
U34
U24
U35
U25
TO BACKPLANE
A70 66
A71 64
A72 62
A73 60
A74 58
A75 56
A76 54
A77 52

FIG. 18H

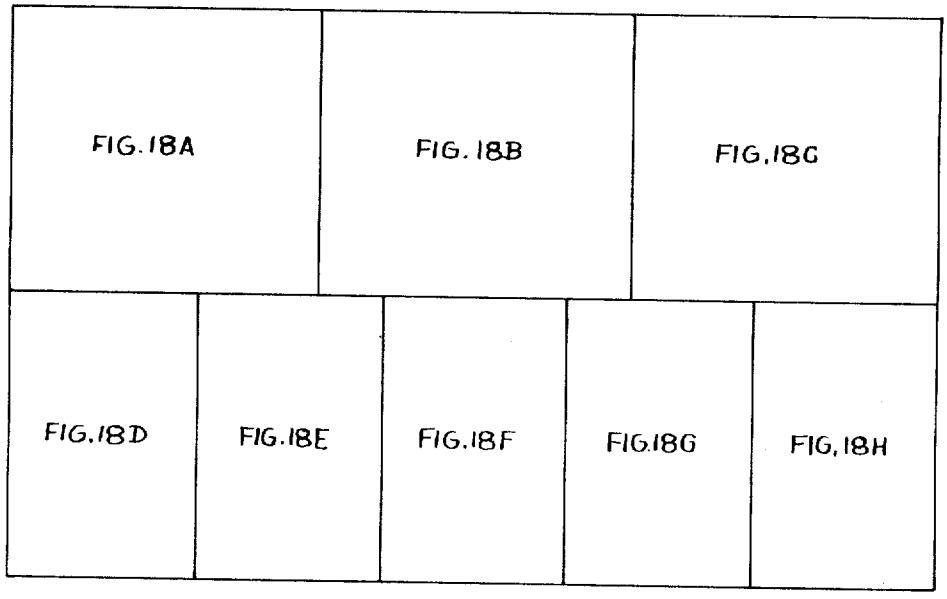



FIG. 19

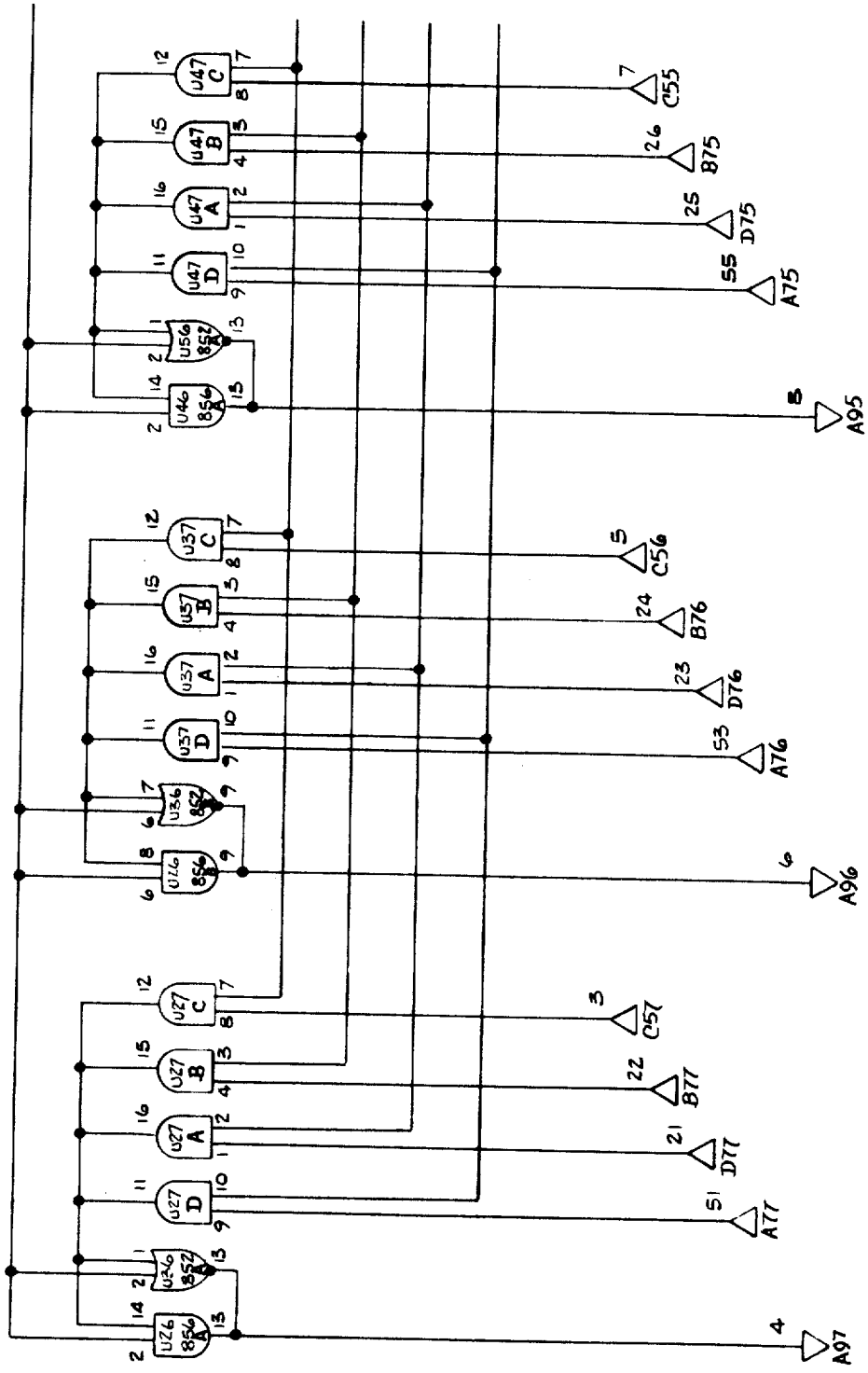

U47 C
U47 B
U47 A
U47 D
U56 B52
U46 B56 A
U37 C
U37 B
U37 A
U37 D
U36 B52
U46 B56 B
U27 C
U27 B
U27 A
U27 D
U26 B52 A
U26 B56 A
C55
B75
D75
A75
A95
C56
B76
D76
A76
A96
C57
B77
D77
A77
A97

FIG 20A

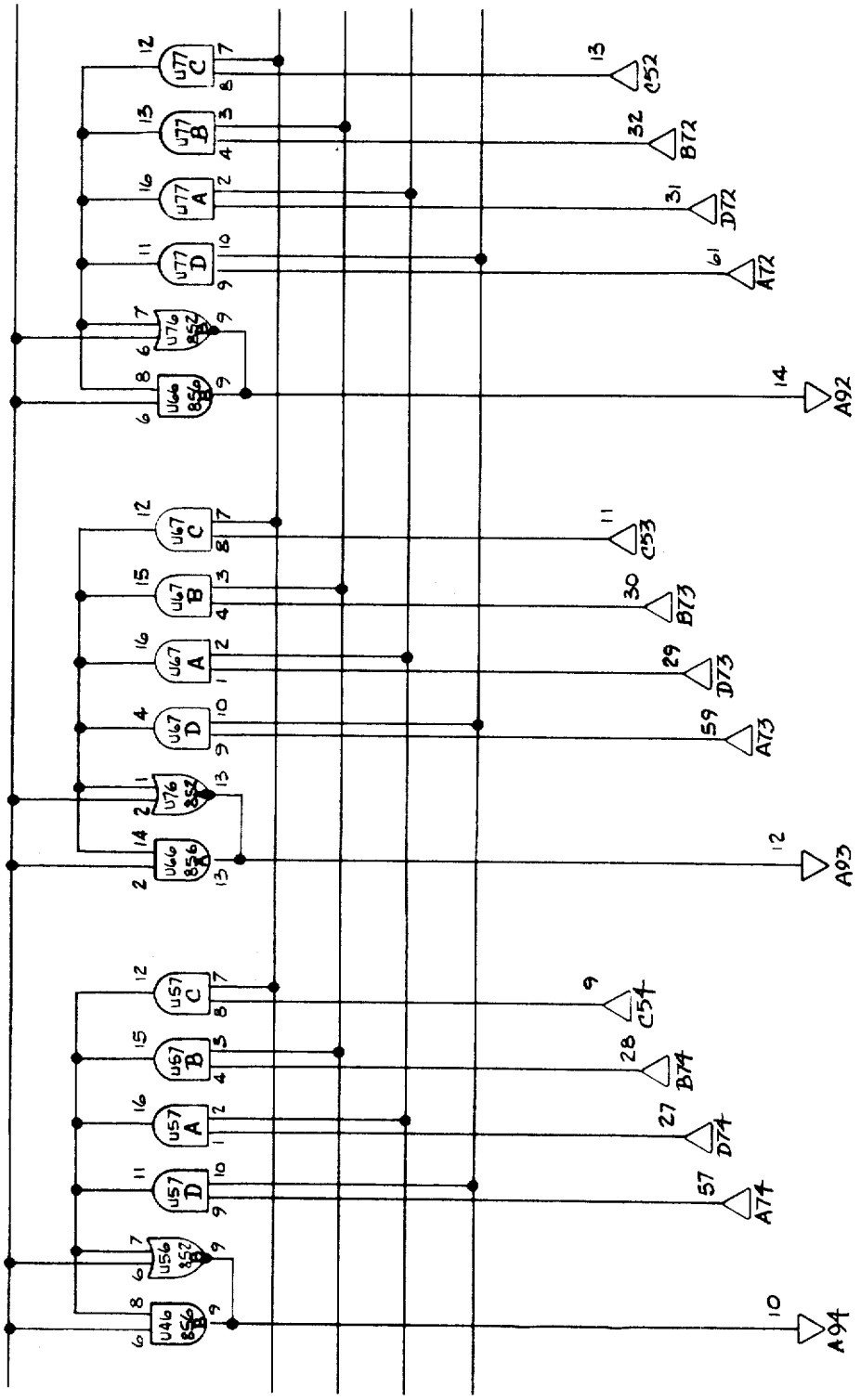

U77 C
U77 B
U77 A
U77 D
U76 B
U66 B
U67 C
U67 B
U67 A
U67 D
U76 A
U66 A
U57 C
U57 B
U57 A
U57 D
U56 B
U46 B
C52
B72
D72
A72
A92
C53
B73
D73
A73
A93
C54
B74
D74
A74
A94

FIG. 20B

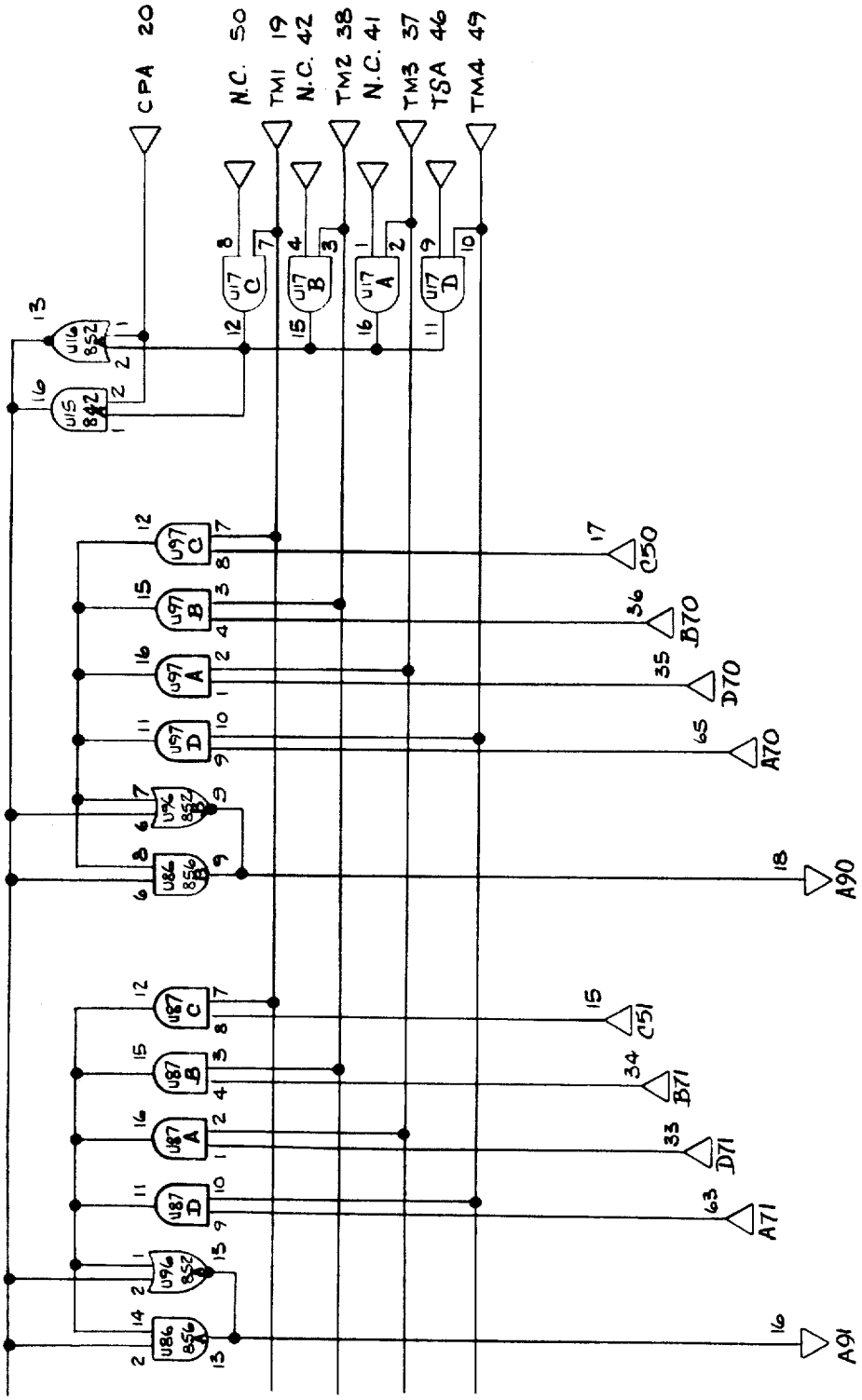

CPA 20
N.C. 50
TM1 19
N.C. 42
TM2 38
N.C. 41
TM3 37
TSA 46
TM4 49
U17 C
U17 B
U17 A
U17 D
U16 852
U15 842
U97 C
U97 B
U97 A
U97 D
U96 852
U86 856
U87 C
U87 B
U87 A
U87 D
U96 852
U86 856
17 C50
36 B70
35 D70
65 A70
18 A90
15 C51
34 B71
33 D71
63 A71
16 A91

FIG. 20C

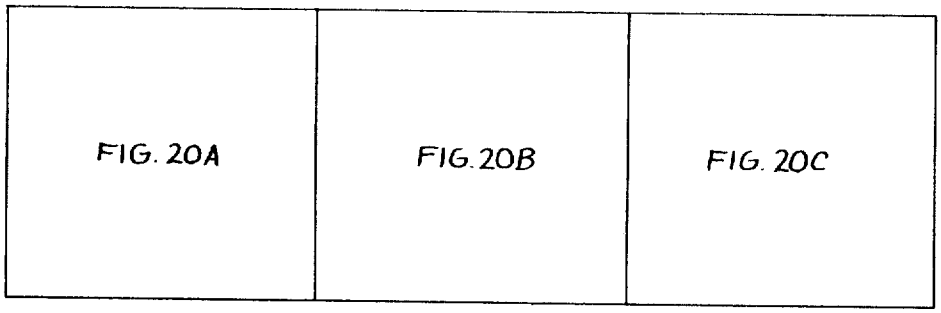



FIG. 21

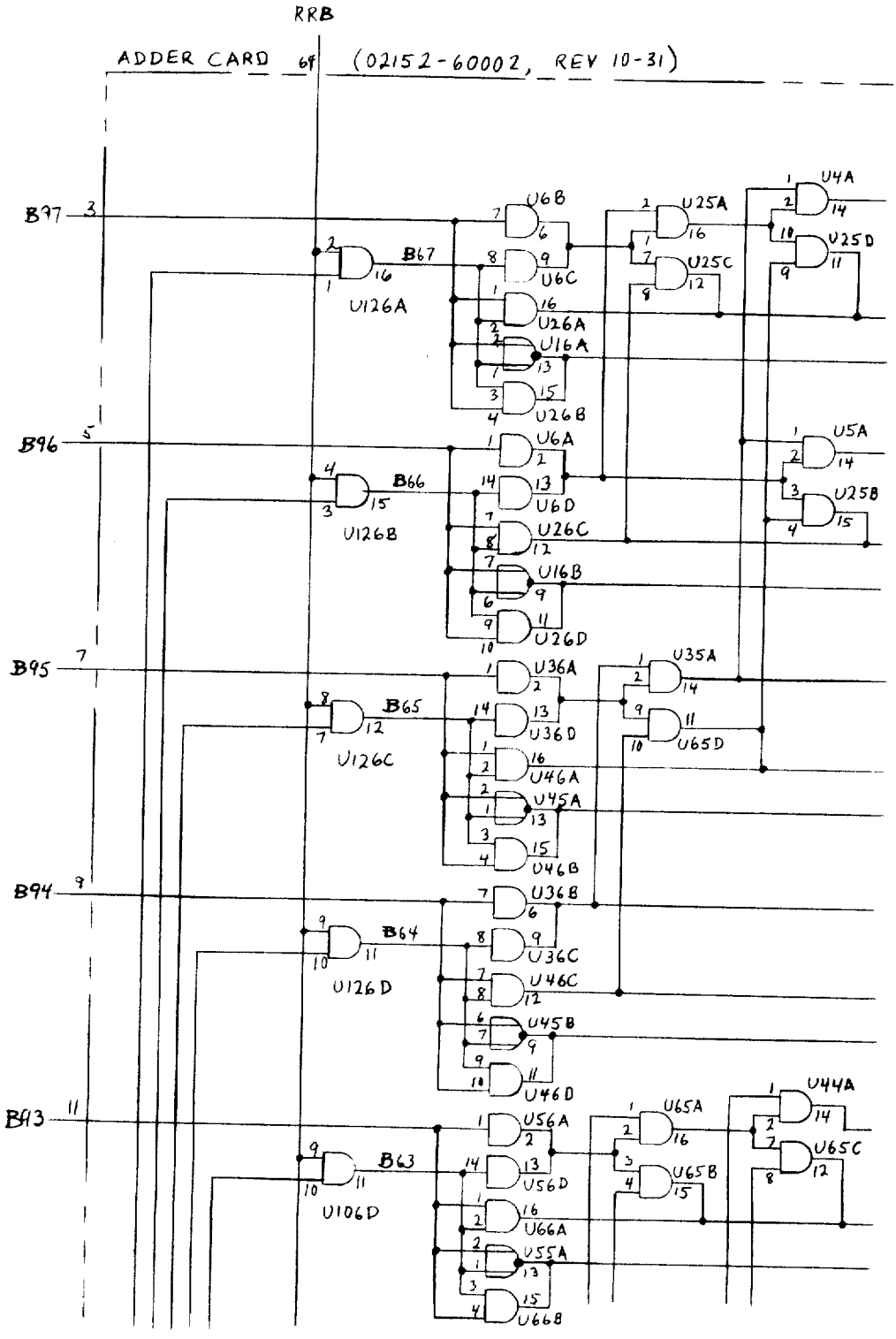

RRB
ADDER CARD 6f (02152-60002, REV 10-31)
B97
B96
B95
B94
B93
B67
B66
B65
B64
B63
U126A
U126B
U126C
U126D
U106D
U6B
U6C
U26A
U16A
U26B
U6A
U6D
U26C
U16B
U26D
U36A
U36D
U46A
U45A
U46B
U36B
U36C
U46C
U45B
U46D
U56A
U56D
U66A
U55A
U66B
U25A
U25C
U4A
U25D
U5A
U25B
U35A
U65D
U65A
U65B
U44A
U65C

FIG. 22A

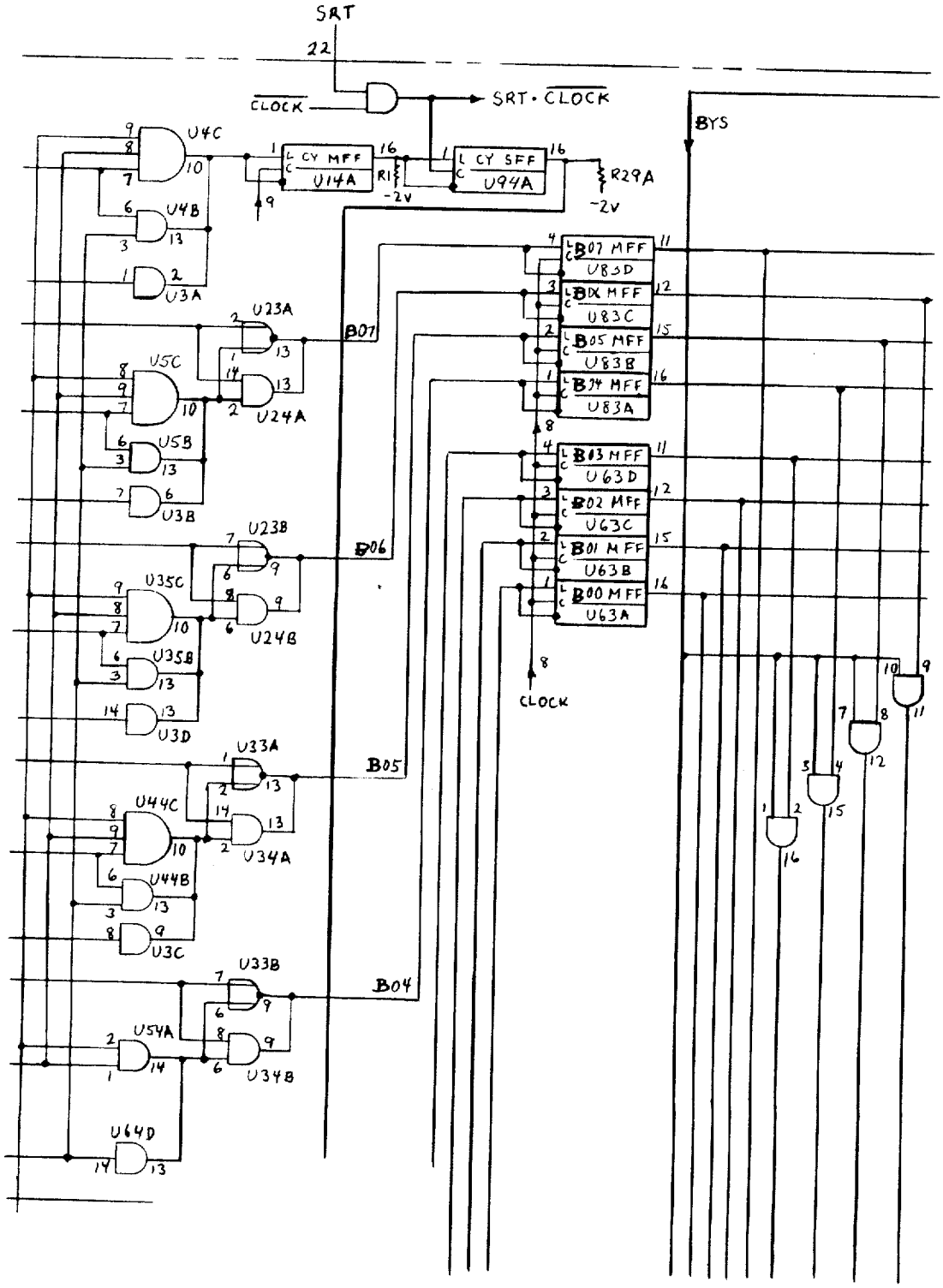

SRT
22
CLOCK
SRT · CLOCK
BYS
U4C
U4B
U3A
U23A
U5C
U24A
U5B
U3B
U23B
U35C
U24B
U35B
U3D
U33A
U44C
U34A
U44B
U3C
U33B
U54A
U34B
U64D
L CY MFF
U14A
L CY SFF
U94A
R1
-2V
R29A
-2V
B07
B06
B05
B04
B07 MFF
U83D
B06 MFF
U83C
B05 MFF
U83B
B04 MFF
U83A
B03 MFF
U63D
B02 MFF
U63C
B01 MFF
U63B
B00 MFF
U63A
CLOCK

FIG. 22B

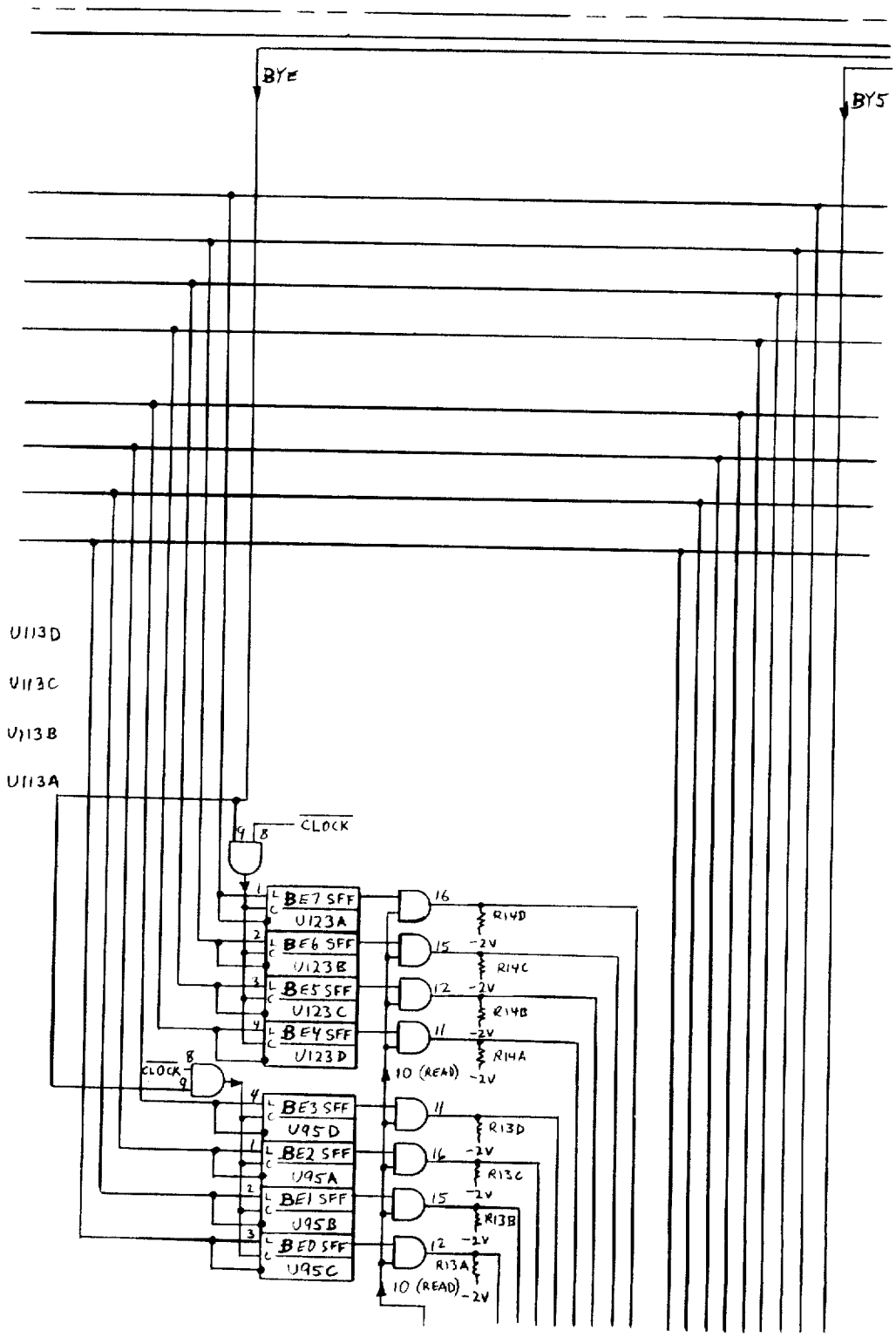

BYE
BY5
U113D
U113C
U113B
U113A
CLOCK
BE7 SFF
U123A
BE6 SFF
U123B
BE5 SFF
U123C
BE4 SFF
U123D
BE3 SFF
U95D
BE2 SFF
U95A
BE1 SFF
U95B
BED SFF
U95C
10 (READ)
R14D
R14C
R14B
R14A
R13D
R13C
R13B
R13A
-2V

FIG.22C

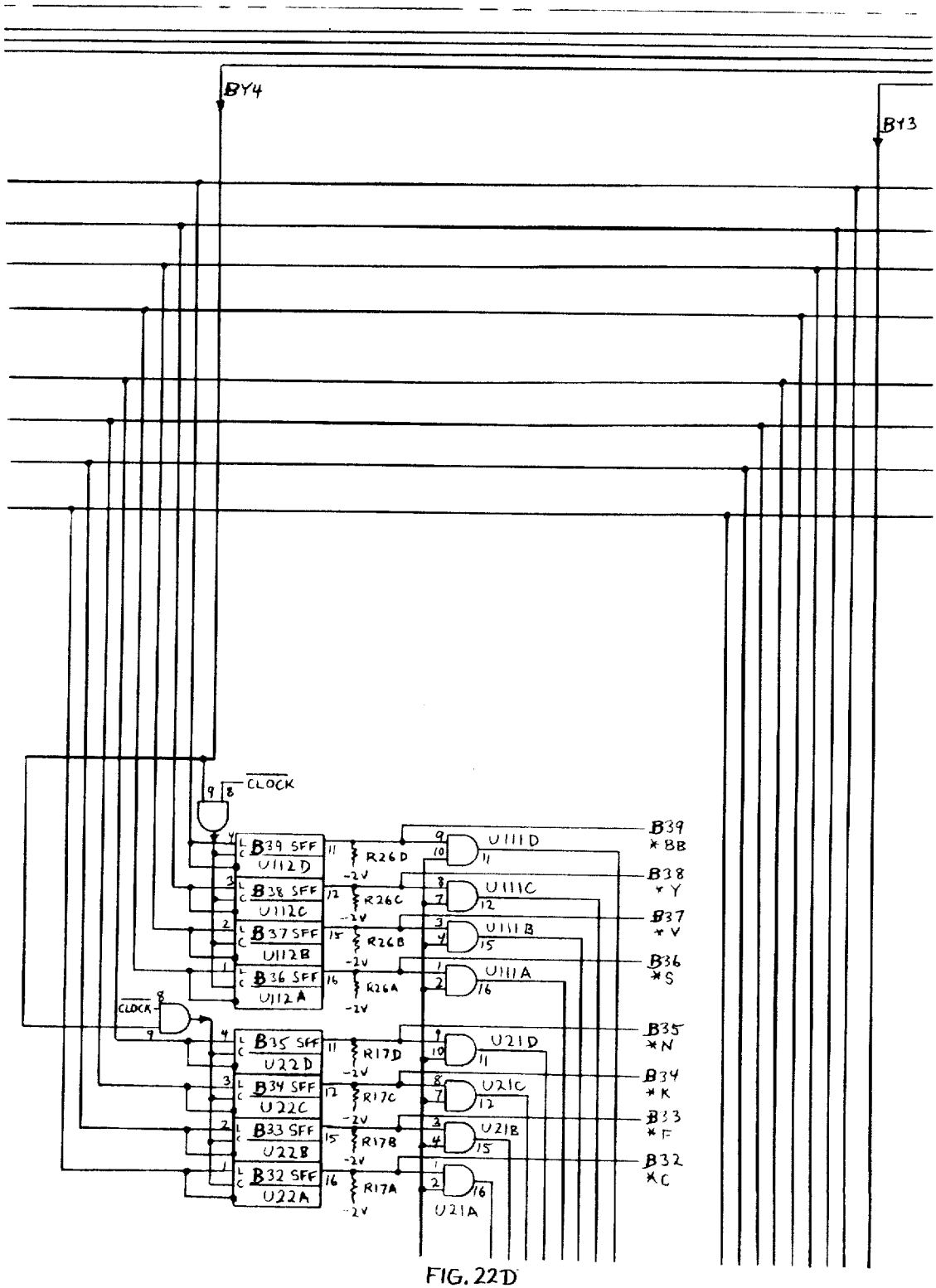

BY4
BY3
CLOCK
B39 SFF
U112D
B38 SFF
U112C
B37 SFF
U112B
B36 SFF
U112A
R26D
R26C
R26B
R26A
-2V
U111D
U111C
U111B
U111A
B39
*BB
B38
*Y
B37
*V
B36
*S
CLOCK
B35 SFF
U22D
B34 SFF
U22C
B33 SFF
U22B
B32 SFF
U22A
R17D
R17C
R17B
R17A
U21D
U21C
U21B
U21A
B35
*N
B34
*K
B33
*F
B32
*C

FIG. 22D

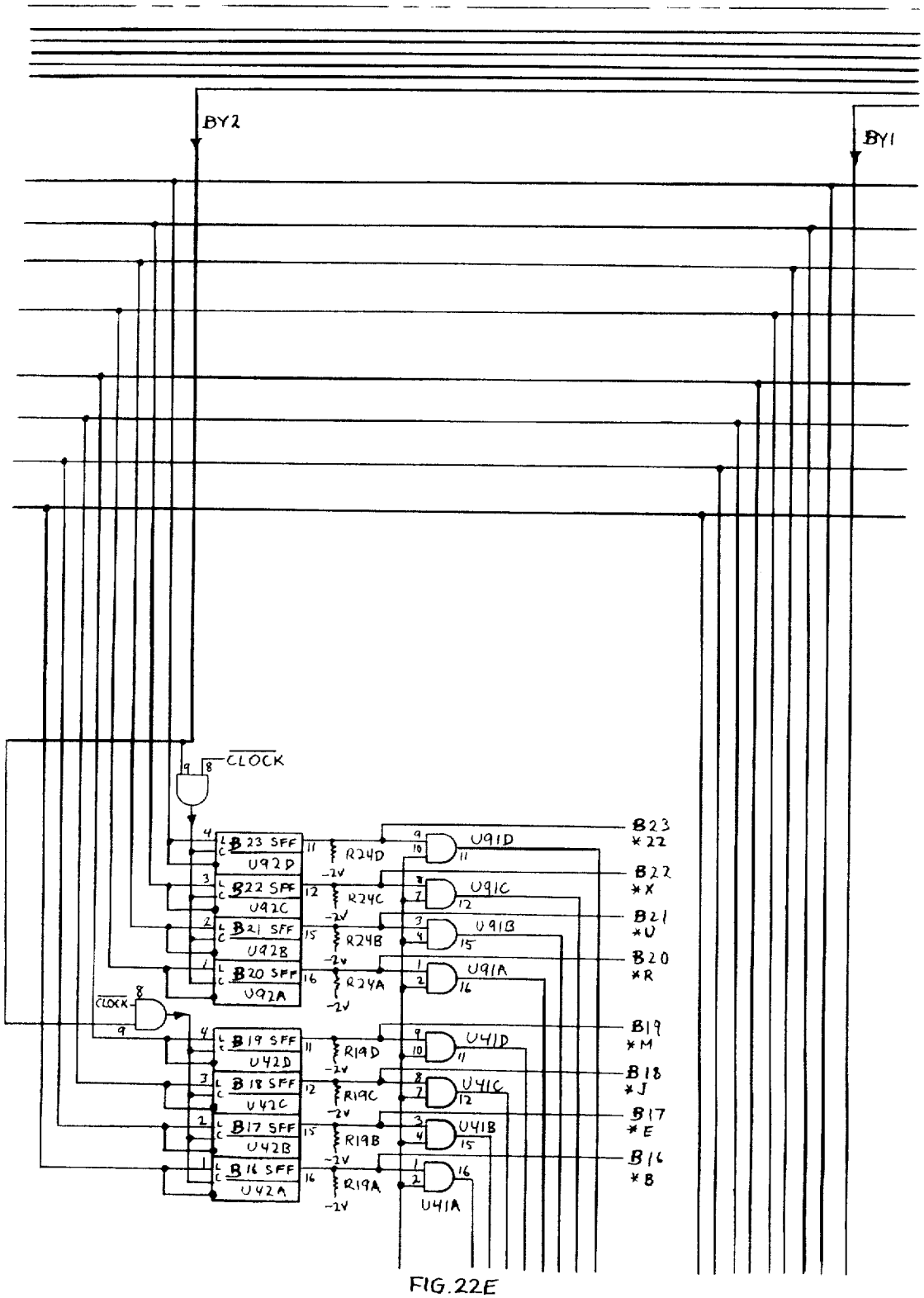

BY2
BY1
CLOCK
B 23 SFF
U92D
B 22 SFF
U92C
B 21 SFF
U92B
B 20 SFF
U92A
R24D
R24C
R24B
R24A
-2V
U91D
U91C
U91B
U91A
B23
*22
B22
*X
B21
*U
B20
*R
CLOCK
B 19 SFF
U42D
B 18 SFF
U42C
B 17 SFF
U42B
B 16 SFF
U42A
R19D
R19C
R19B
R19A
U41D
U41C
U41B
U41A
B19
*M
B18
*J
B17
*E
B16
*B

FIG. 22E

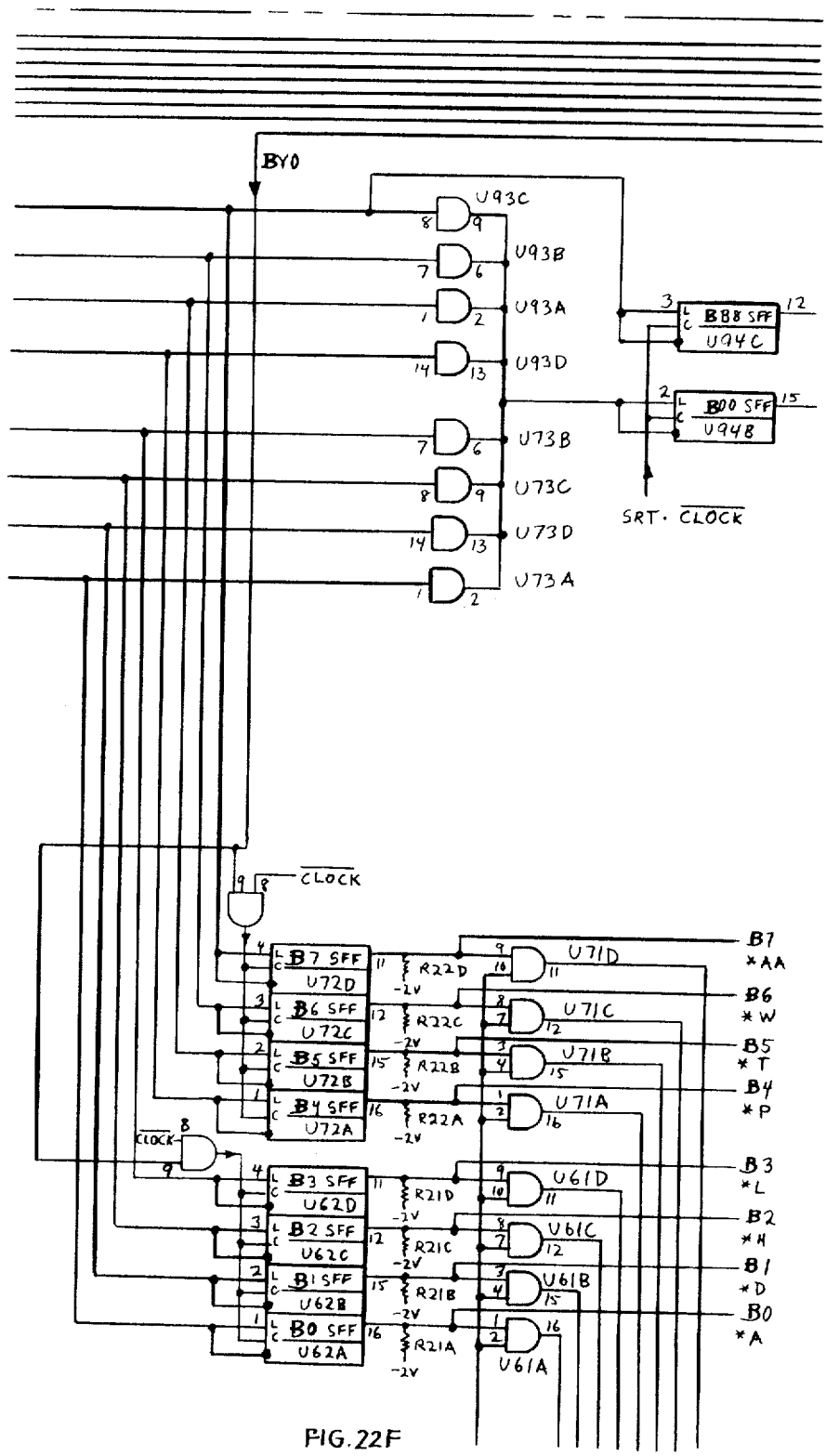

BY0
U93C
U93B
U93A
U93D
U73B
U73C
U73D
U73A
B88 SFF
U94C
B00 SFF
U94B
SRT · CLOCK
CLOCK
B7 SFF
U72D
B6 SFF
U72C
B5 SFF
U72B
B4 SFF
U72A
R22D
R22C
R22B
R22A
U71D
U71C
U71B
U71A
B7
*AA
B6
*W
B5
*T
B4
*P
B3 SFF
U62D
B2 SFF
U62C
B1 SFF
U62B
B0 SFF
U62A
R21D
R21C
R21B
R21A
U61D
U61C
U61B
U61A
B3
*L
B2
*H
B1
*D
B0
*A
-2V

FIG. 22F

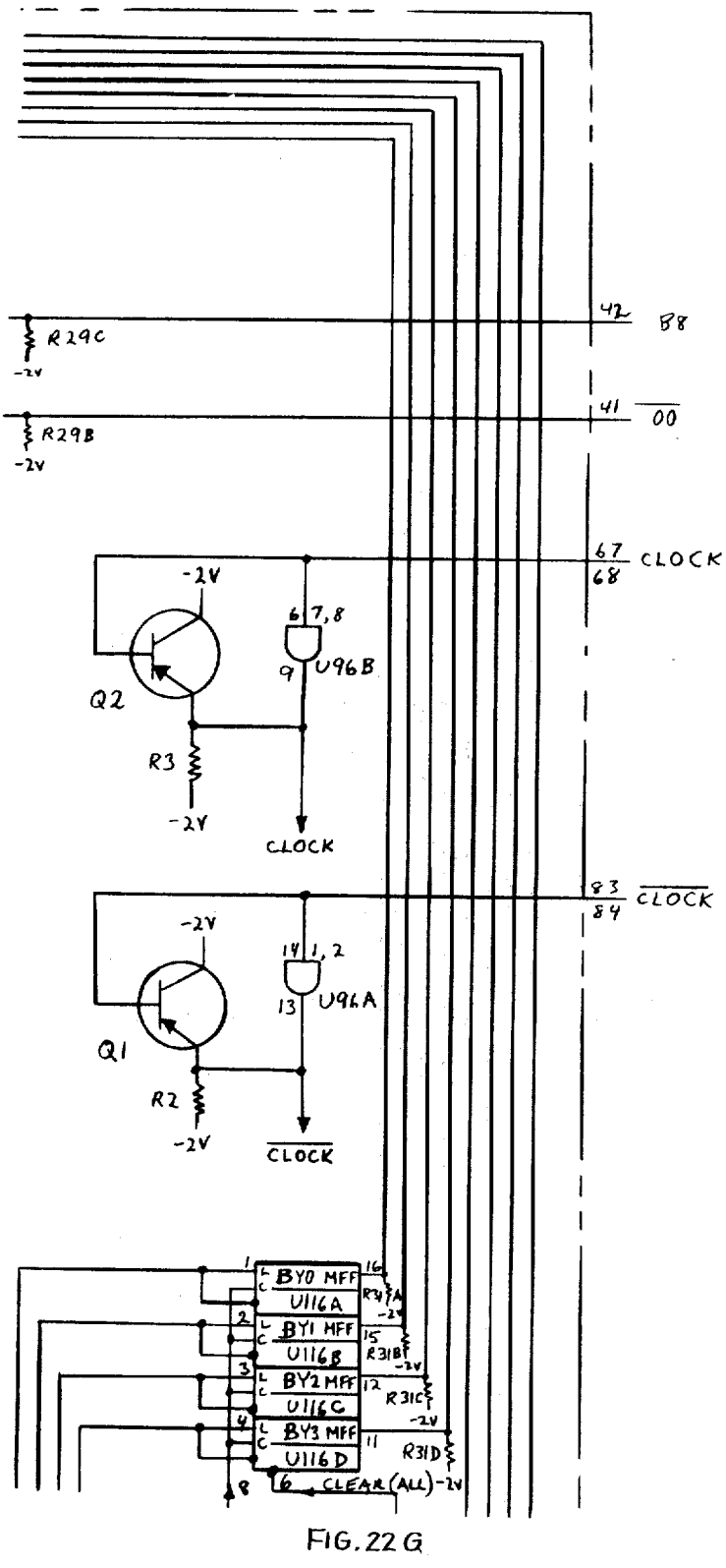

R29C
-2V
R29B
-2V
42
B8
41
00
67
68
CLOCK
-2V
6
7,8
9
U96B
Q2
R3
-2V
CLOCK
83
84
CLOCK
-2V
14
1, 2
13
U96A
Q1
R2
-2V
CLOCK
BY0 MFF
U116A
BY1 MFF
U116B
BY2 MFF
U116C
BY3 MFF
U116D
R31A
R31B
R31C
R31D
-2V
CLEAR (ALL)
-2V

FIG. 22G

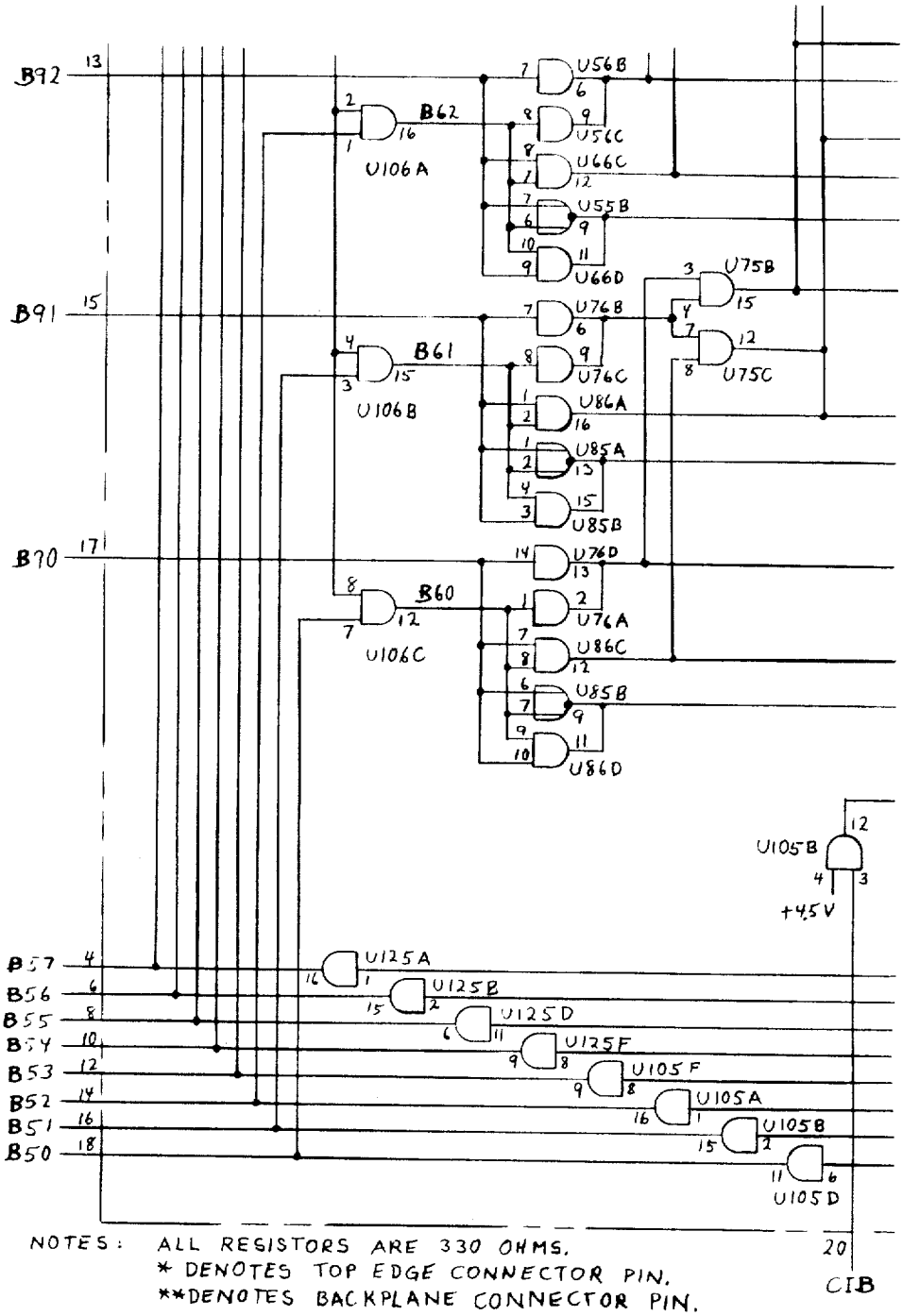

B92
B91
B90
B57
B56
B55
B54
B53
B52
B51
B50
B62
B61
B60
U106A
U106B
U106C
U56B
U56C
U66C
U55B
U66D
U75B
U75C
U76B
U76C
U86A
U85A
U85B
U76D
U76A
U86C
U85B
U86D
U125A
U125B
U125D
U125F
U105F
U105A
U105B
U105D
+4.5V
CIB
NOTES: ALL RESISTORS ARE 330 OHMS.
* DENOTES TOP EDGE CONNECTOR PIN.
**DENOTES BACKPLANE CONNECTOR PIN.

FIG. 22H

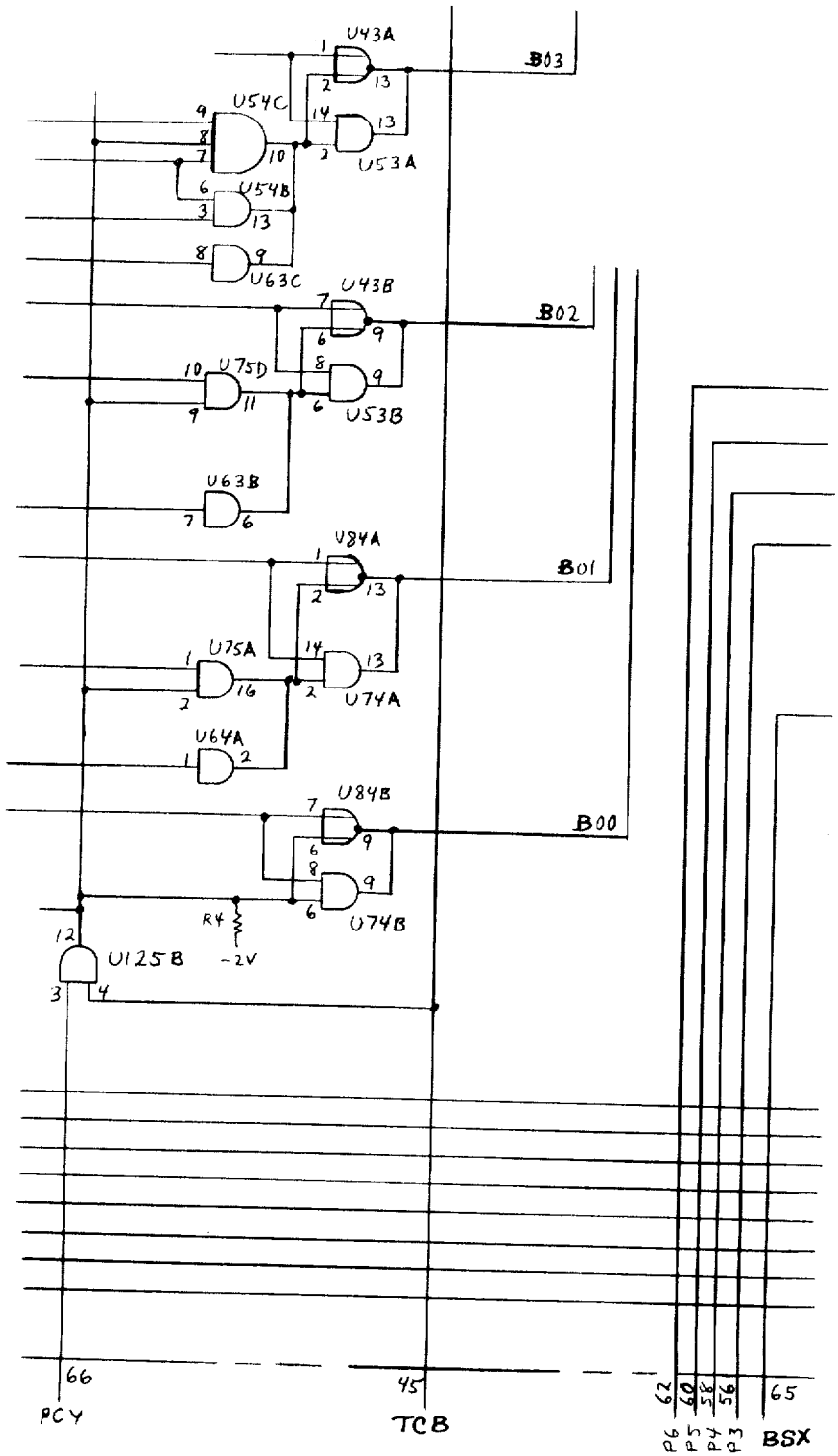

U43A
B03
U54C
U53A
U54B
U63C
U43B
B02
U75D
U53B
U63B
U84A
B01
U75A
U74A
U64A
U84B
B00
R4
-2V
U125B
U74B
66
PCY
45
TCB
62
60
58
56
65
P6
P5
P4
P3
BSX

FIG. 22I

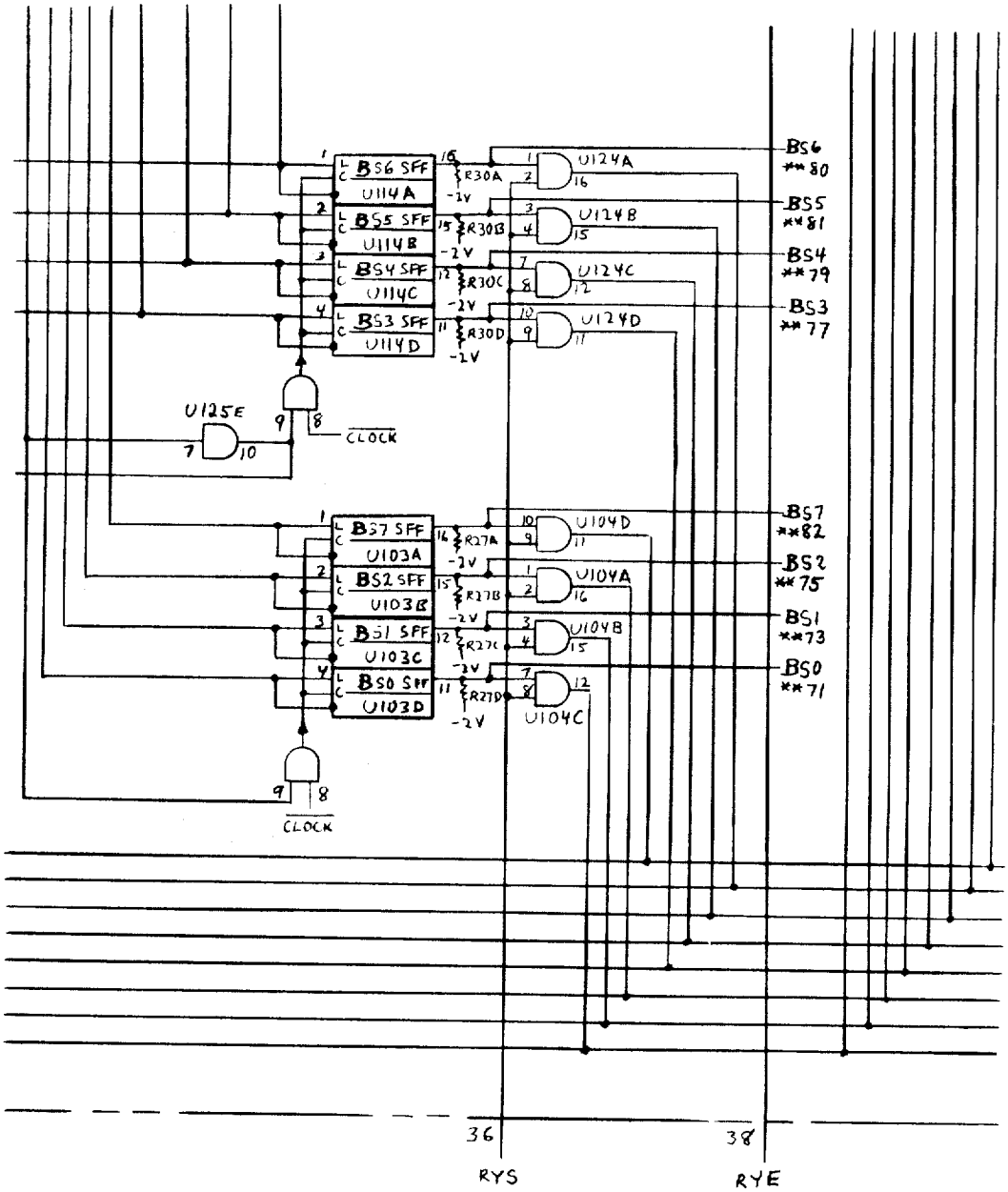

BS6 SFF
U114A
BS5 SFF
U114B
BS4 SFF
U114C
BS3 SFF
U114D
R30A
R30B
R30C
R30D
-2V
U124A
U124B
U124C
U124D
BS6
**80
BS5
**81
BS4
**79
BS3
**77
U125E
CLOCK
BS7 SFF
U103A
BS2 SFF
U103B
BS1 SFF
U103C
BS0 SFF
U103D
R27A
R27B
R27C
R27D
U104D
U104A
U104B
U104C
BS7
**82
BS2
**75
BS1
**73
BS0
**71
36
RYS
38
RYE

FIG. 22J

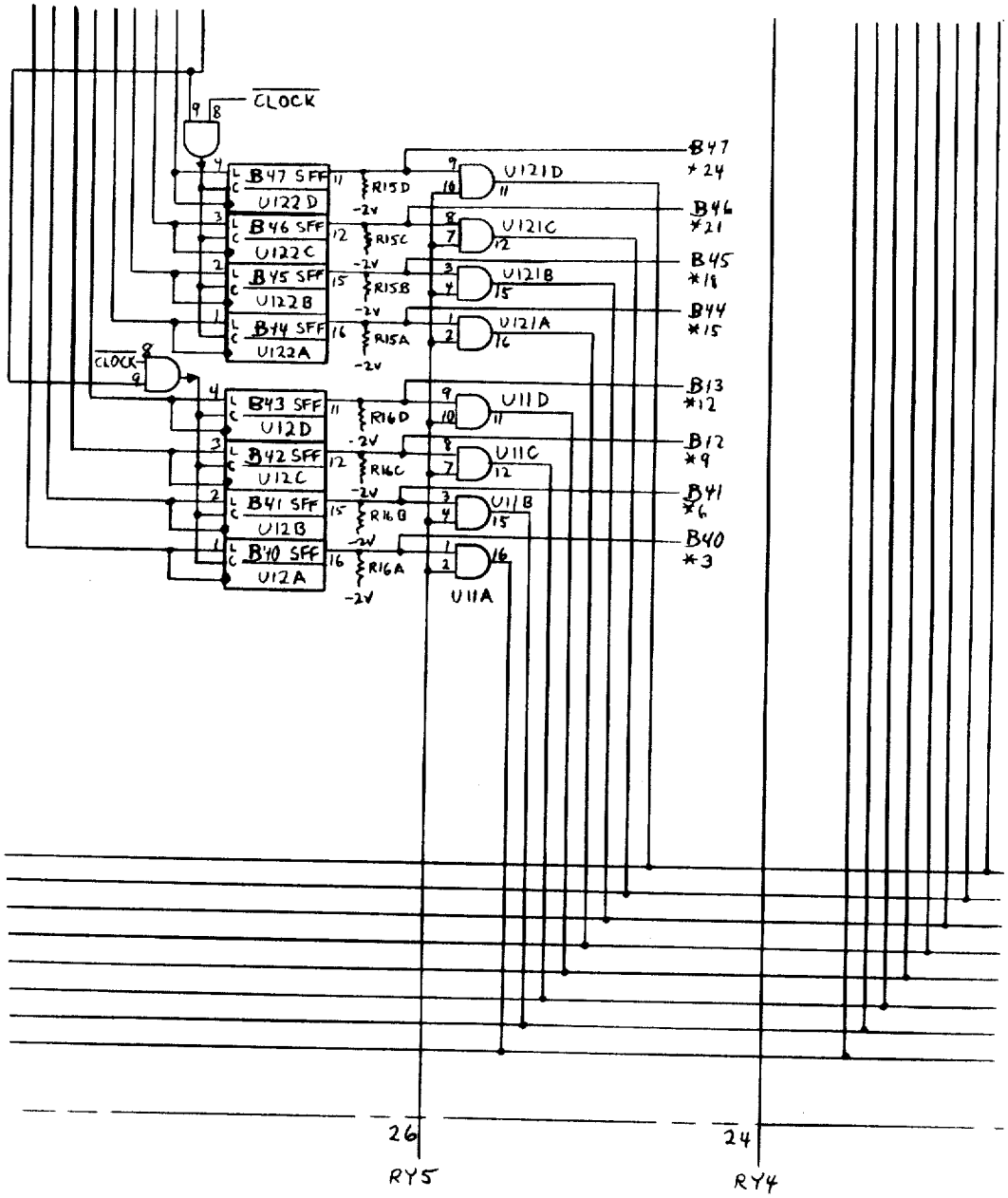

CLOCK
B47 SFF
U122D
B46 SFF
U122C
B45 SFF
U122B
B44 SFF
U122A
B43 SFF
U12D
B42 SFF
U12C
B41 SFF
U12B
B40 SFF
U12A
R15D
R15C
R15B
R15A
R16D
R16C
R16B
R16A
-2V
U121D
U121C
U121B
U121A
U11D
U11C
U11B
U11A
B47 *24
B46 *21
B45 *18
B44 *15
B13 *12
B12 *9
B41 *6
B40 *3
26
RY5
24
RY4

FIG. 22K

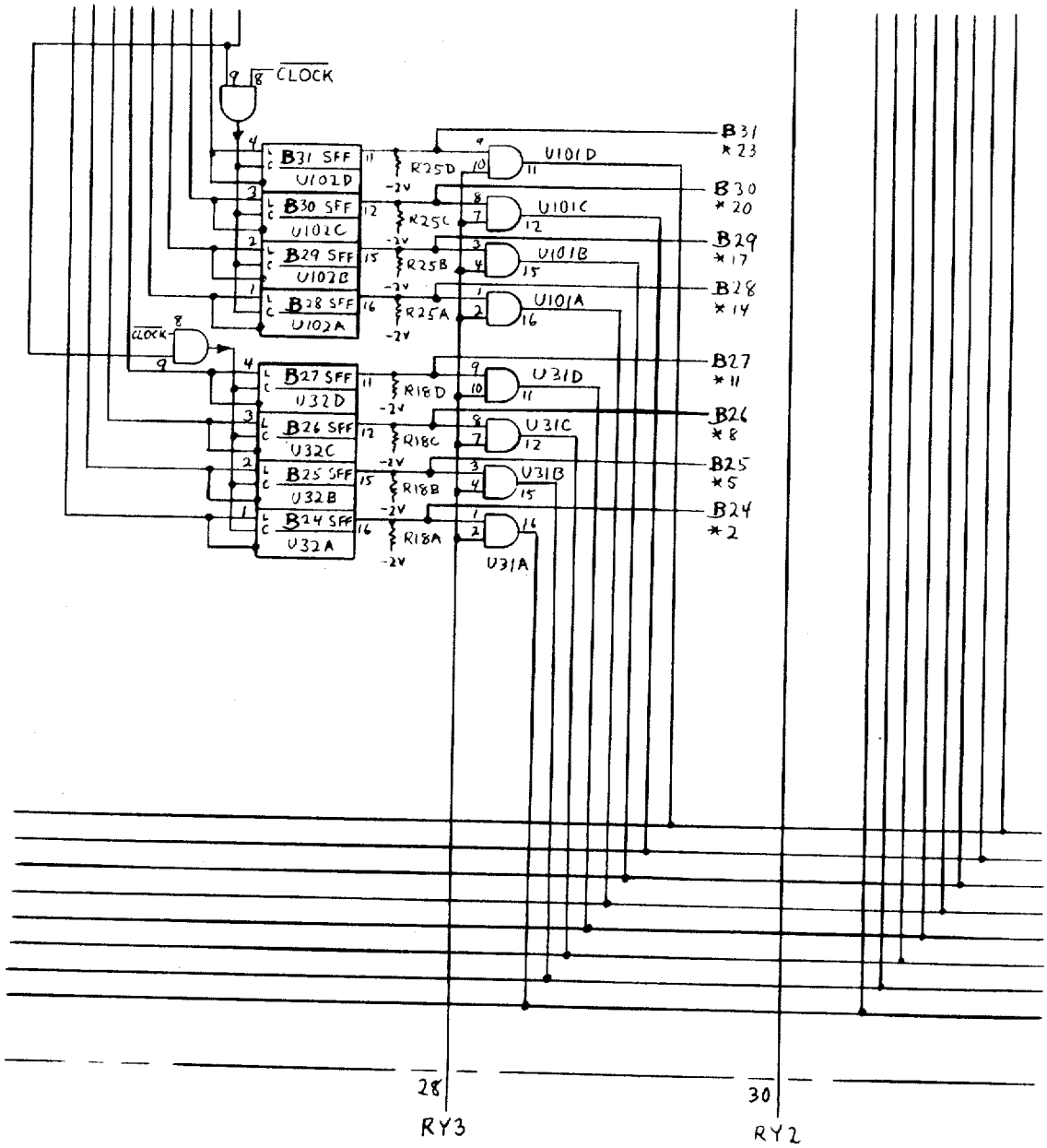

CLOCK
B31 SFF
U102D
B30 SFF
U102C
B29 SFF
U102B
B28 SFF
U102A
R25D
R25C
R25B
R25A
-2V
U101D
U101C
U101B
U101A
B31
*23
B30
*20
B29
*17
B28
*14
B27 SFF
U32D
B26 SFF
U32C
B25 SFF
U32B
B24 SFF
U32A
R18D
R18C
R18B
R18A
U31D
U31C
U31B
U31A
B27
*11
B26
*8
B25
*5
B24
*2
28
RY3
30
RY2

FIG. 22L

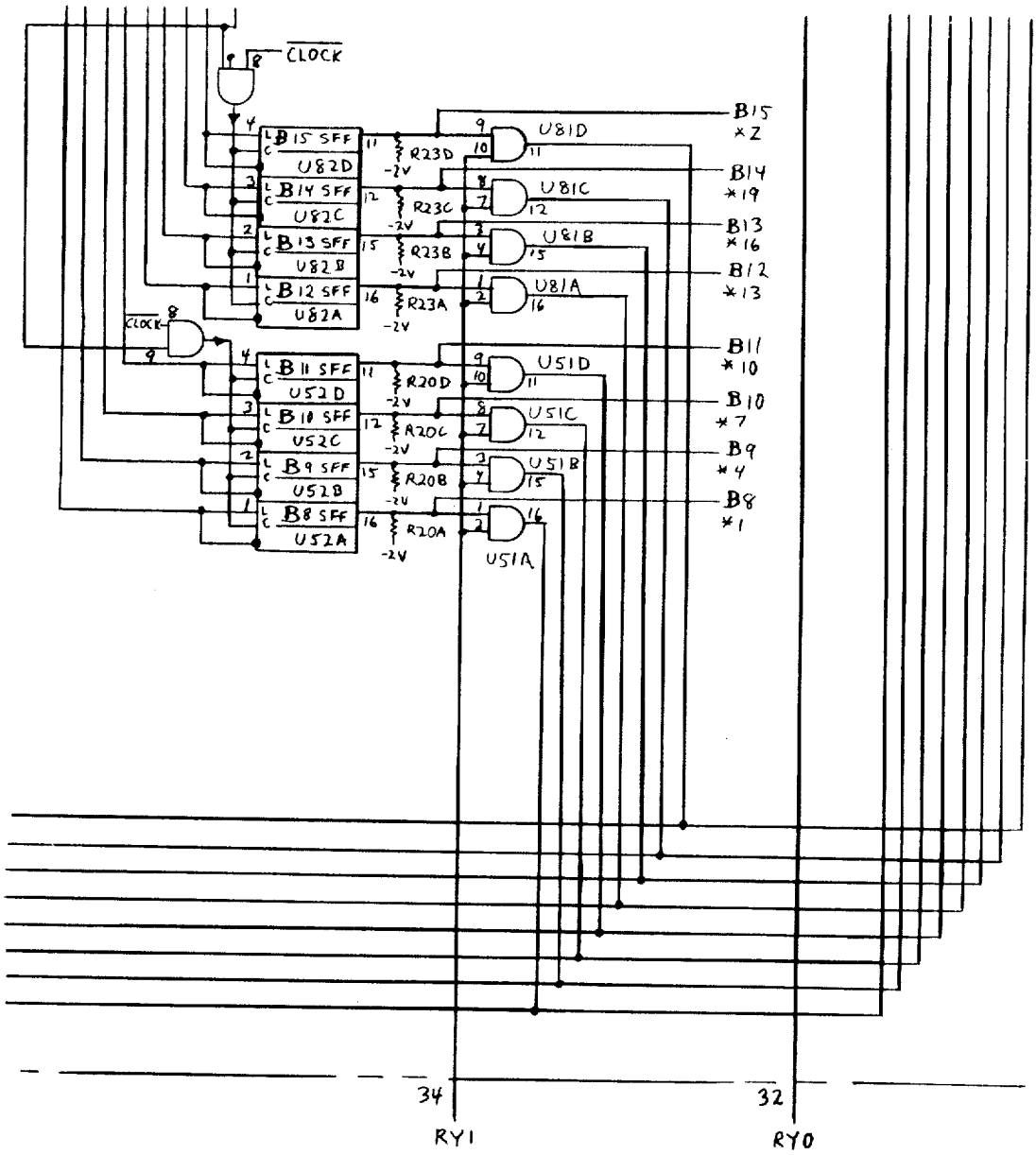

CLOCK
B15 SFF
U82D
B14 SFF
U82C
B13 SFF
U82B
B12 SFF
U82A
B11 SFF
U52D
B10 SFF
U52C
B9 SFF
U52B
B8 SFF
U52A
R23D
R23C
R23B
R23A
R20D
R20C
R20B
R20A
-2V
U81D
U81C
U81B
U81A
U51D
U51C
U51B
U51A
B15 *2
B14 *19
B13 *16
B12 *13
B11 *10
B10 *7
B9 *4
B8 *1
34
32
RY1
RY0

FIG. 22M

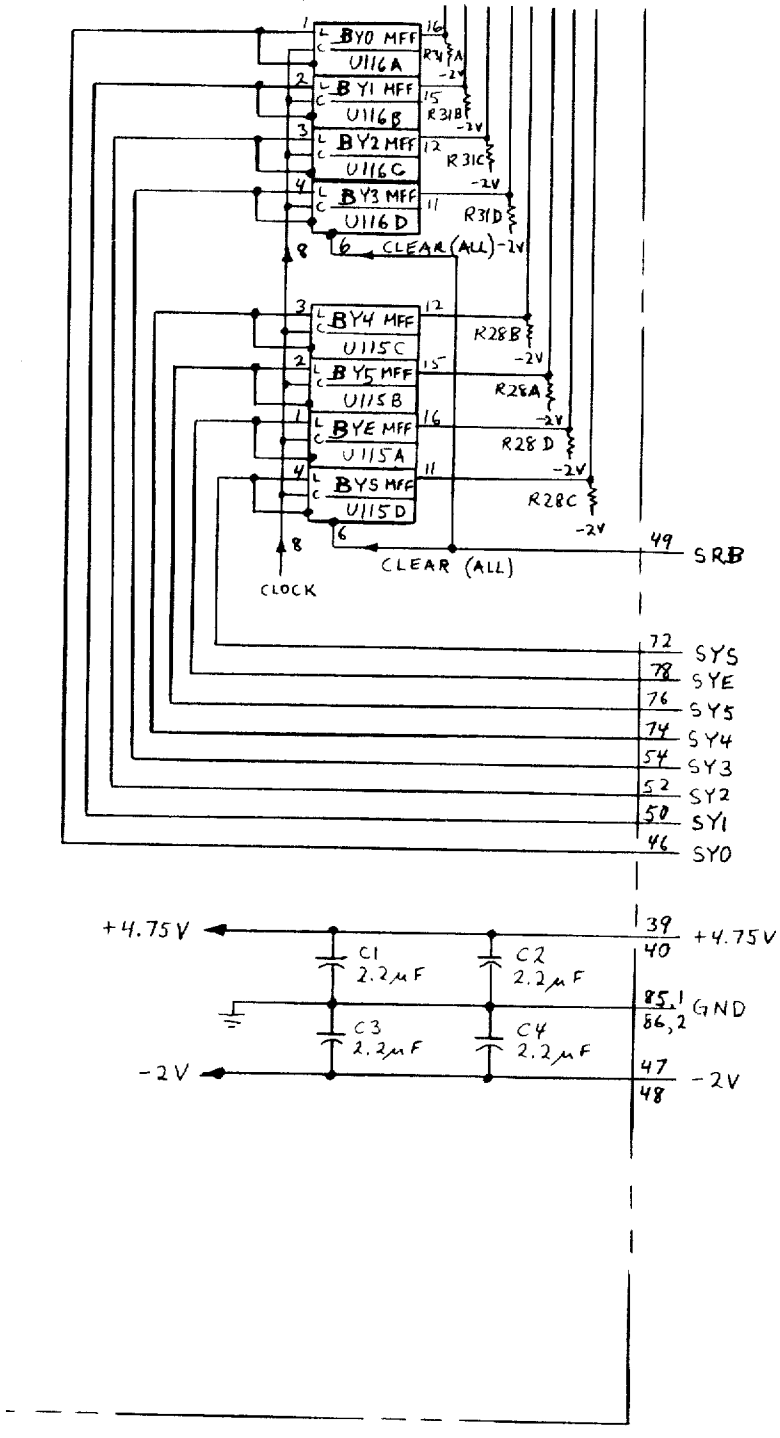

BY0 MFF
U116A
BY1 MFF
U116B
BY2 MFF
U116C
BY3 MFF
U116D
CLEAR (ALL) -2V
BY4 MFF
U115C
BY5 MFF
U115B
BYE MFF
U115A
BYS MFF
U115D
R31A
R31B
R31C
R31D
R28B
R28A
R28D
R28C
-2V
CLOCK
CLEAR (ALL)
49 SRB
72 SYS
78 SYE
76 SY5
74 SY4
54 SY3
52 SY2
50 SY1
46 SY0
+4.75V
C1 2.2μF
C2 2.2μF
C3 2.2μF
C4 2.2μF
39 +4.75V
40
85,1 GND
86,2
47 -2V
48

FIG. 22N

| | | | | | | |
|---|---|---|---|---|---|---|
| FIG 22A | FIG. 22B | FIG. 22C | FIG. 22D | FIG. 22E | FIG. 22F | FIG 22G |
| FIG. 22H | FIG 22I | FIG. 22J | FIG. 22K | FIG. 22L | FIG. 22M | FIG. 22N |

FIG. 23

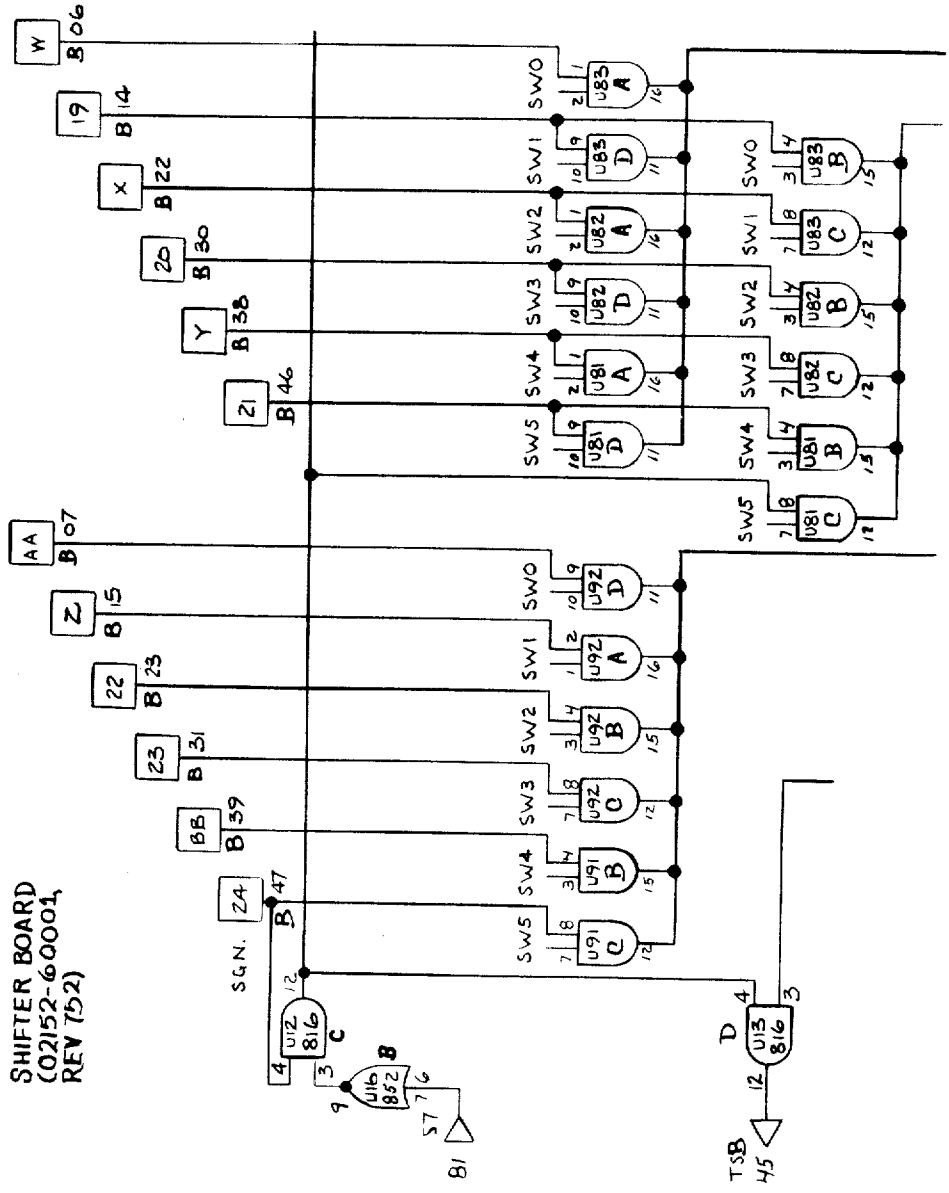

SHIFTER BOARD
(02152-60001,
REV (52)
W B 06
19 B 14
X B 22
20 B 30
Y B 38
21 B 46
AA B 07
Z B 15
22 B 23
23 B 31
BB B 39
Z4 B 47
SGN.
SW0
SW1
SW2
SW3
SW4
SW5
U83 A
U83 D
U82 A
U82 D
U81 A
U81 D
U92 D
U92 A
U92 B
U92 C
U91 B
U91 C
U83 B
U83 C
U82 B
U82 C
U81 B
U81 C
U112 816 C
U116 852 B
S7
81
D
U113 816
TSB
45

FIG. 24A

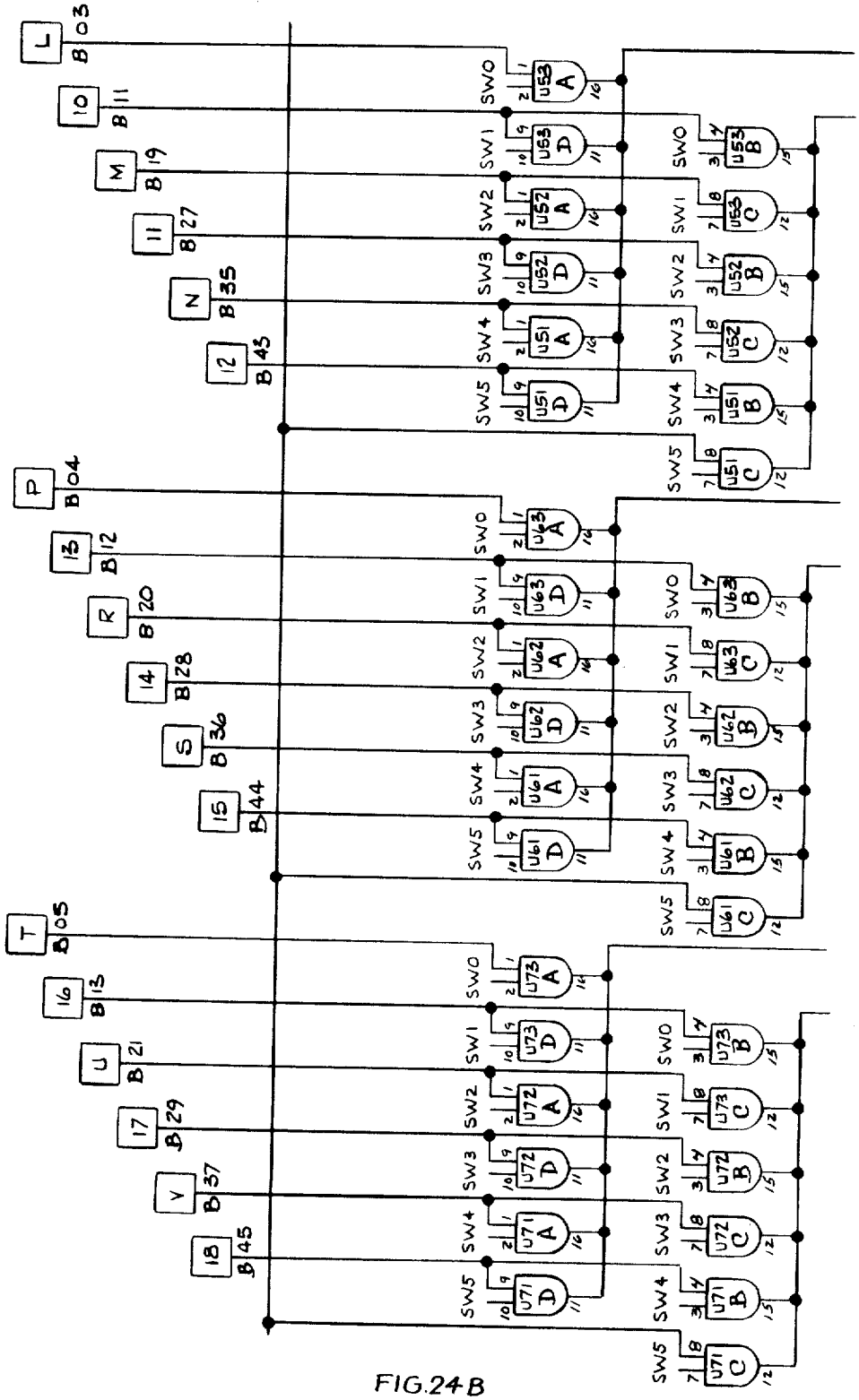

L B03
10 B11
M B19
11 B27
N B35
12 B45
P B04
13 B12
R B20
14 B28
S B36
15 B44
T B05
16 B13
U B21
17 B29
V B37
18 B45
SW0
SW1
SW2
SW3
SW4
SW5
U51 A
U51 B
U51 C
U51 D
U52 A
U52 B
U52 C
U52 D
U53 A
U53 B
U53 C
U53 D
U61 A
U61 B
U61 C
U61 D
U62 A
U62 B
U62 C
U62 D
U63 A
U63 B
U63 C
U63 D
U71 A
U71 B
U71 C
U71 D
U72 A
U72 B
U72 C
U72 D
U73 A
U73 B
U73 C
U73 D

FIG.24B

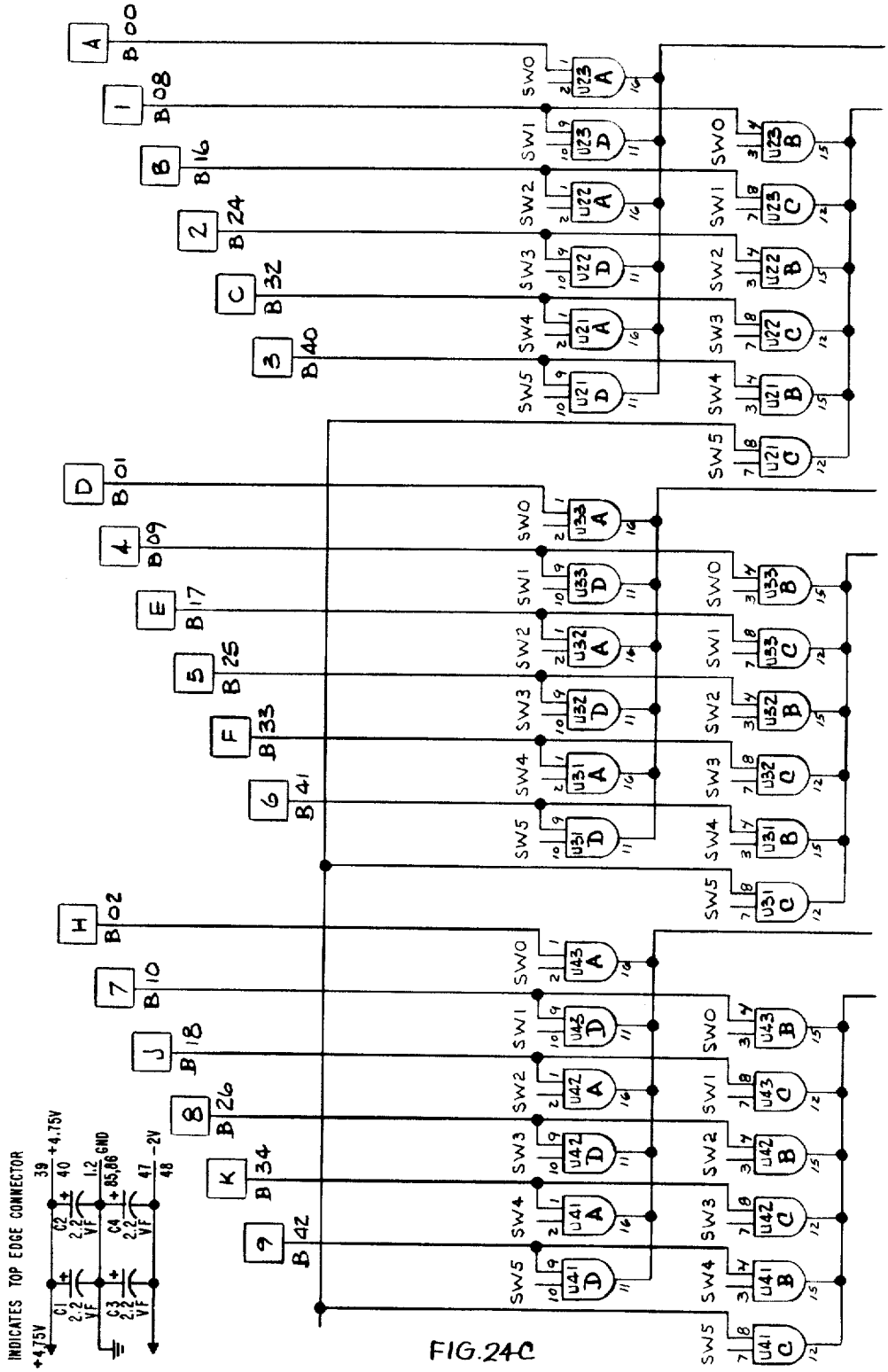
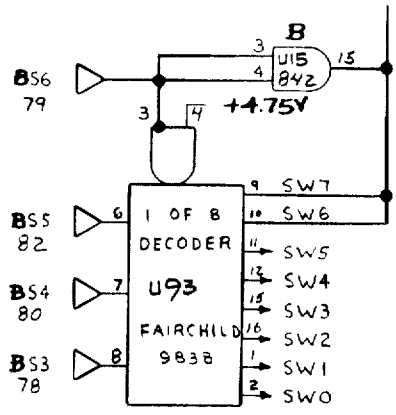

INDICATES TOP EDGE CONNECTOR
+4.75V
C1 2.2 VF
C2 2.2 VF
C3 2.2 VF
C4 2.2 VF
39 +4.75V
40
1,2 +5,6 GND
47 -2V
48
A B00
1 B08
B B16
2 B24
C B32
3 B40
D B01
4 B09
E B17
5 B25
F B33
6 B41
H B02
7 B10
J B18
8 B26
K B34
9 B42
SW0
SW1
SW2
SW3
SW4
SW5
U23 A
U23 D
U23 B
U22 A
U22 D
U22 B
U21 A
U21 D
U21 B
U21 C
U33 A
U33 D
U33 B
U32 A
U32 D
U32 B
U31 A
U31 D
U31 B
U31 C
U43 A
U43 D
U43 B
U42 A
U42 D
U42 B
U41 A
U41 D
U41 B
U41 C

FIG. 24C


B
U15
842
+4.75V
BS6
79
BS5
82
BS4
80
BS3
78
1 OF 8
DECODER
U93
FAIRCHILD
9838
SW7
SW6
SW5
SW4
SW3
SW2
SW1
SW0

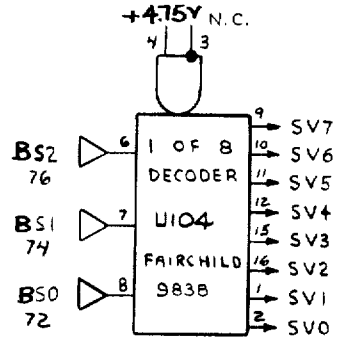

+4.75V
N.C.
BS2
76
BS1
74
BS0
72
1 OF 8
DECODER
U104
FAIRCHILD
9838
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0

FIG. 24D

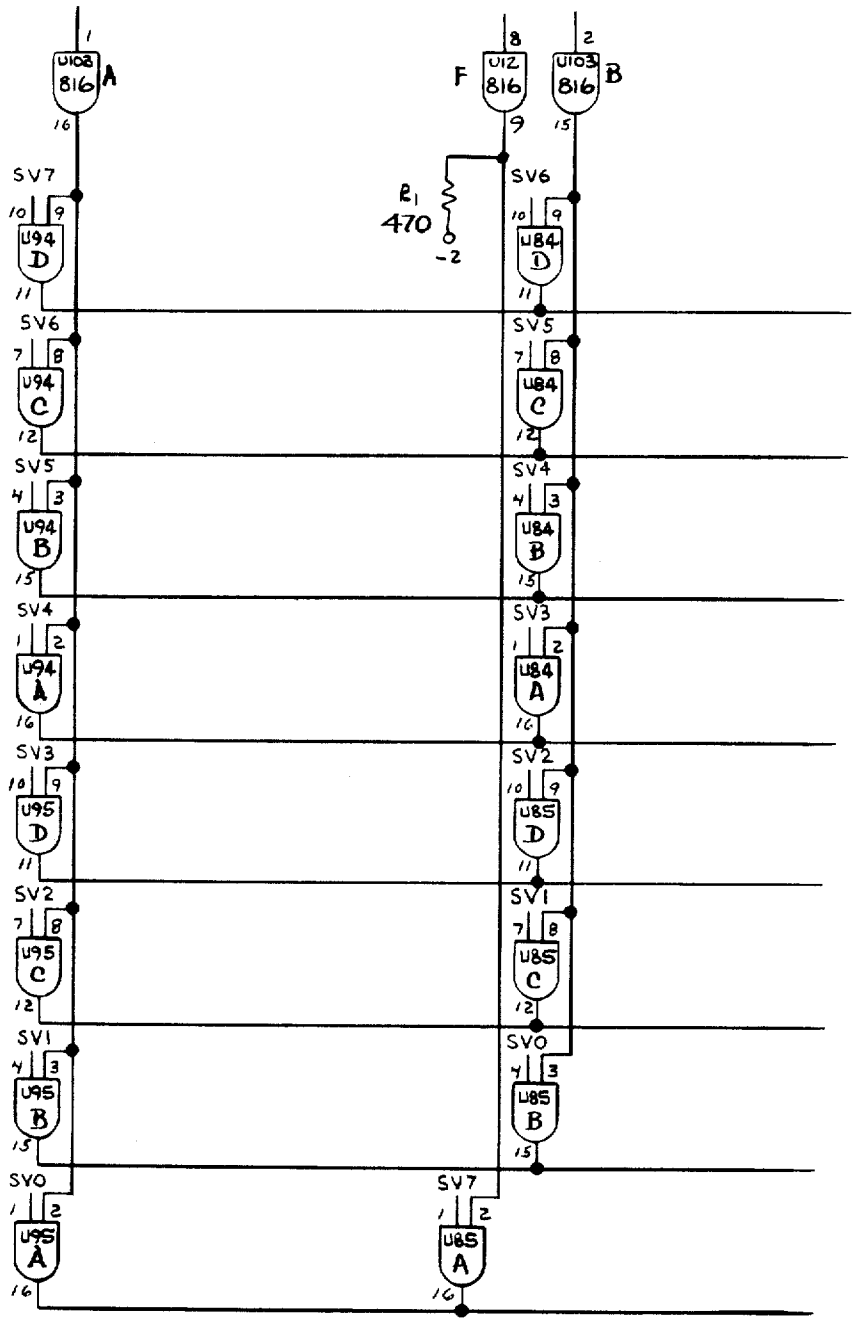

U108 816 A
U12 816 F
U103 816 B
R1 470 -2
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0
U94 D
U94 C
U94 B
U94 A
U95 D
U95 C
U95 B
U95 A
U84 D
U84 C
U84 B
U84 A
U85 D
U85 C
U85 B
U85 A

FIG. 24E

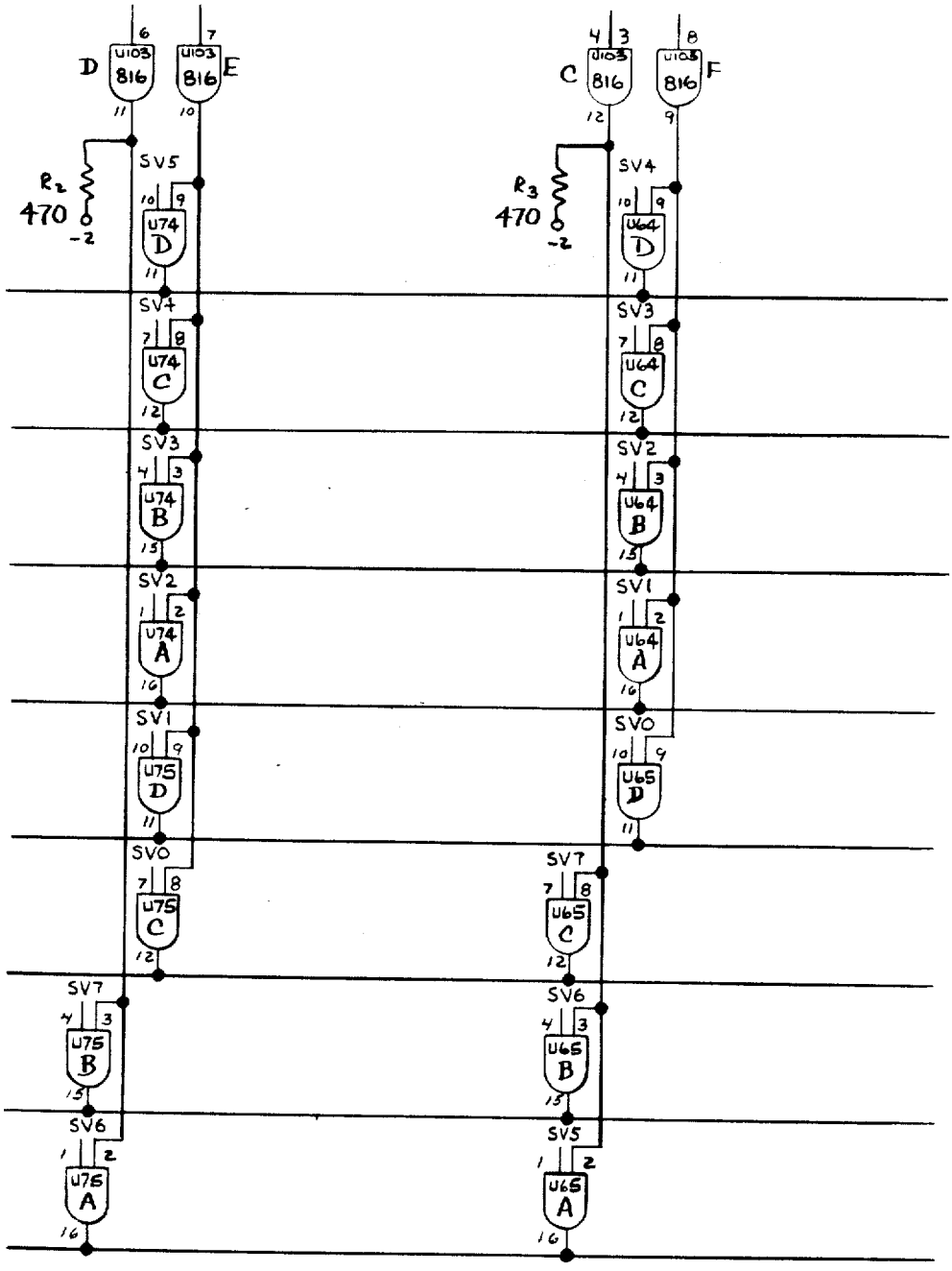

D
E
C
F
U103
816
R2
470
-2
R3
SV5
SV4
SV3
SV2
SV1
SV0
SV7
SV6
U74
U64
U75
U65
A
B
C
D

FIG. 24F

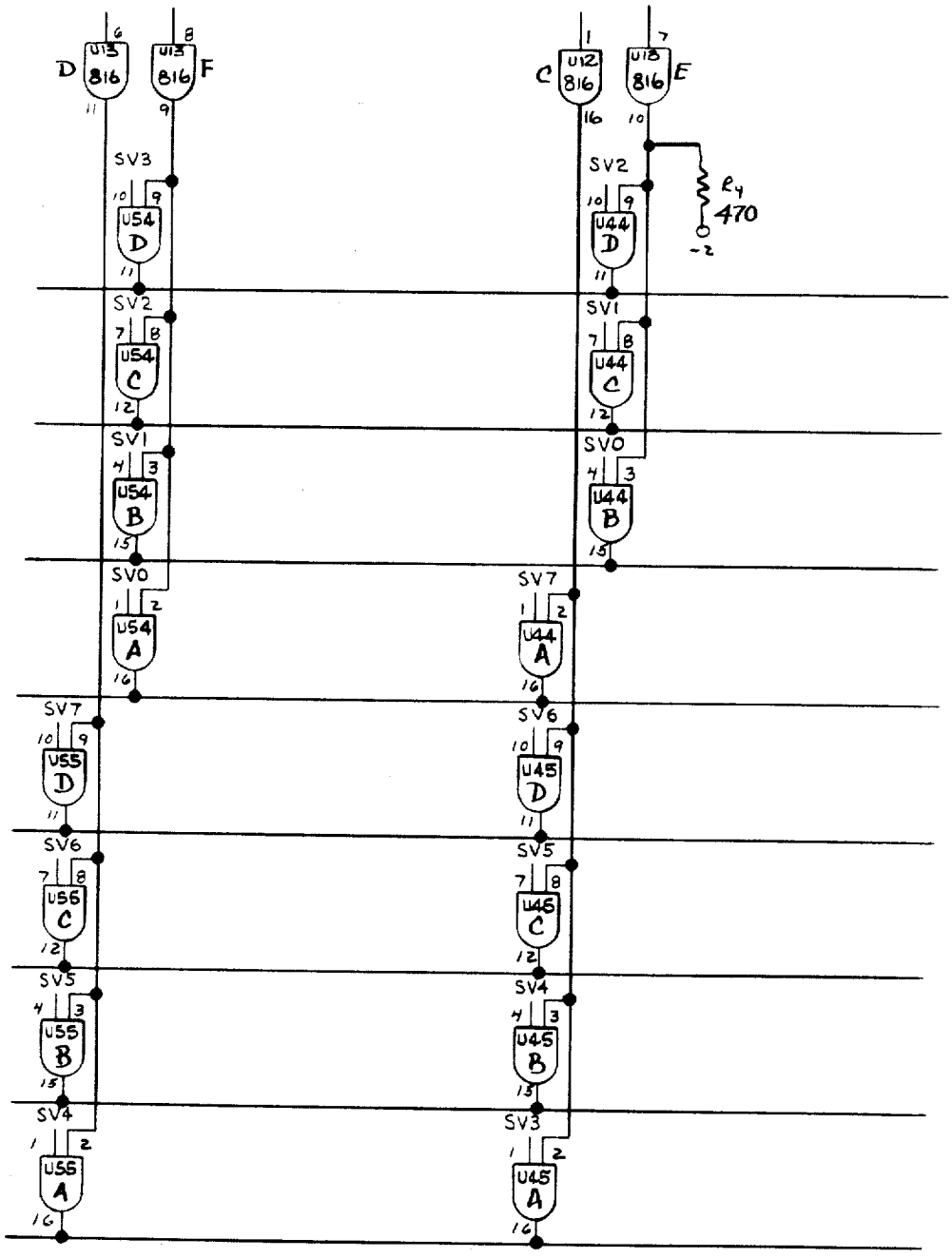

D
U13 816
F
U13 816
C
U12 816
E
U18 816
SV3
U54 D
SV2
U54 C
SV1
U54 B
SV0
U54 A
SV7
U55 D
SV6
U56 C
SV5
U55 B
SV4
U55 A
SV2
U44 D
SV1
U44 C
SV0
U44 B
SV7
U44 A
SV6
U45 D
SV5
U46 C
SV4
U45 B
SV3
U45 A
R4
470
-2

FIG. 24G

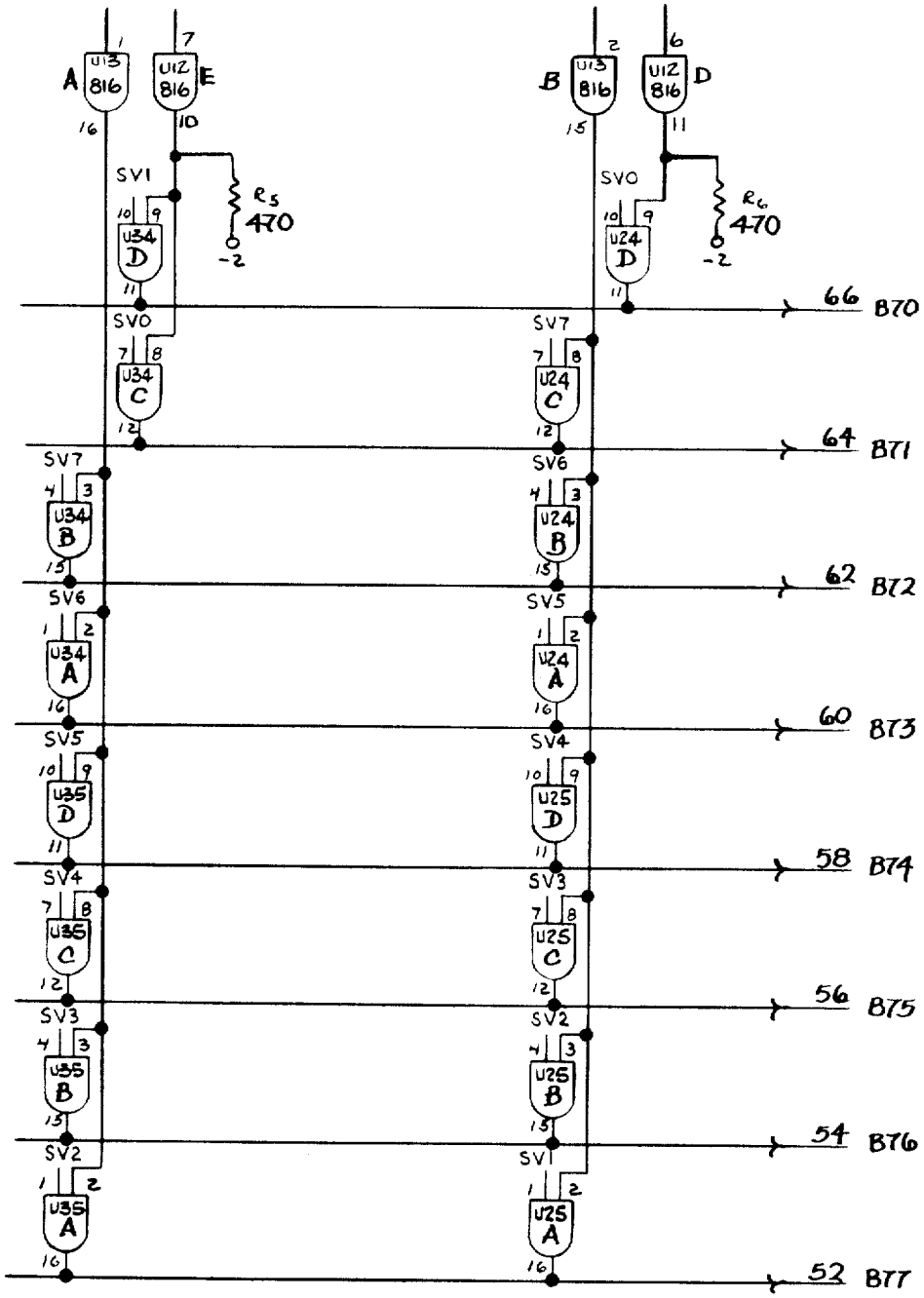

A
B
E
D
U13 816
U12 816
SV1
SV0
SV7
SV6
SV5
SV4
SV3
SV2
SV
R5 470
R6 470
-2
U34 D
U34 C
U34 B
U34 A
U35 D
U35 C
U35 B
U35 A
U24 D
U24 C
U24 B
U24 A
U25 D
U25 C
U25 B
U25 A
66 BT0
64 BT1
62 BT2
60 BT3
58 BT4
56 BT5
54 BT6
52 BT7

FIG. 24H

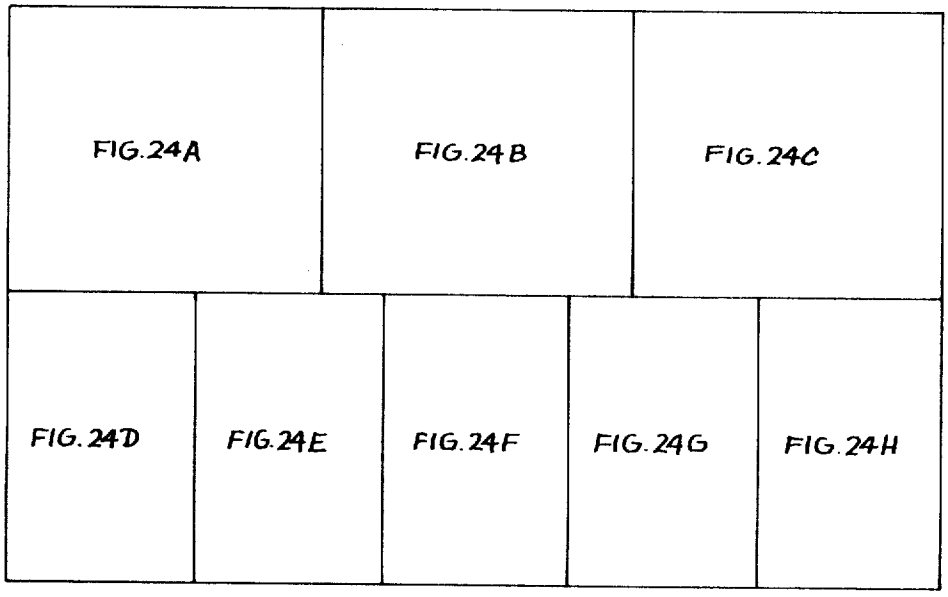



FIG. 25

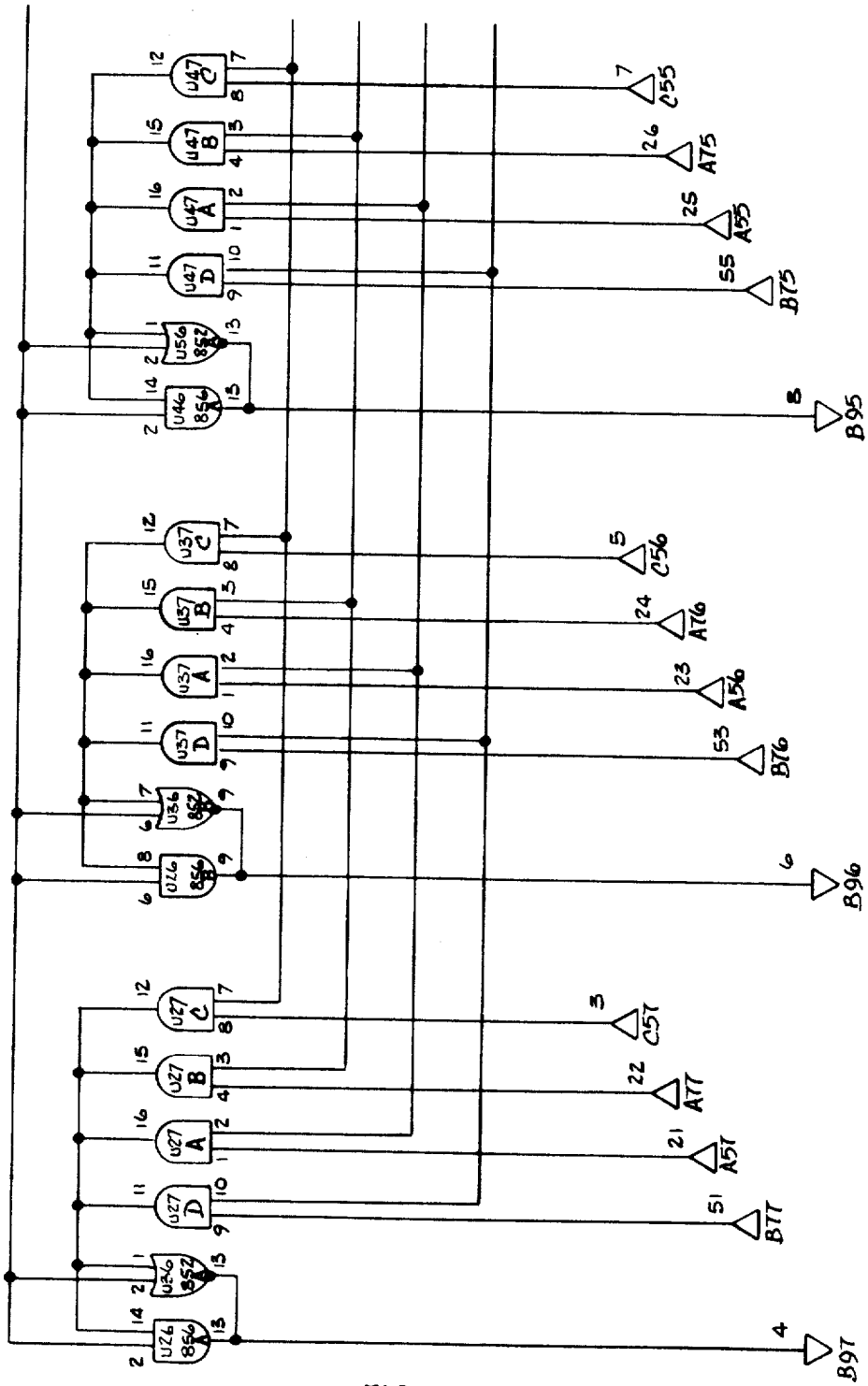

U47 C
U47 B
U47 A
U47 D
U56 852
U46 856
U37 C
U37 B
U37 A
U37 D
U36 852
U26 856
U27 C
U27 B
U27 A
U27 D
U340 852
U24 856
C55
A75
A55
B75
B95
C56
A76
A56
B76
B96
C57
A77
A57
B77
B97

FIG. 26A

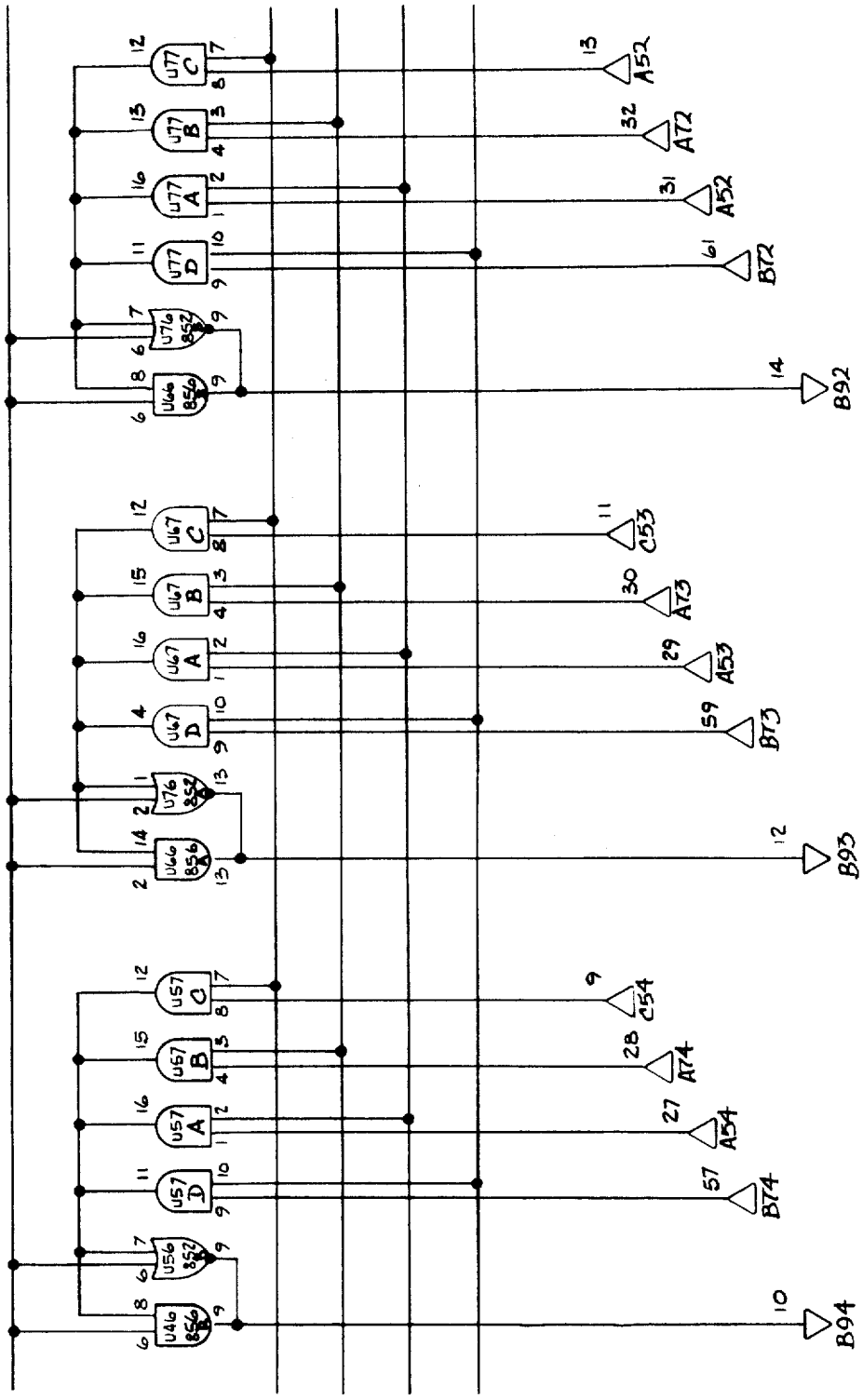

U77 C
U77 B
U77 A
U77 D
U76 852
U66 856
U67 C
U67 B
U67 A
U67 D
U76 852
U66 856
U57 C
U57 B
U57 A
U57 D
U56 852
U46 856
A52
A72
A52
B72
B92
C53
A73
A53
B73
B93
C54
A74
A54
B74
B94

FIG. 26B

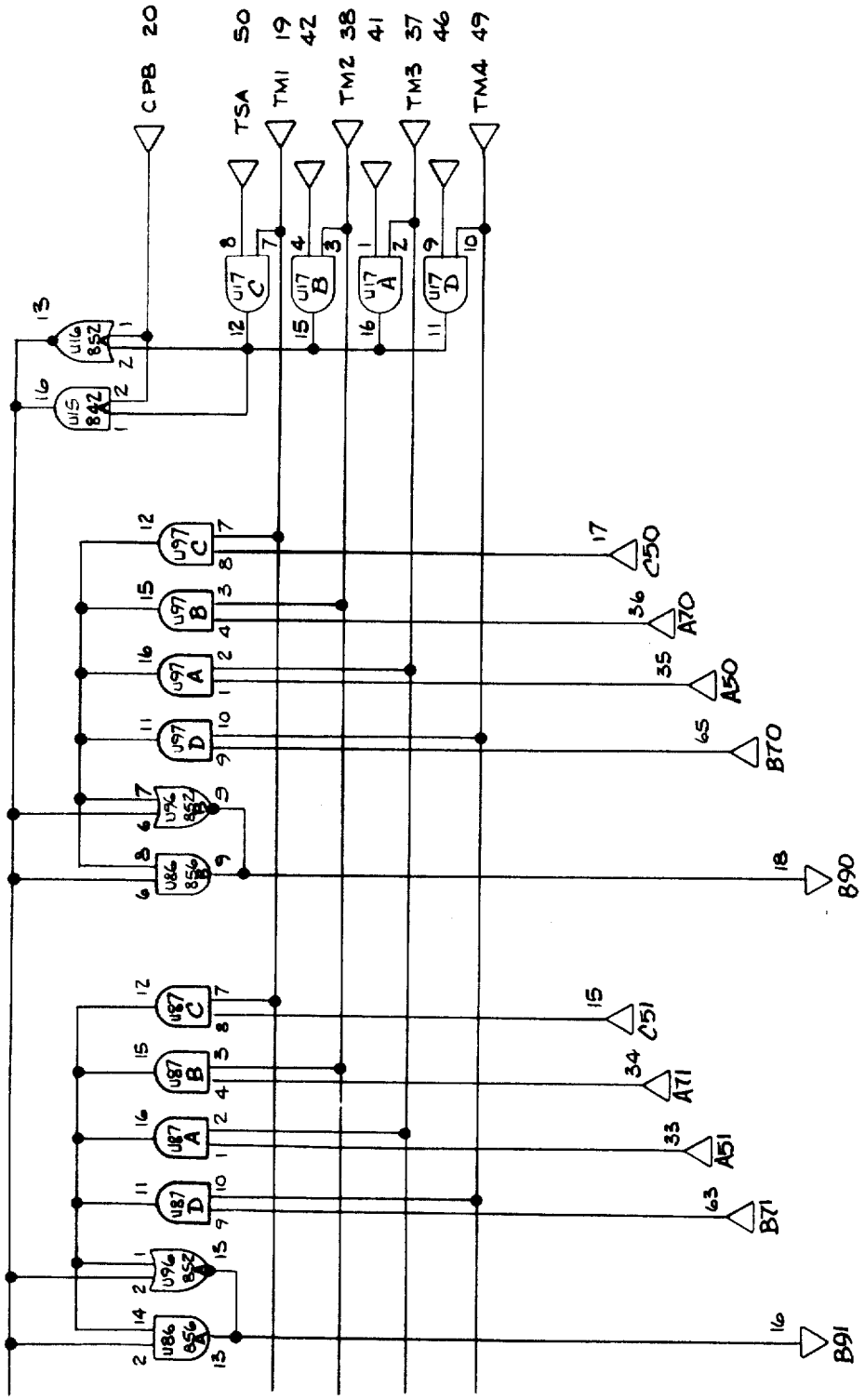

CPB 20
TSA 50
TM1 19 42
TM2 38 41
TM3 37 46
TM4 49
U17 C
U17 B
U17 A
U17 D
U16 852
U15 842
U97 C
U97 B
U97 A
U97 D
U96 852
U86 856
U87 C
U87 B
U87 A
U87 D
U96 852
U86 856
C50
A70
A50
B70
B90
C51
A71
A51
B71
B91

FIG. 26C

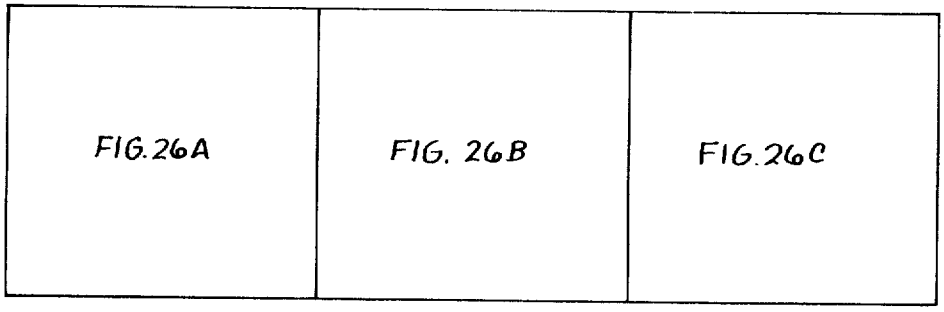



FIG. 27

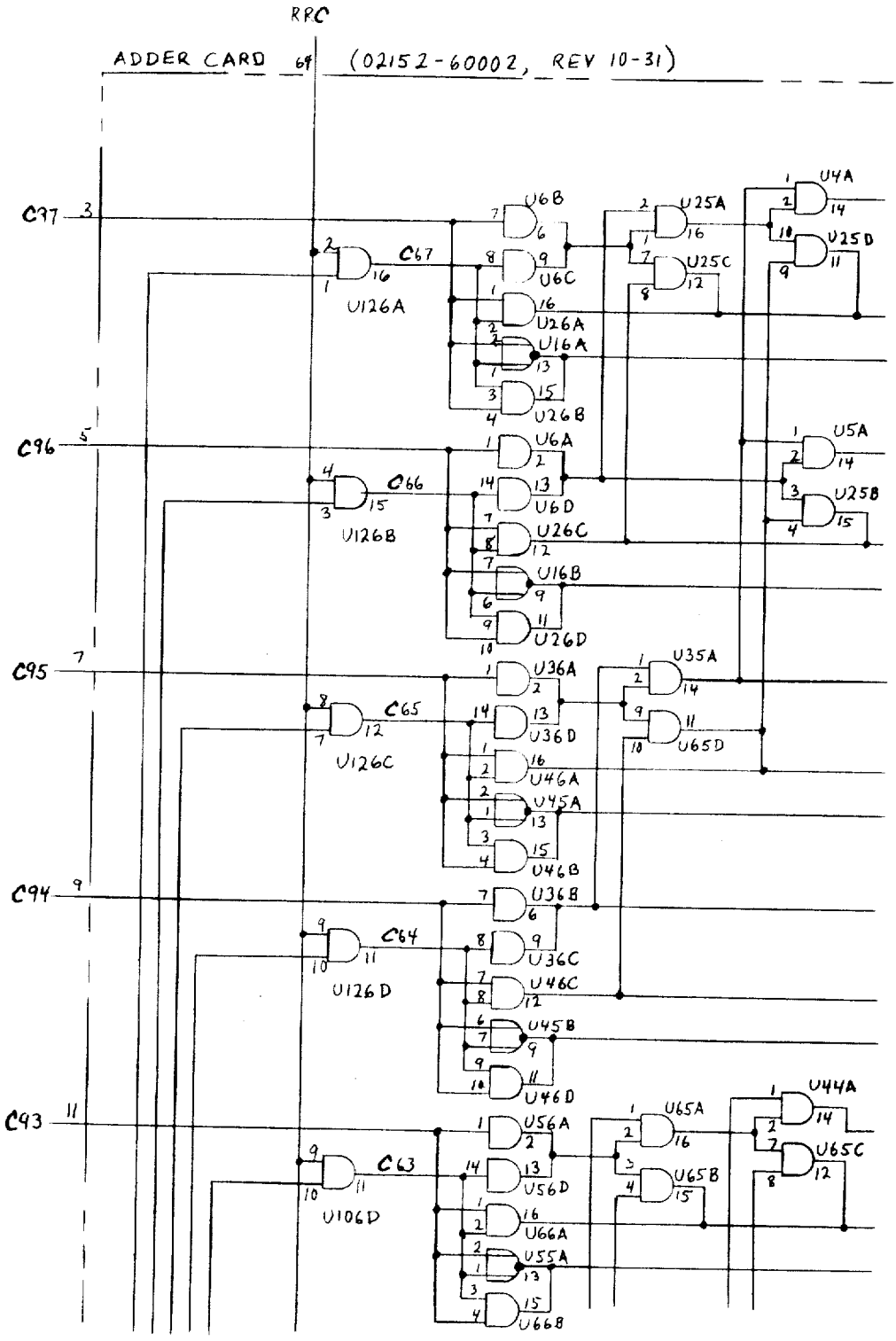

RRC
ADDER CARD 69 (02152-60002, REV 10-31)
C97
C96
C95
C94
C93
C67
C66
C65
C64
C63
U126A
U126B
U126C
U126D
U106D
U6B
U6C
U26A
U16A
U26B
U6A
U6D
U26C
U16B
U26D
U36A
U36D
U46A
U45A
U46B
U36B
U36C
U46C
U45B
U46D
U56A
U56D
U66A
U55A
U66B
U25A
U25C
U35A
U65D
U65A
U65B
U4A
U25D
U5A
U25B
U44A
U65C

FIG 28A

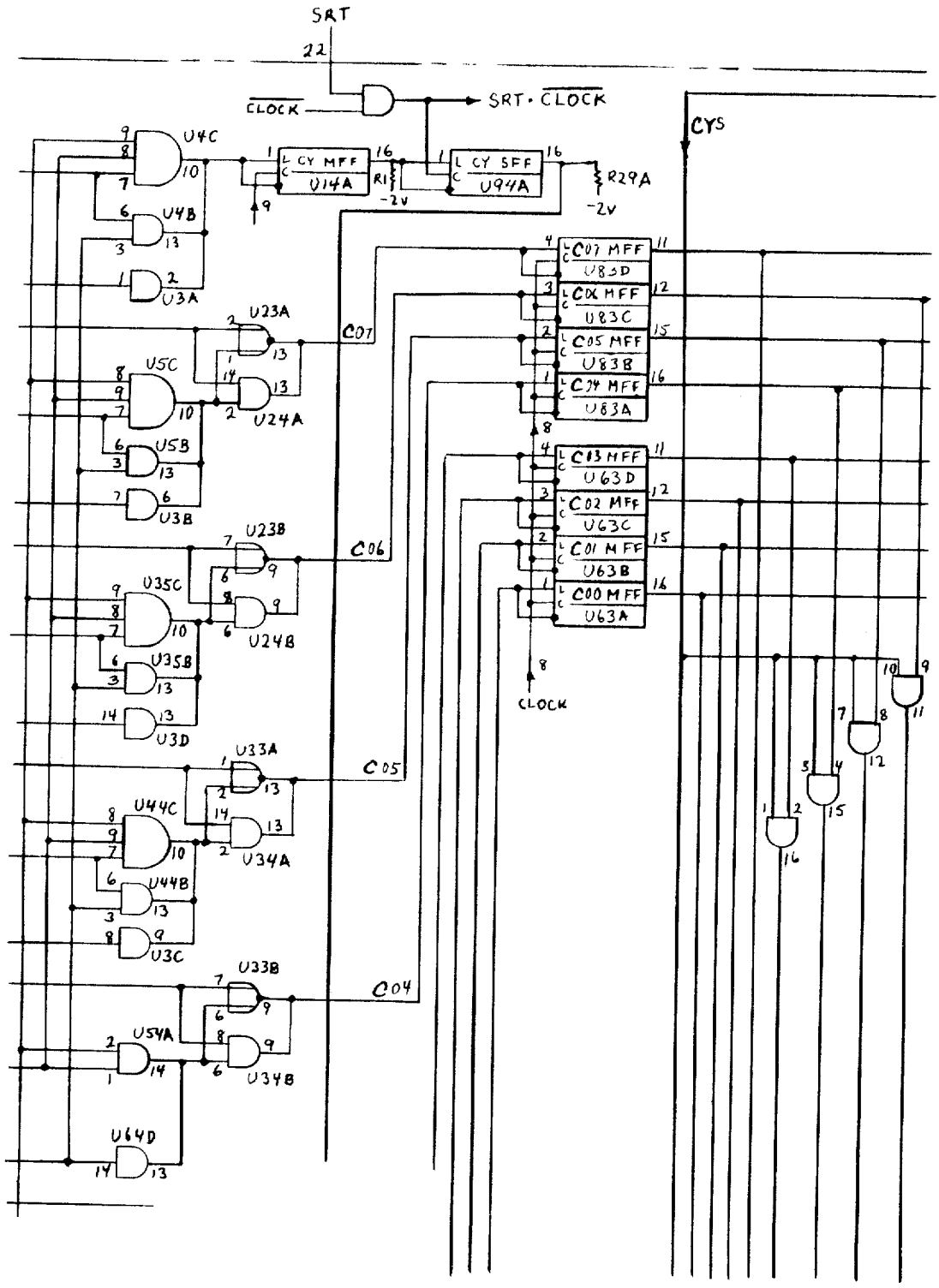

SRT
22
CLOCK
SRT · CLOCK
CYS
U4C
U4B
U3A
U23A
U5C
U24A
U5B
U3B
U23B
U35C
U24B
U35B
U3D
U33A
U44C
U34A
U44B
U3C
U33B
U54A
U34B
U64D
L CY MFF
U14A
L CY SFF
U94A
R1
-2V
R29A
-2V
C07
C06
C05
C04
C07 MFF
U83D
C06 MFF
U83C
C05 MFF
U83B
C04 MFF
U83A
C03 MFF
U63D
C02 MFF
U63C
C01 MFF
U63B
C00 MFF
U63A
CLOCK

FIG.28B

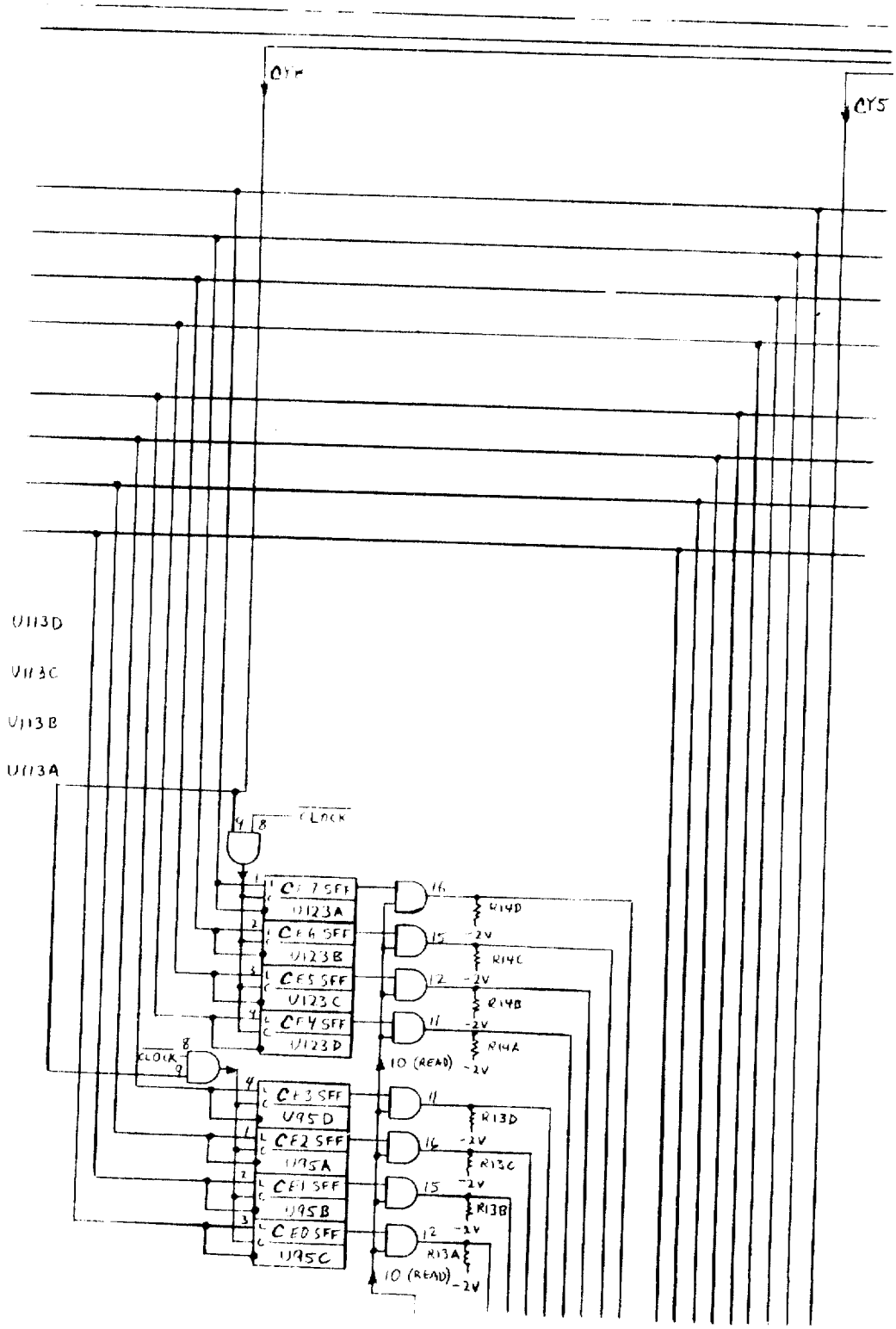

CY5
U113D
U113C
U113B
U113A
CLOCK
CF7 SFF
U123A
CF6 SFF
U123B
CF5 SFF
U123C
CF4 SFF
U123D
CLOCK
CF3 SFF
U95D
CF2 SFF
U95A
CF1 SFF
U95B
CF0 SFF
U95C
IO (READ)
IO (READ)
R14D
R14C
R14B
R14A
R13D
R13C
R13B
R13A
-2V

FIG. 28C

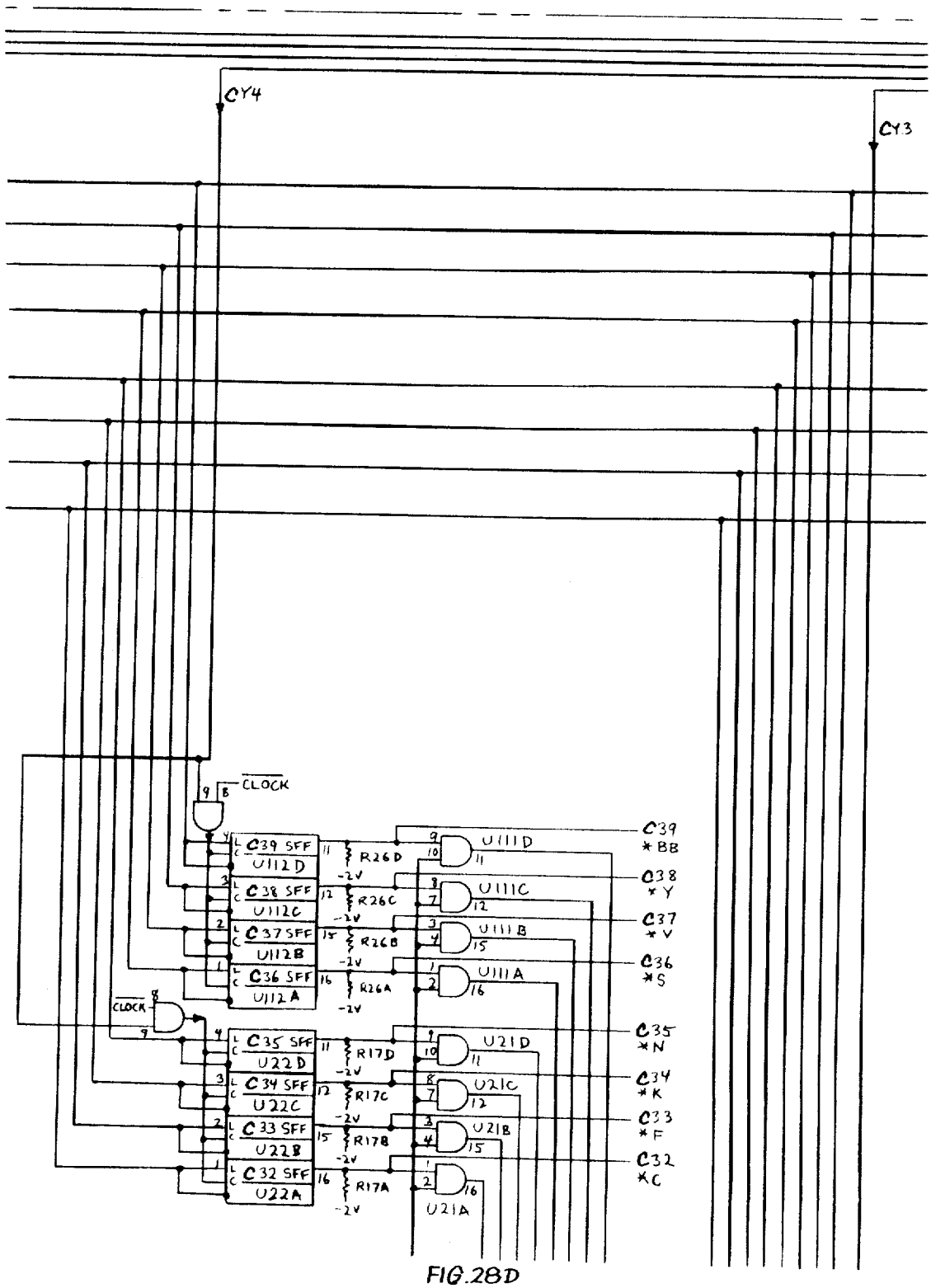

CY4
CY3
CLOCK
C39 SFF U112D
C38 SFF U112C
C37 SFF U112B
C36 SFF U112A
C35 SFF U22D
C34 SFF U22C
C33 SFF U22B
C32 SFF U22A
R26D
R26C
R26B
R26A
R17D
R17C
R17B
R17A
-2V
U111D
U111C
U111B
U111A
U21D
U21C
U21B
U21A
C39 * BB
C38 * Y
C37 * V
C36 * S
C35 * N
C34 * K
C33 * F
C32 * C

FIG. 28D

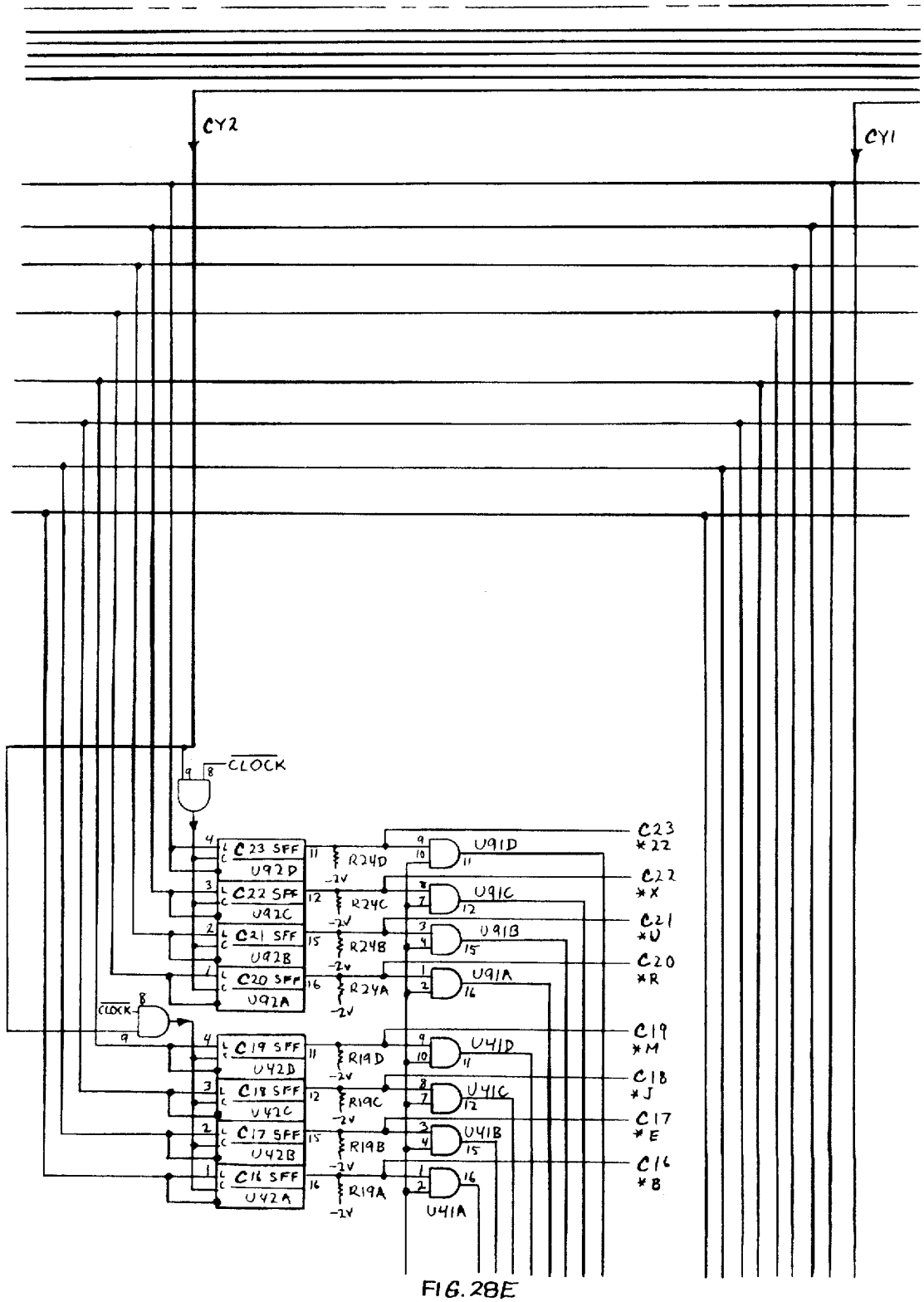

CY2
CY1
CLOCK
C23 SFF
U92D
C22 SFF
U92C
C21 SFF
U92B
C20 SFF
U92A
R24D
R24C
R24B
R24A
-2V
U91D
U91C
U91B
U91A
C23
*Z2
C22
*X
C21
*U
C20
*R
CLOCK
C19 SFF
U42D
C18 SFF
U42C
C17 SFF
U42B
C16 SFF
U42A
R19D
R19C
R19B
R19A
U41D
U41C
U41B
U41A
C19
*M
C18
*J
C17
*E
C16
*B

FIG. 28E

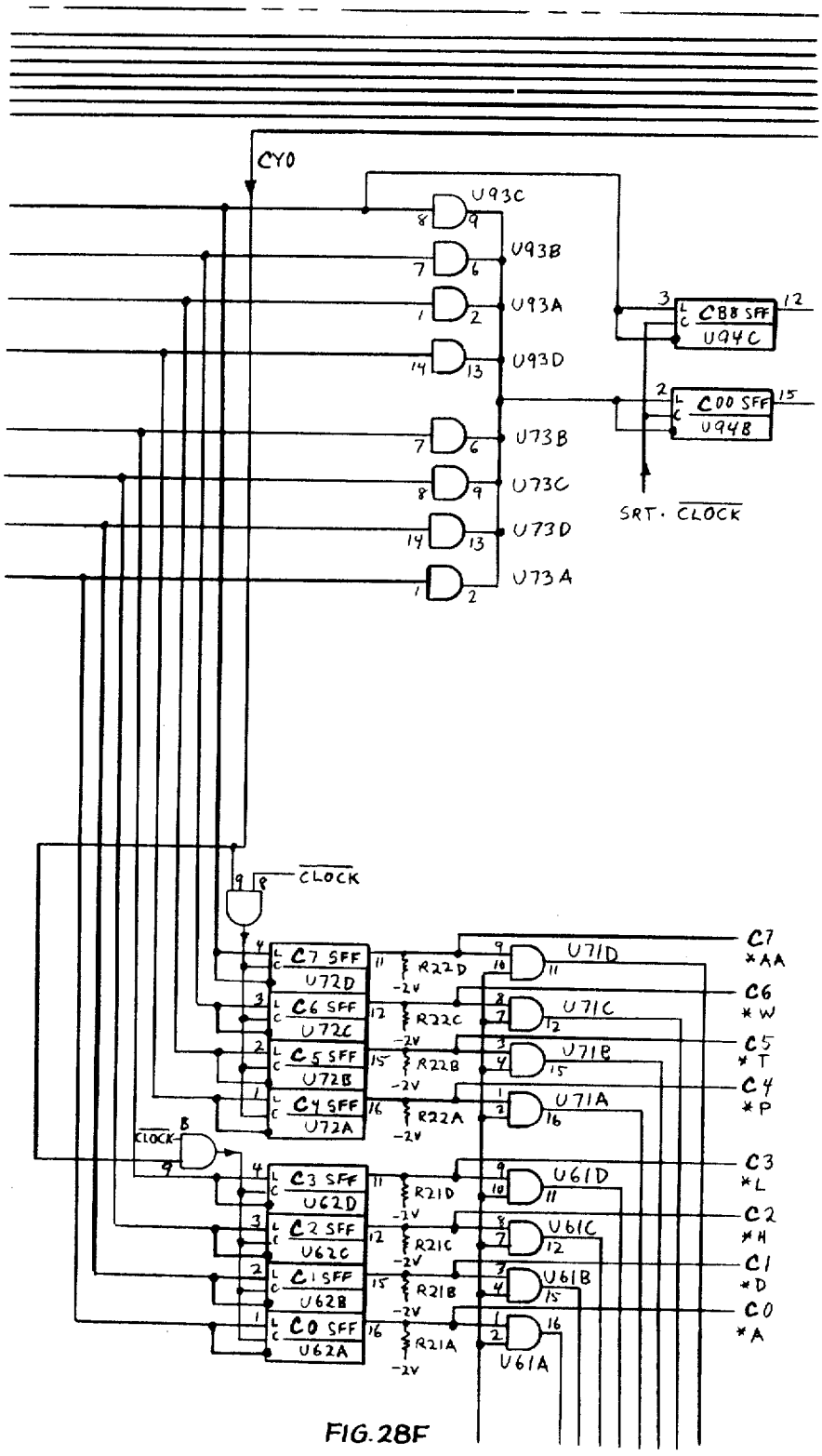

CY0
U93C
U93B
U93A
U93D
U73B
U73C
U73D
U73A
C88 SFF
U94C
C00 SFF
U94B
SRT. CLOCK
CLOCK
C7 SFF
U72D
C6 SFF
U72C
C5 SFF
U72B
C4 SFF
U72A
C3 SFF
U62D
C2 SFF
U62C
C1 SFF
U62B
C0 SFF
U62A
R22D
R22C
R22B
R22A
R21D
R21C
R21B
R21A
-2V
U71D
U71C
U71B
U71A
U61D
U61C
U61B
U61A
C7
*AA
C6
*W
C5
*T
C4
*P
C3
*L
C2
*H
C1
*D
C0
*A

FIG. 28F

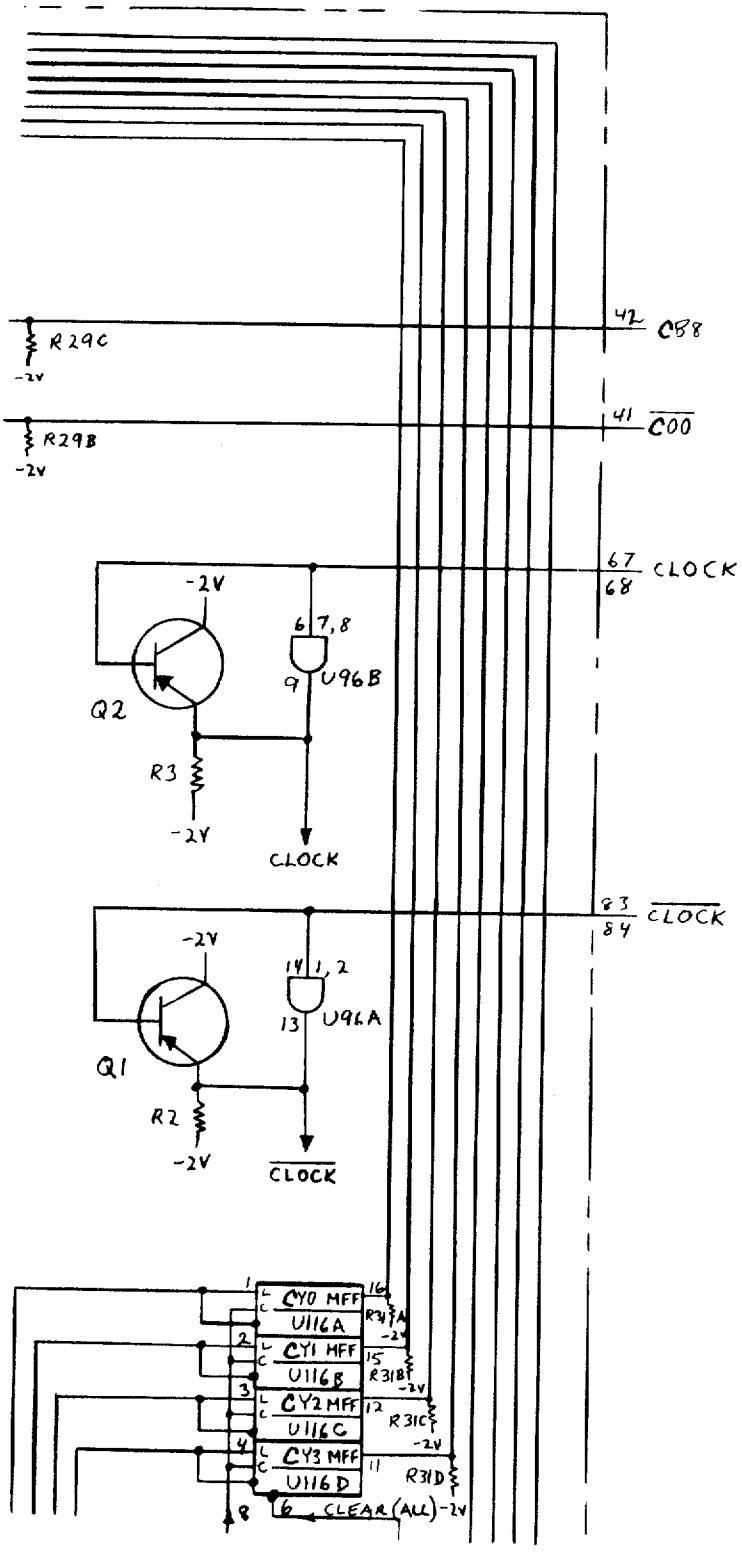
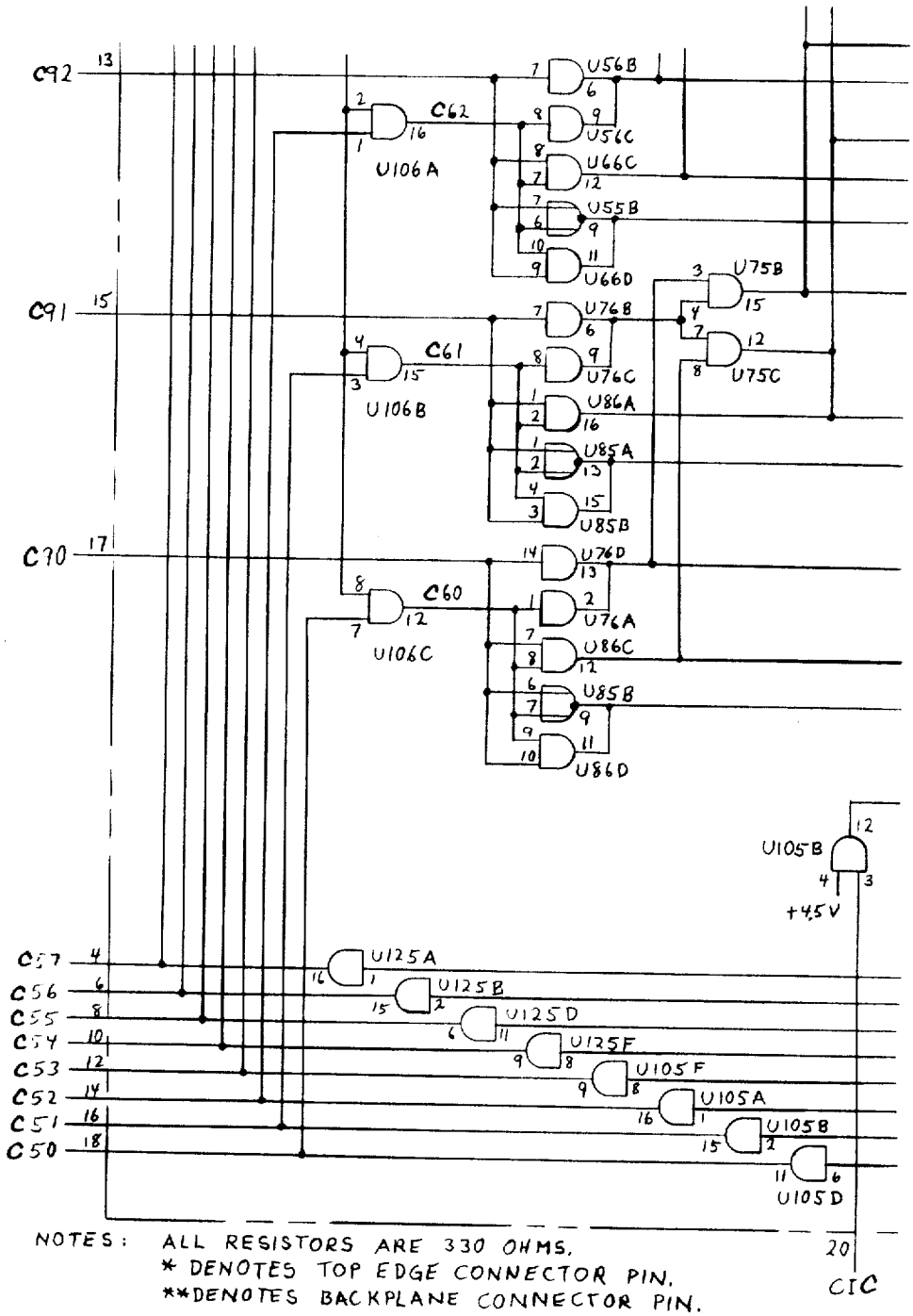

R29C
-2V
R29B
-2V
42 CB8
41 C00
67 CLOCK
68
-2V
6 7, 8
9 U96B
Q2
R3
-2V
CLOCK
83 CLOCK
84
-2V
14 1, 2
13 U96A
Q1
R2
-2V
CLOCK
CY0 MFF U116A
CY1 MFF U116B
CY2 MFF U116C
CY3 MFF U116D
R31A
R31B
R31C
R31D
-2V
CLEAR (ALL) -2V

FIG. 28G




NOTES: ALL RESISTORS ARE 330 OHMS.
* DENOTES TOP EDGE CONNECTOR PIN.
**DENOTES BACKPLANE CONNECTOR PIN.

FIG. 28H

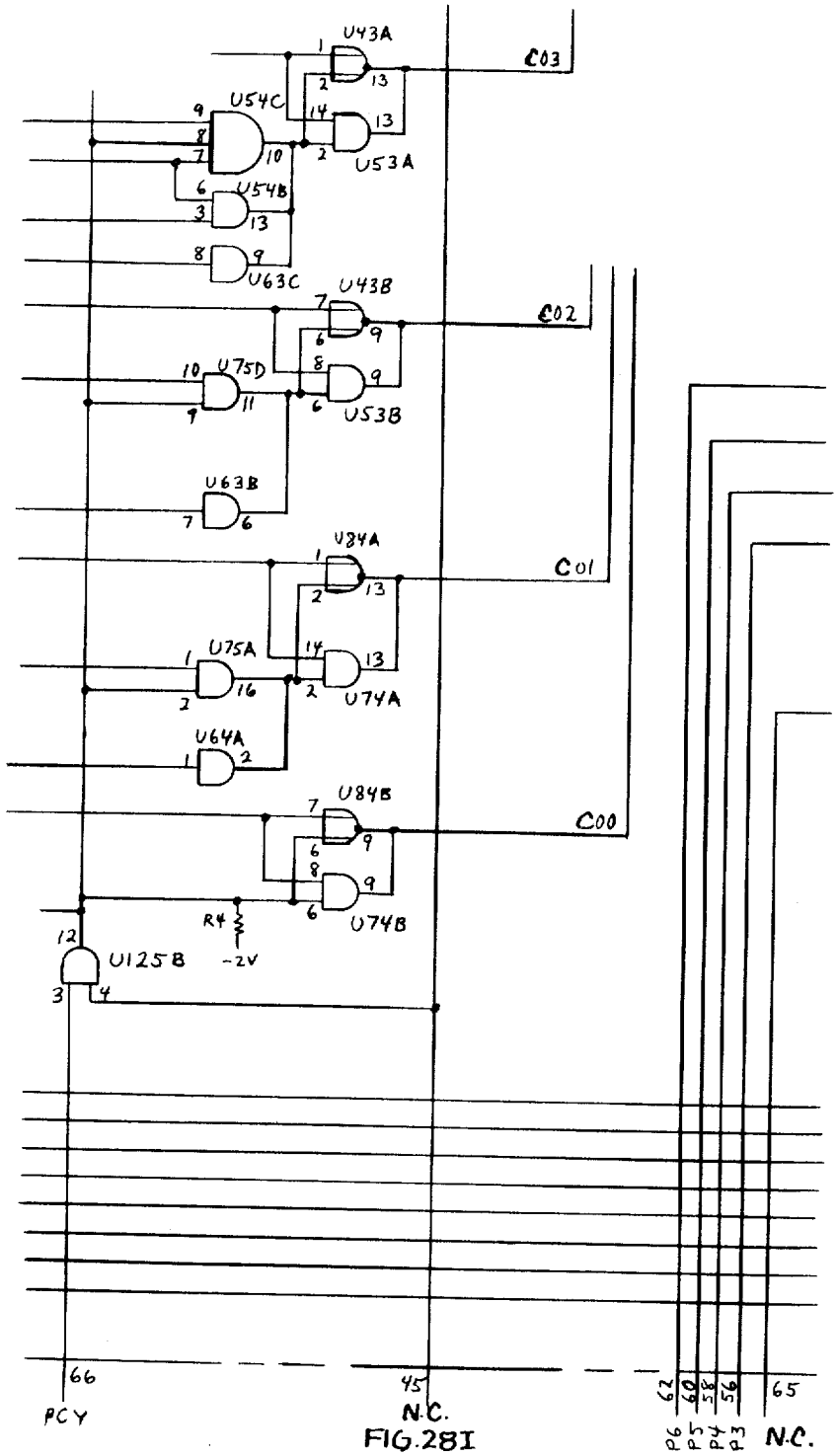

U43A
C03
U54C
U53A
U54B
U63C
U43B
C02
U75D
U53B
U63B
U84A
C01
U75A
U74A
U64A
U84B
C00
R4
-2V
U125B
U74B
66
PCY
45
N.C.
62
60
58
56
65
P6
P5
P4
P3
N.C.

FIG. 28I

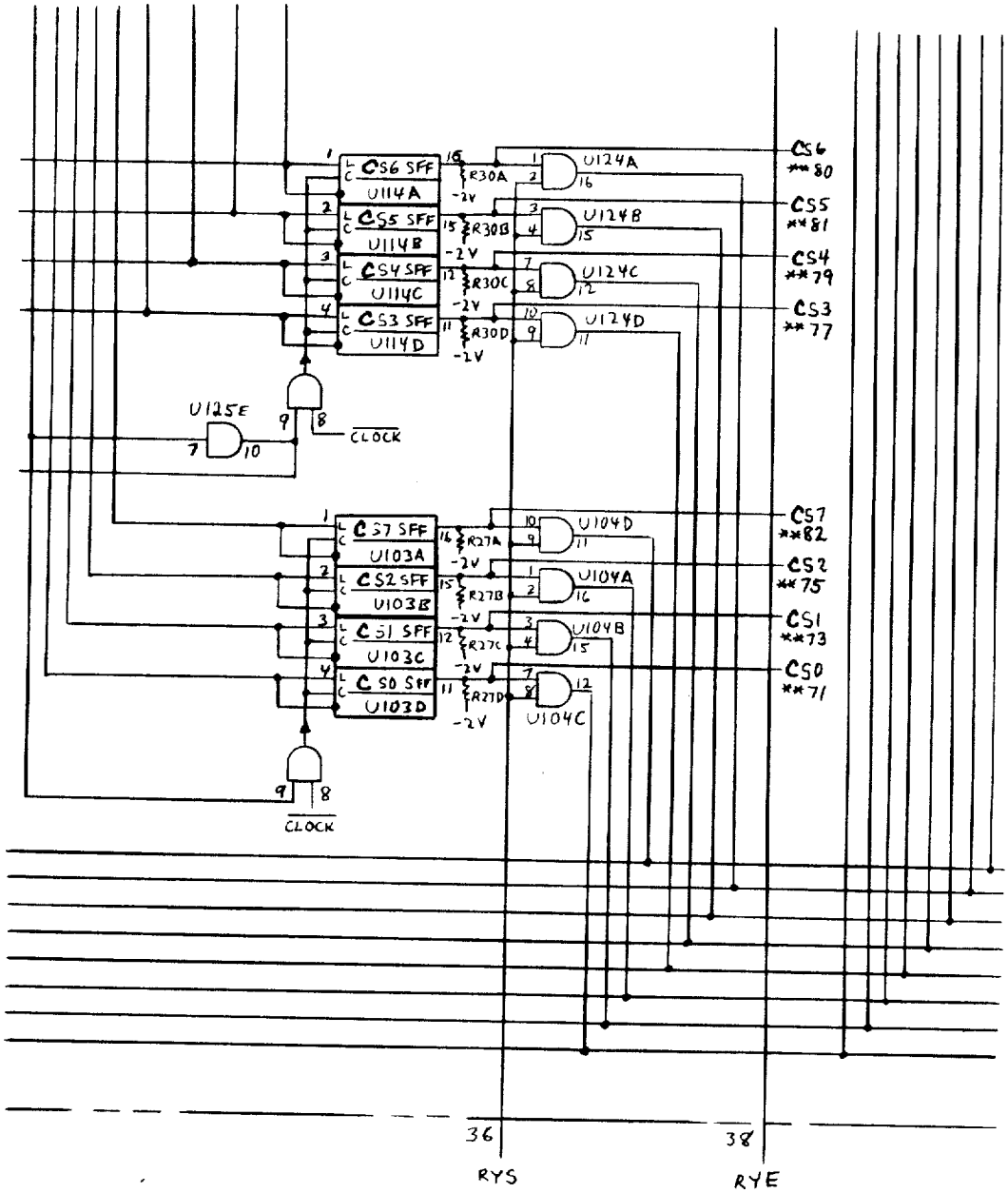

CS6 SFF
U114A
CS5 SFF
U114B
CS4 SFF
U114C
CS3 SFF
U114D
U125E
CLOCK
CS7 SFF
U103A
CS2 SFF
U103B
CS1 SFF
U103C
CS0 SFF
U103D
CLOCK
R30A
R30B
R30C
R30D
R27A
R27B
R27C
R27D
-2V
U124A
U124B
U124C
U124D
U104D
U104A
U104B
U104C
CS6 **80
CS5 **81
CS4 **79
CS3 **77
CS7 **82
CS2 **75
CS1 **73
CS0 **71
36
38
RYS
RYE

FIG. 28d

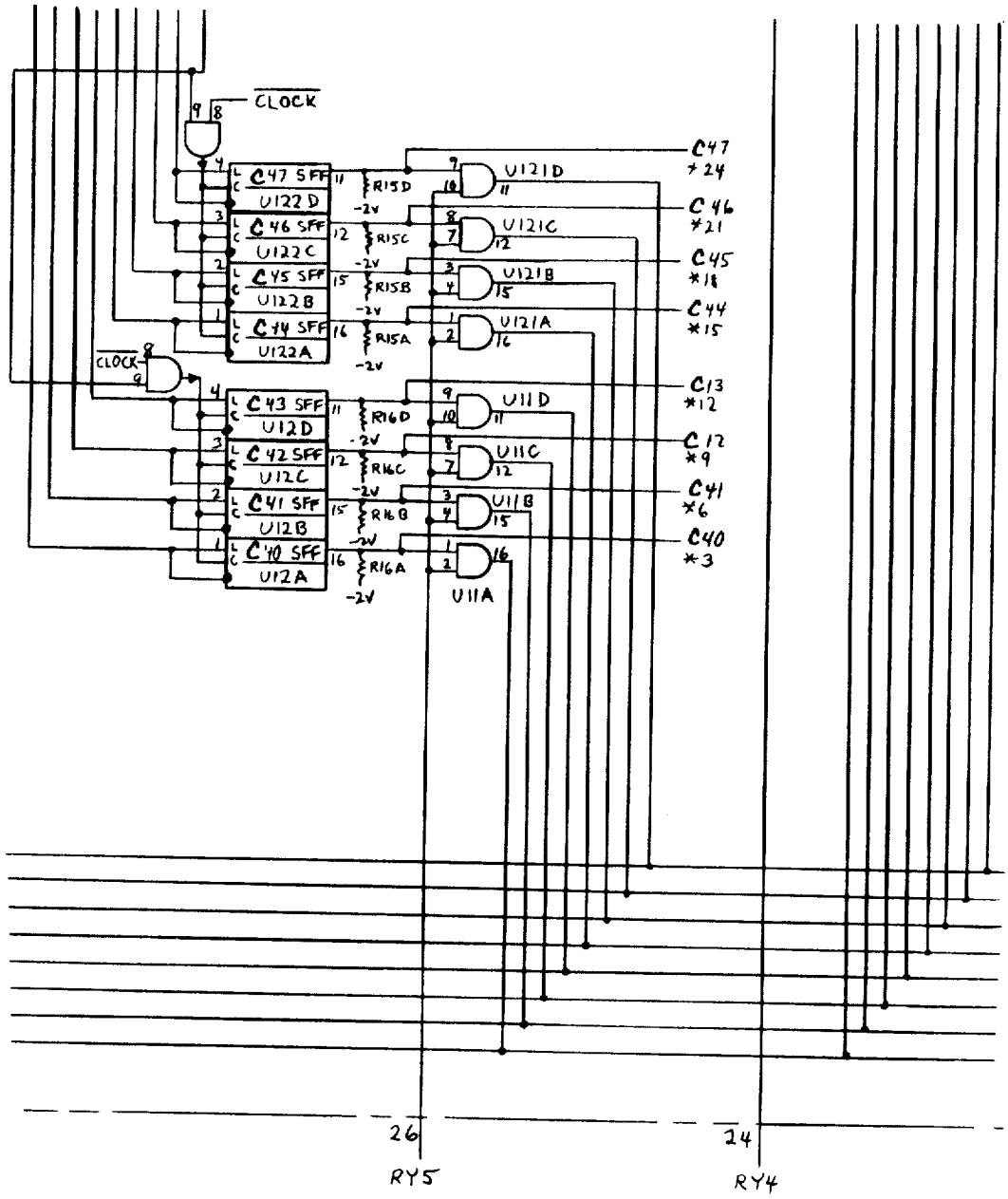

CLOCK
C47 SFF
U122D
C46 SFF
U122C
C45 SFF
U122B
C44 SFF
U122A
C43 SFF
U12D
C42 SFF
U12C
C41 SFF
U12B
C40 SFF
U12A
R15D
R15C
R15B
R15A
R16D
R16C
R16B
R16A
-2V
U121D
U121C
U121B
U121A
U11D
U11C
U11B
U11A
C47 *24
C46 *21
C45 *18
C44 *15
C13 *12
C12 *9
C41 *6
C40 *3
26
RY5
24
RY4

FIG. 28K

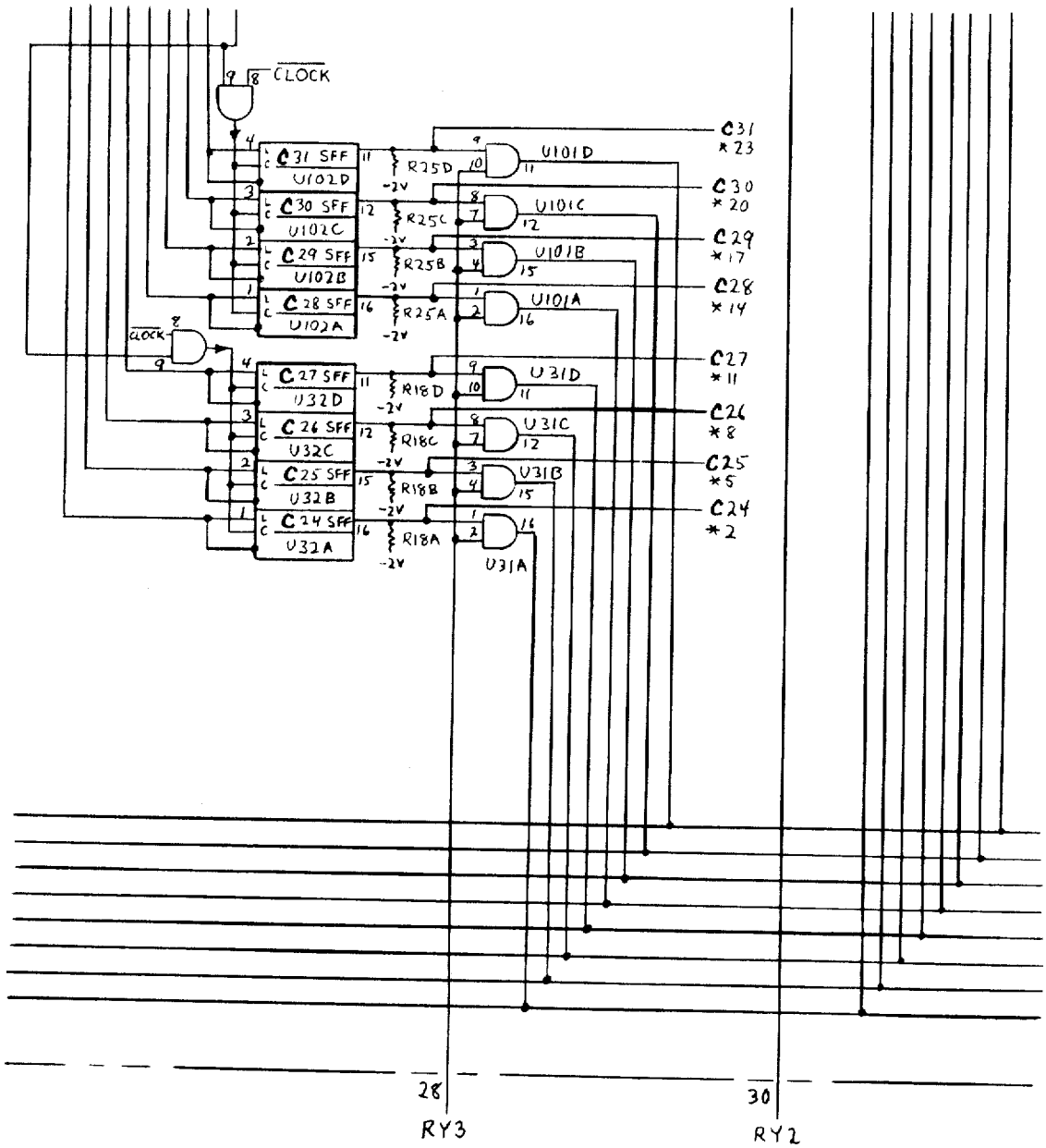

CLOCK
C31 SFF
U102D
C30 SFF
U102C
C29 SFF
U102B
C28 SFF
U102A
C27 SFF
U32D
C26 SFF
U32C
C25 SFF
U32B
C24 SFF
U32A
R25D
R25C
R25B
R25A
R18D
R18C
R18B
R18A
-2V
U101D
U101C
U101B
U101A
U31D
U31C
U31B
U31A
C31 *23
C30 *20
C29 *17
C28 *14
C27 *11
C26 *8
C25 *5
C24 *2
28
RY3
30
RY2

FIG. 28L

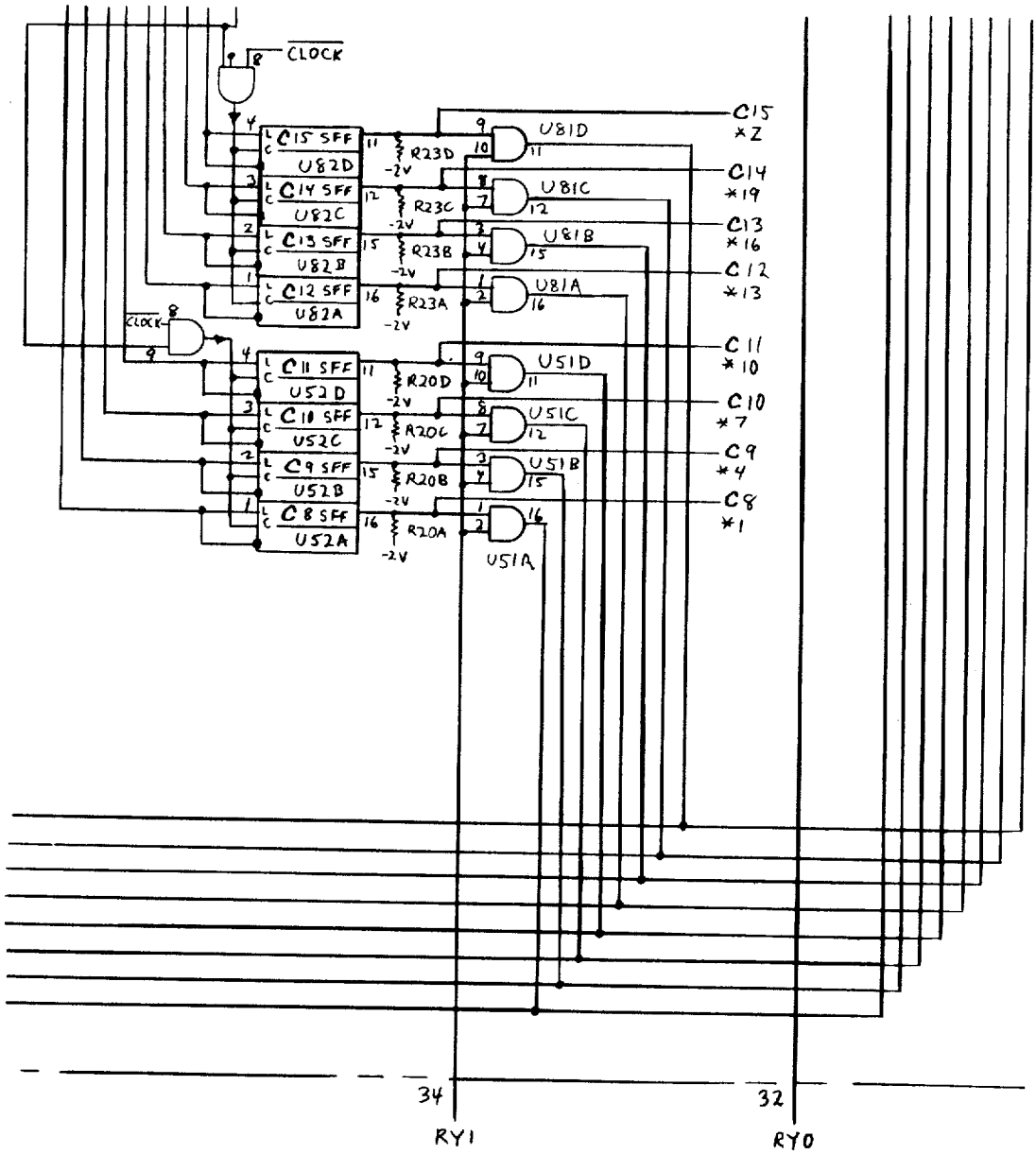

CLOCK
C15 SFF
U82D
C14 SFF
U82C
C13 SFF
U82B
C12 SFF
U82A
C11 SFF
U52D
C10 SFF
U52C
C9 SFF
U52B
C8 SFF
U52A
R23D
R23C
R23B
R23A
R20D
R20C
R20B
R20A
-2V
U81D
U81C
U81B
U81A
U51D
U51C
U51B
U51A
C15 *2
C14 *19
C13 *16
C12 *13
C11 *10
C10 *7
C9 *4
C8 *1
34
32
RY1
RY0

FIG. 28M

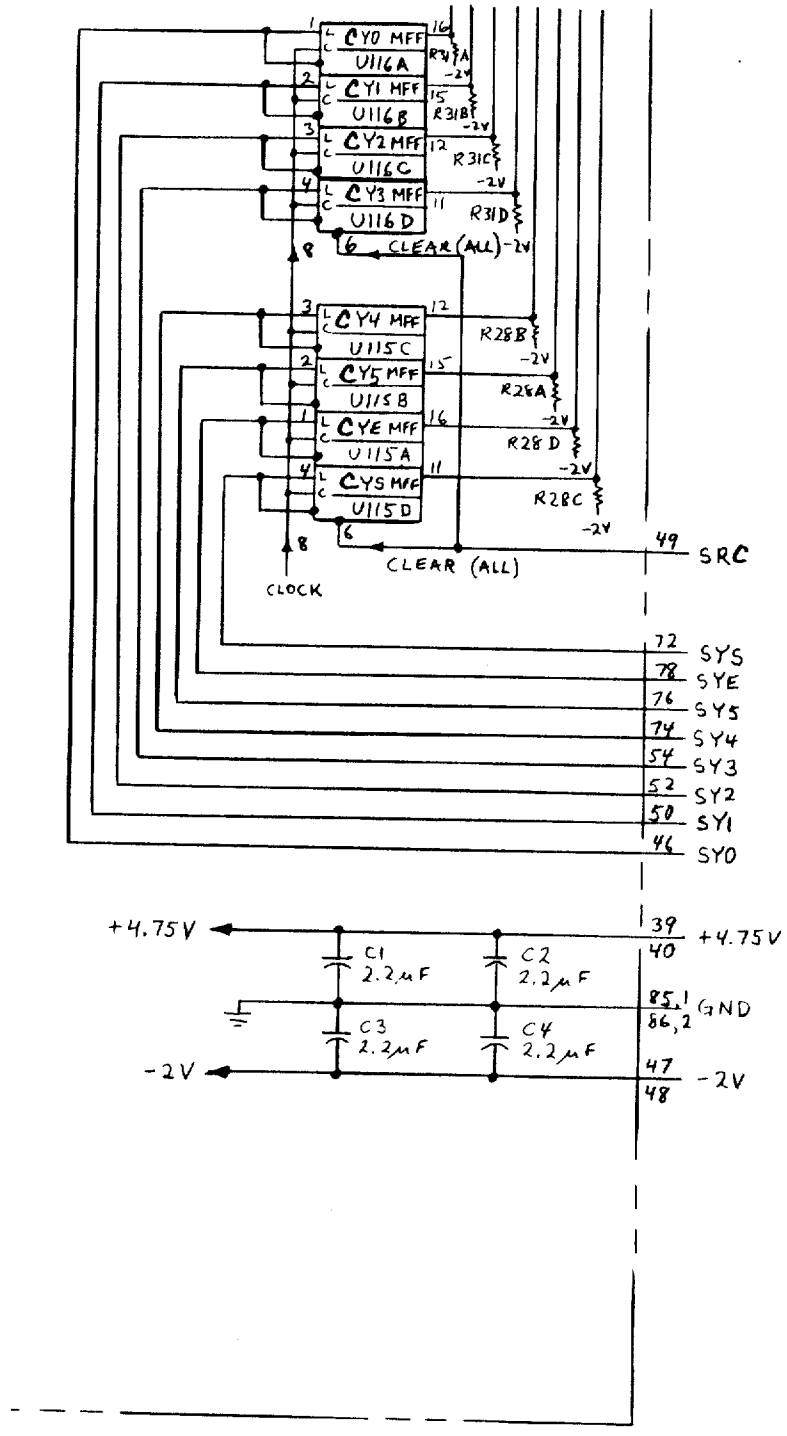

CY0 MFF
U116A
CY1 MFF
U116B
CY2 MFF
U116C
CY3 MFF
U116D
CLEAR (ALL) -2V
CY4 MFF
U115C
CY5 MFF
U115B
CYE MFF
U115A
CYS MFF
U115D
R31A
R31B
R31C
R31D
R28B
R28A
R28D
R28C
-2V
CLOCK
CLEAR (ALL)
49 SRC
72 SYS
78 SYE
76 SY5
74 SY4
54 SY3
52 SY2
50 SY1
46 SY0
+4.75V
C1 2.2μF
C2 2.2μF
C3 2.2μF
C4 2.2μF
39 +4.75V
40
85,1 GND
86,2
47 -2V
48

FIG.28N

| | | | | | | |
|---|---|---|---|---|---|---|
| FIG. 28A | FIG. 28B | FIG. 28C | FIG. 28D | FIG. 28E | FIG. 28F | FIG. 28G |
| FIG. 28H | FIG. 28I | FIG. 28J | FIG. 28K | FIG. 28L | FIG. 28M | FIG. 28N |

FIG. 29

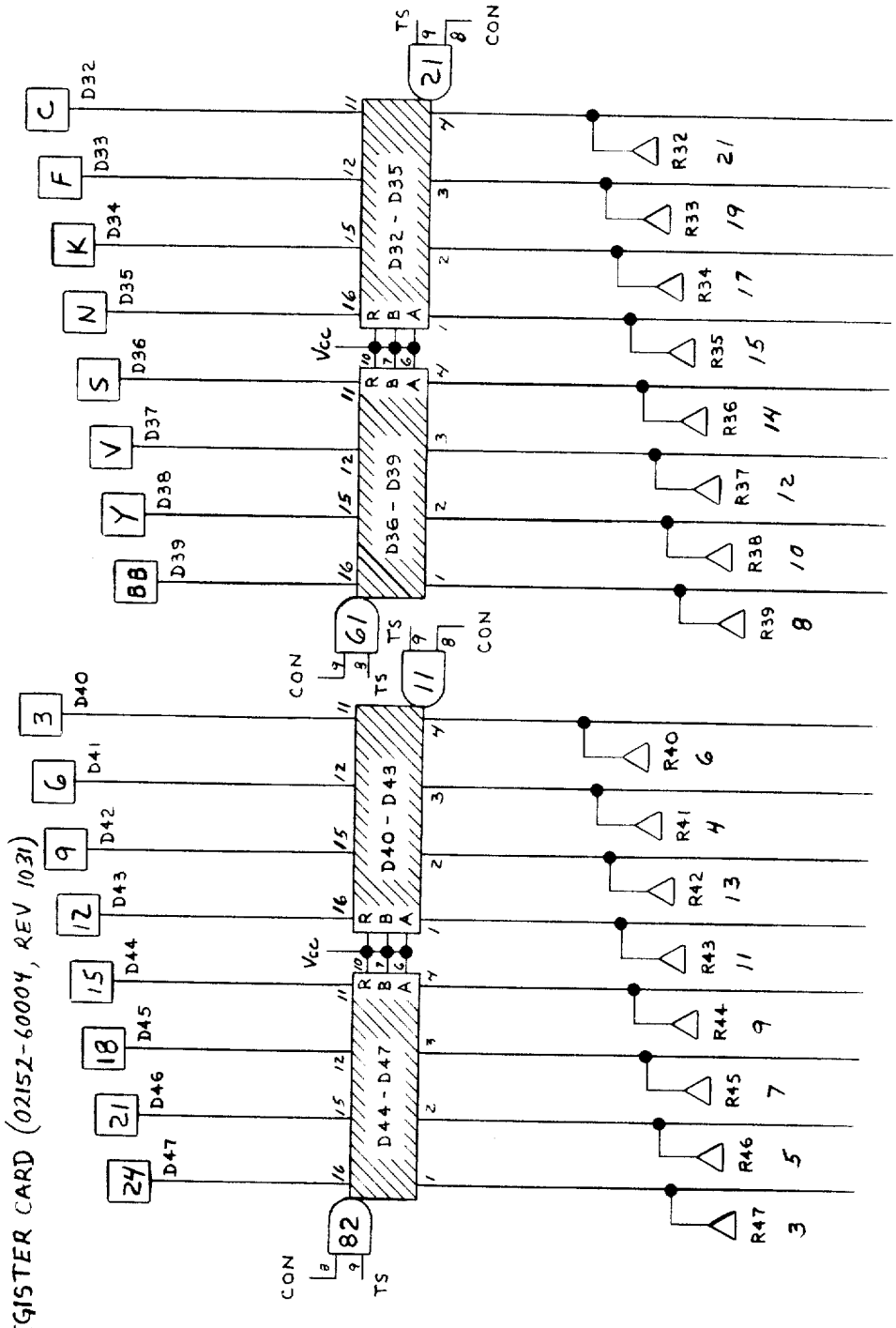

D-REGISTER CARD (02152-60004, REV 1031)
C
F
K
N
S
V
Y
BB
3
6
9
12
15
18
21
24
D32
D33
D34
D35
D36
D37
D38
D39
D40
D41
D42
D43
D44
D45
D46
D47
D32 - D35
D36 - D39
D40 - D43
D44 - D47
Vcc
CON
TS
21
61
11
82
R32
R33
R34
R35
R36
R37
R38
R39
R40
R41
R42
R43
R44
R45
R46
R47

FIG. 30A

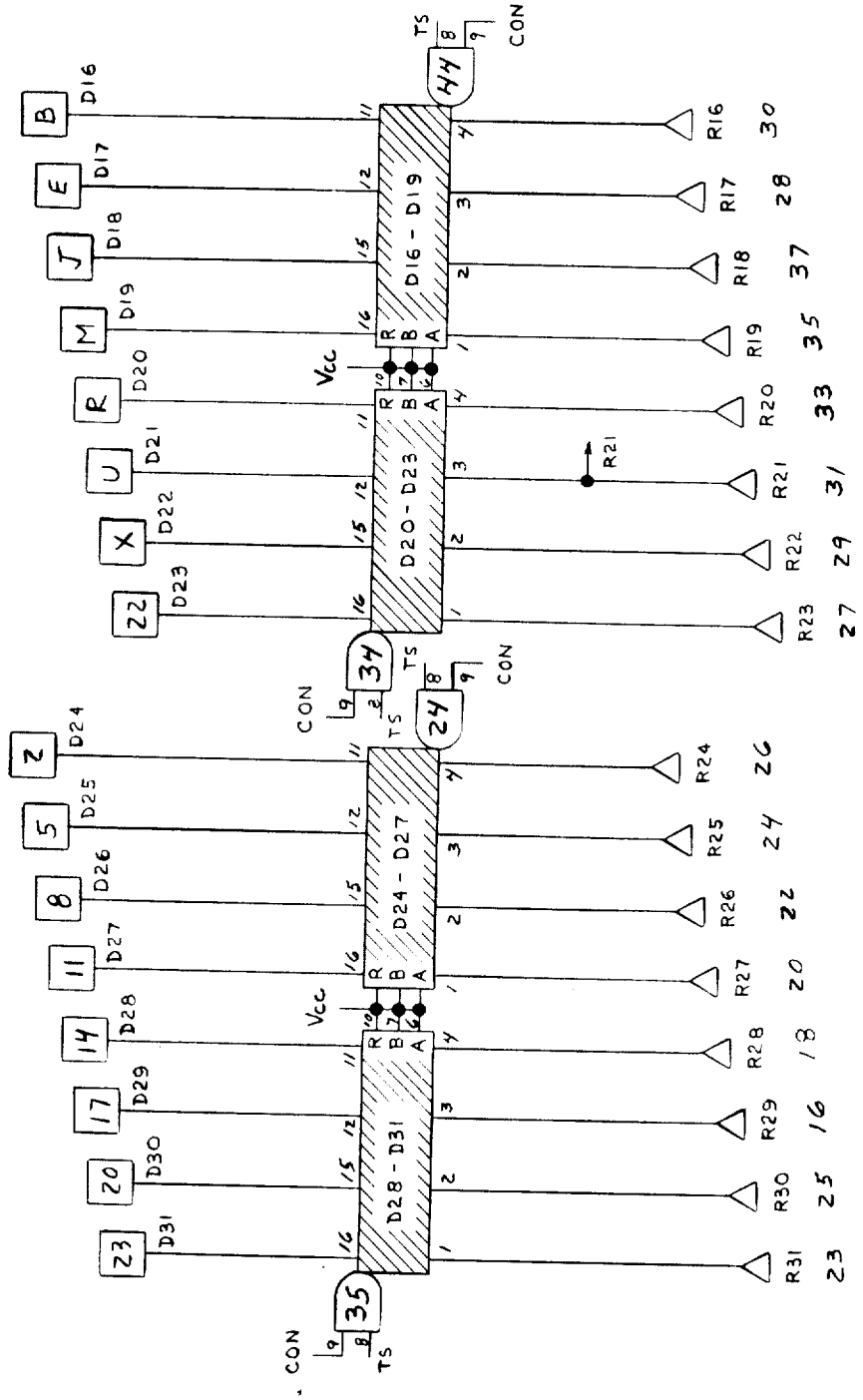

B
D16
E
D17
J
D18
M
D19
R
D20
U
D21
X
D22
22
D23
Z
D24
5
D25
8
D26
11
D27
14
D28
17
D29
20
D30
23
D31
D16 - D19
D20 - D23
D24 - D27
D28 - D31
Vcc
R21
R16
R17
R18
R19
R20
R21
R22
R23
R24
R25
R26
R27
R28
R29
R30
R31
30
28
37
35
33
31
29
27
26
24
22
20
18
16
25
23
CON
TS
44
34
24
35

FIG.30B

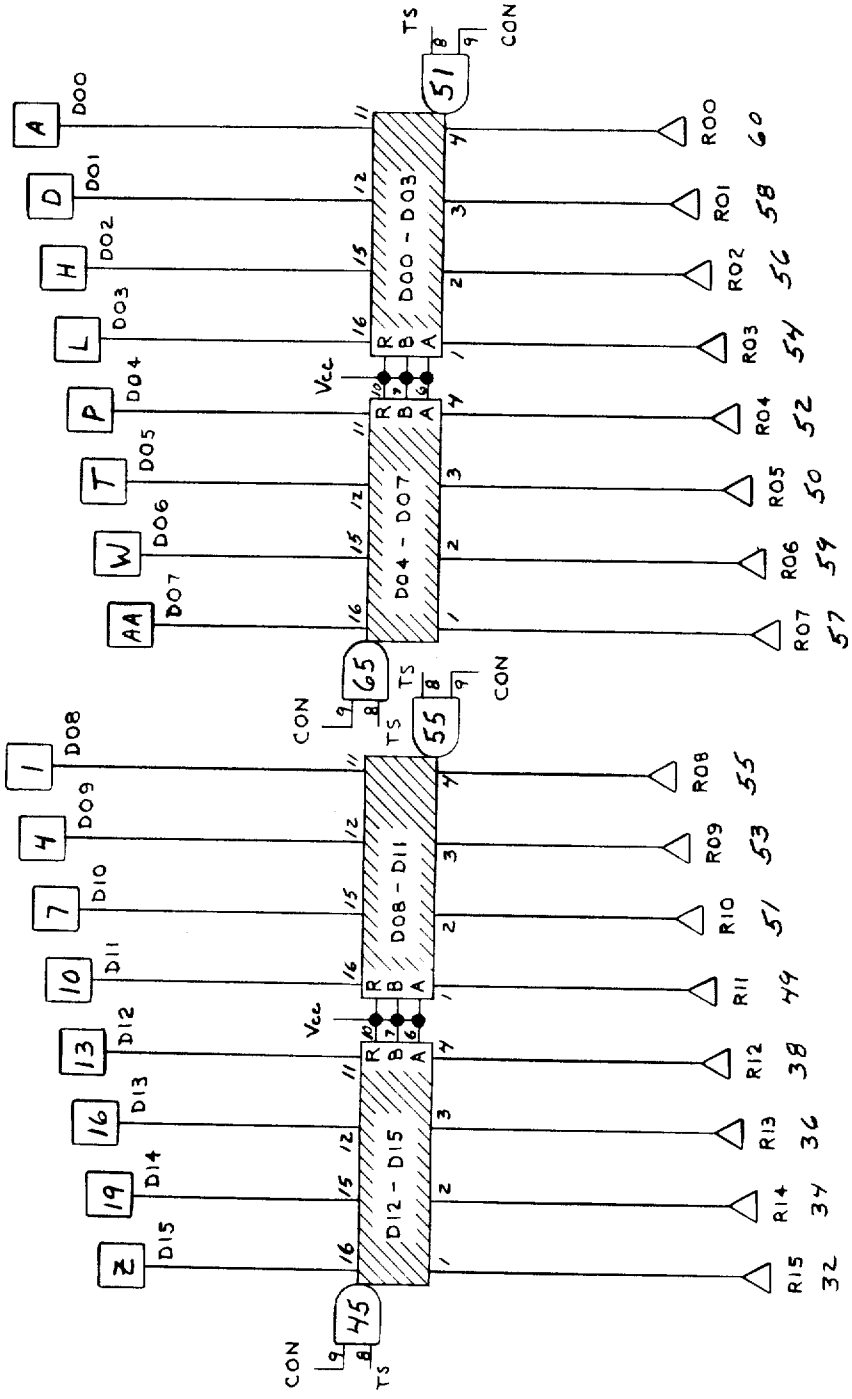

A
D
H
L
P
T
W
AA
1
4
7
10
13
16
19
Z
D00
D01
D02
D03
D04
D05
D06
D07
D08
D09
D10
D11
D12
D13
D14
D15
TS
CON
Vcc
51
65
55
45
D00 - D03
D04 - D07
D08 - D11
D12 - D15
R
B
A
R00
R01
R02
R03
R04
R05
R06
R07
R08
R09
R10
R11
R12
R13
R14
R15
60
58
56
54
52
50
59
57
55
53
51
49
38
36
34
32

FIG.30C

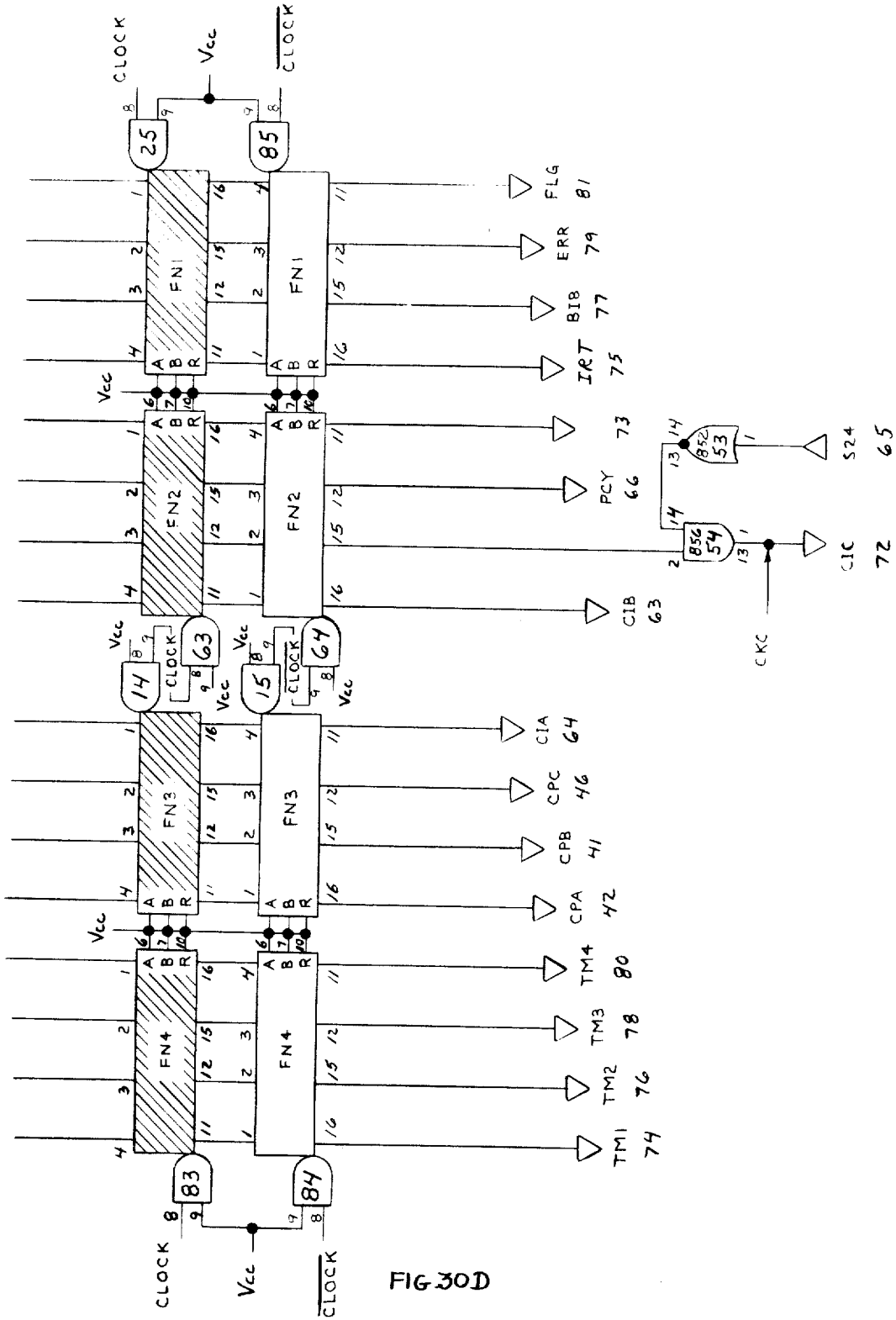

CLOCK
Vcc
CLOCK
25
85
FN1
FN1
FLG 81
ERR 79
BIB 77
IRT 75
FN2
FN2
PCY 73
66
C1B 63
S24 65
C1C 72
CKC
63
64
14
15
FN3
FN3
CIA 64
CPC 46
CPB 41
CPA 42
FN4
FN4
TM4 80
TM3 78
TM2 76
TM1 74
83
84

FIG 30D

10 MHz OSCILLATOR
DESIGN TAKEN FROM 2116
TIMING GENERATOR CARD
(02116 - 6281)

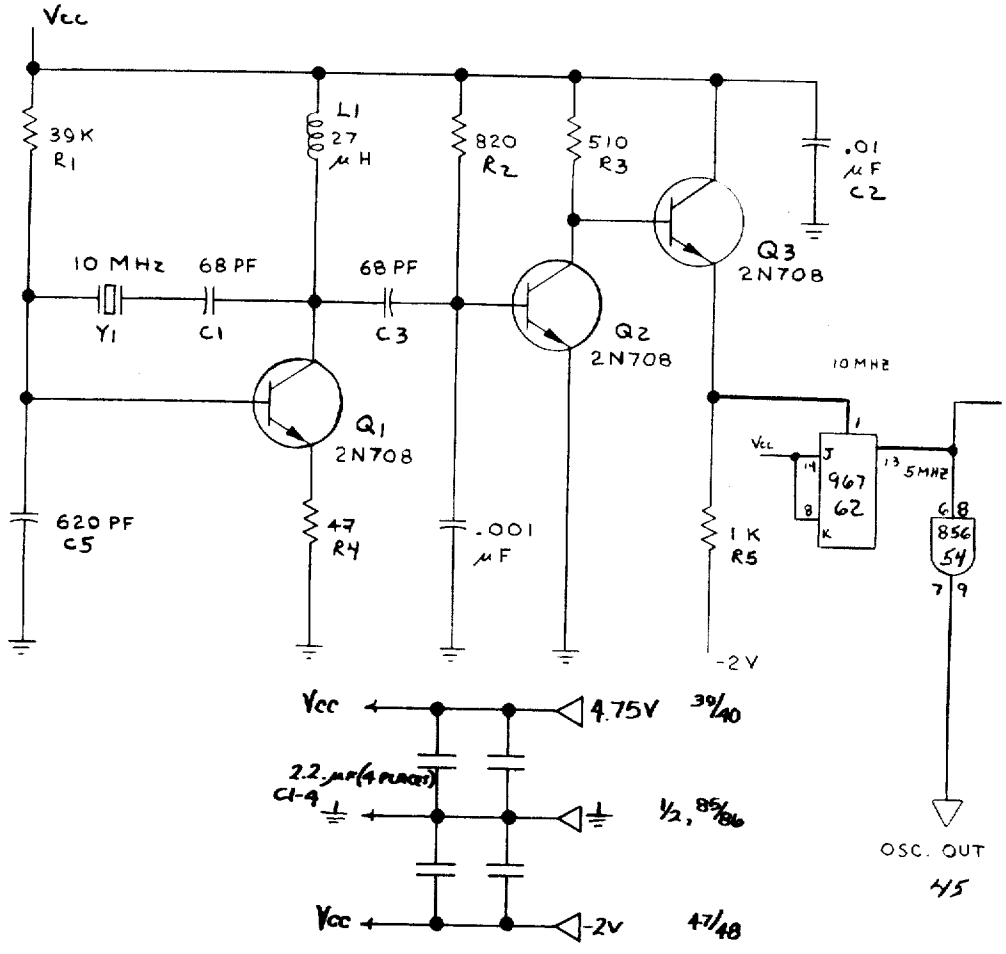



FIG. 30E

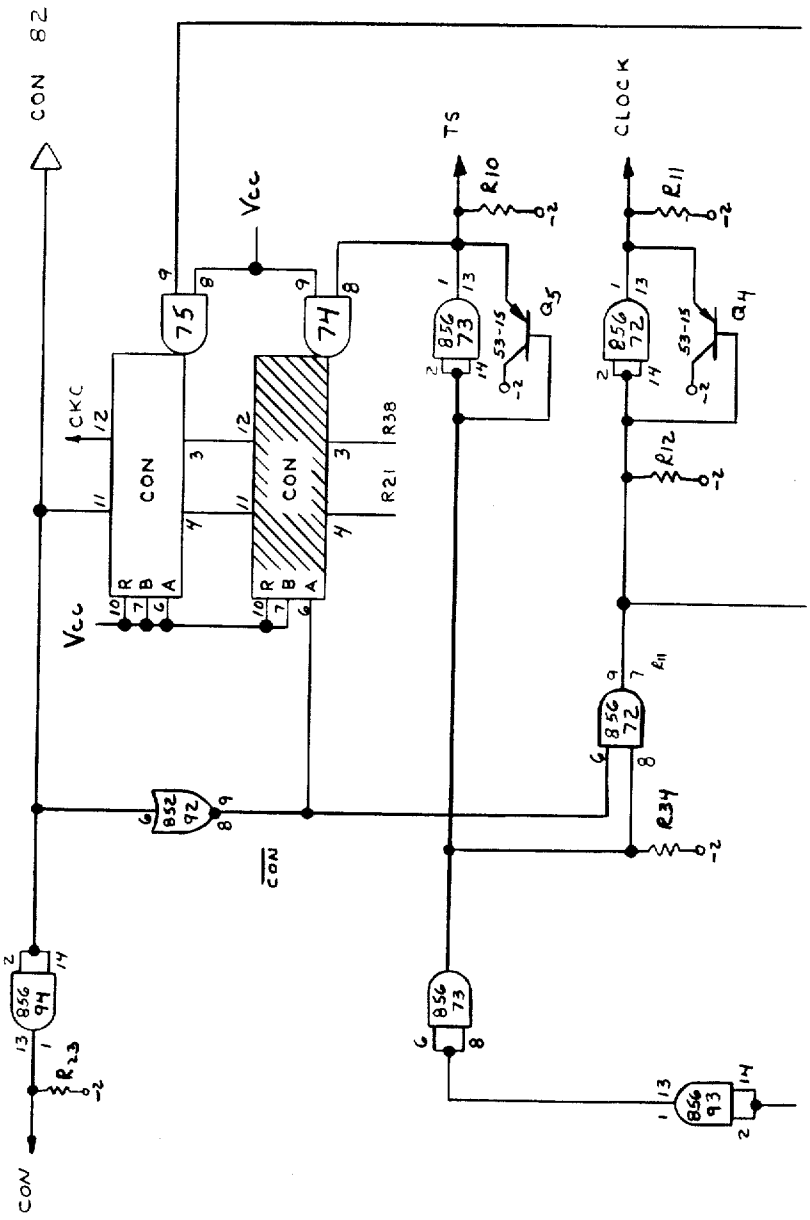

CON 82
Vcc
TS
CLOCK
R10
R11
R12
R34
R23
Q5
Q4
75
74
CON
CKC
R38
R21
856 73
856 72
856 92
856 94
856 93
53-15
CON
CON

FIG.30F

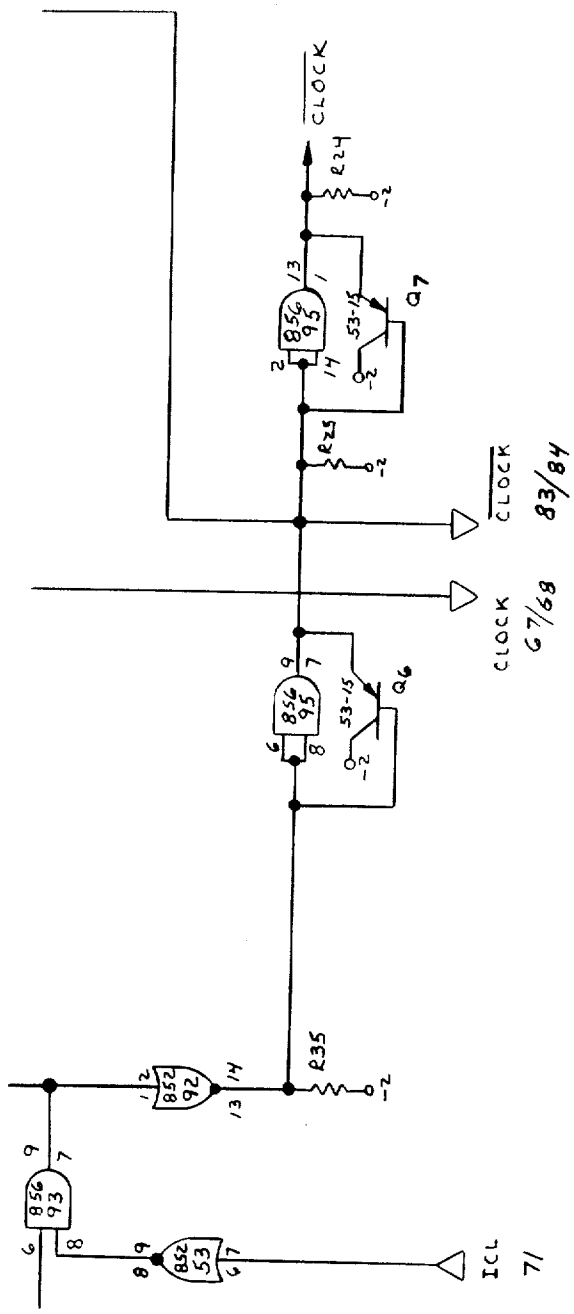



FIG. 30G

NOTE:

1. ALL RESISTORS ARE 330Ω ¼ WATT 5% EXCEPT WHERE INDICATED
2. ALL QUAD LATCHES HAVE AN EXTERNAL PULL DOWN RESISTOR ADJACENT TO OUTPUT

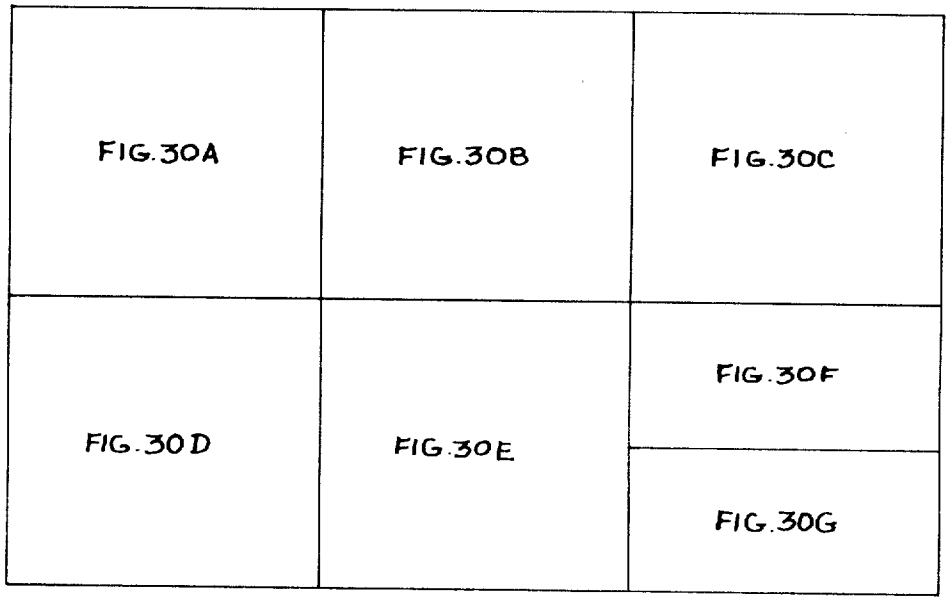



FIG. 31

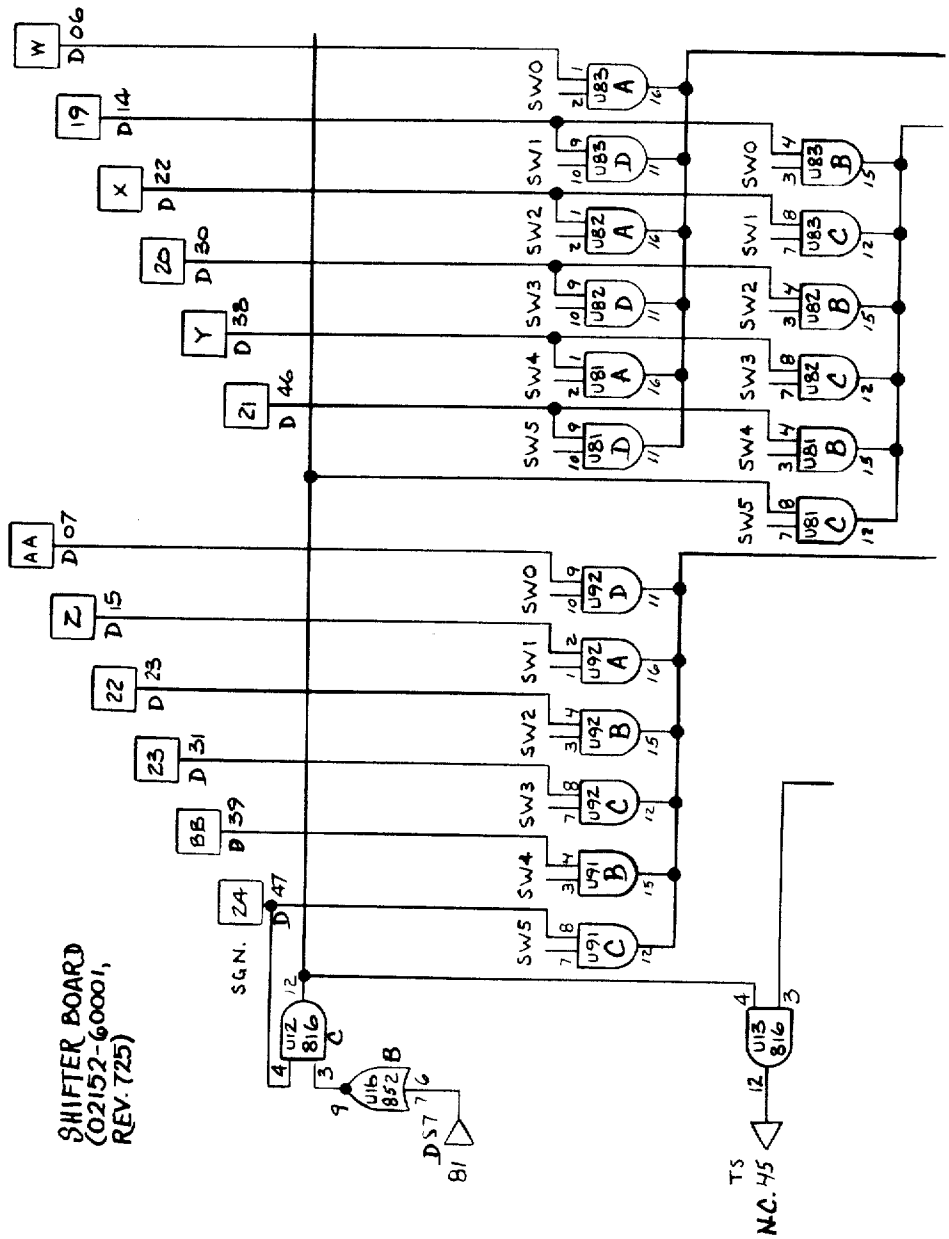

W D06
19 D14
X D22
20 D30
Y D38
21 D46
AA D07
Z D15
22 D23
23 D31
BB D39
ZA D47
SW0
SW1
SW2
SW3
SW4
SW5
U83 A
U83 D
U82 A
U82 D
U81 A
U81 D
U92 D
U92 A
U92 B
U92 C
U91 B
U91 C
U83 B
U83 C
U82 B
U82 C
U81 B
U81 C
SGN.
U12 816 C
U16 852 B
DS7 B1
U13 816
TS N.C. 45
SHIFTER BOARD (02152-60001, REV. 725)

FIG 32A

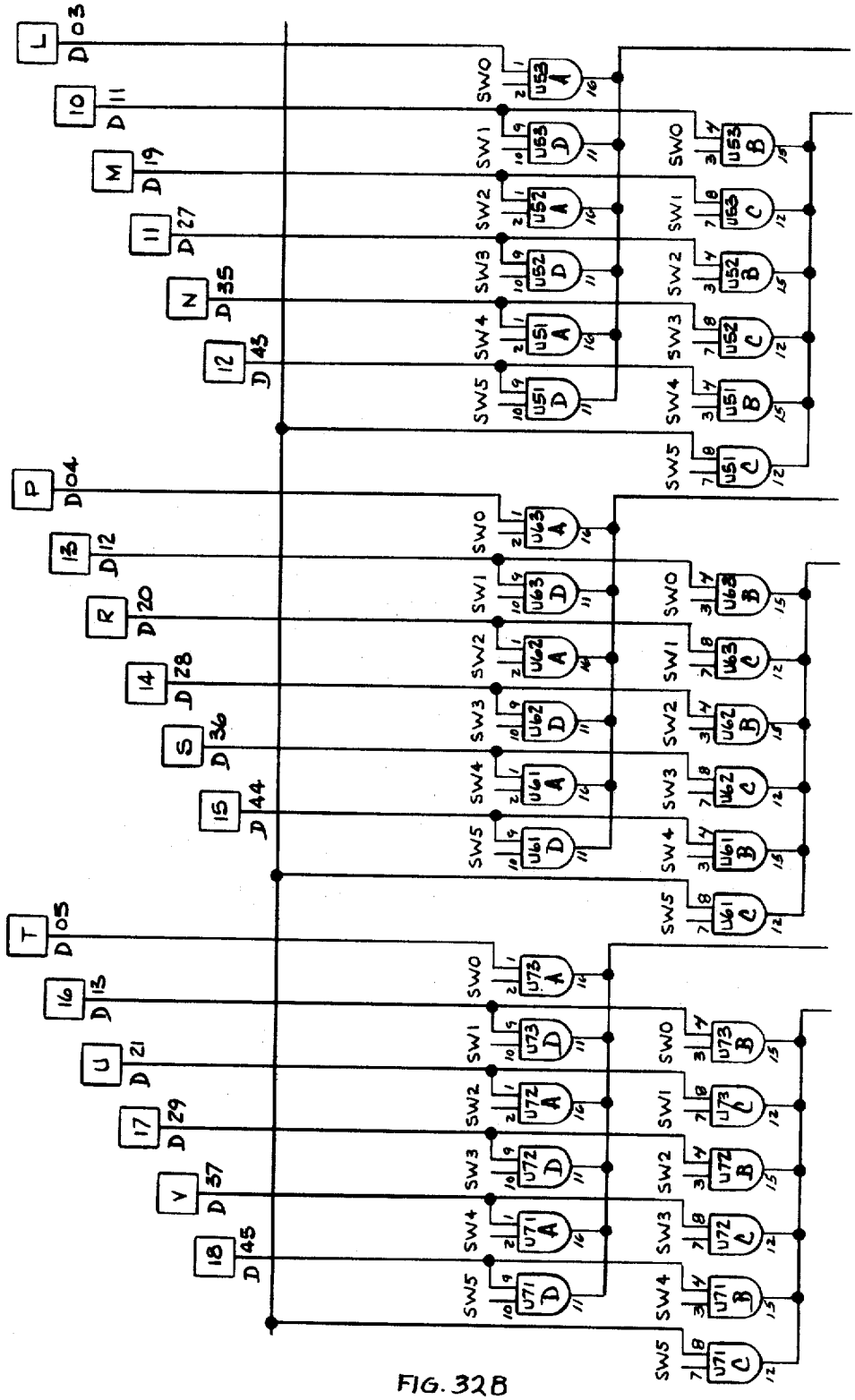

L D03
10 D11
M D19
11 D27
N D35
12 D45
P D04
13 D12
R D20
14 D28
S D36
15 D44
T D05
16 D13
U D21
17 D29
V D37
18 D45
SW0
SW1
SW2
SW3
SW4
SW5
U53 A
U53 D
U52 A
U52 D
U51 A
U51 D
U63 A
U63 D
U62 A
U62 D
U61 A
U61 D
U73 A
U73 D
U72 A
U72 D
U71 A
U71 D
U53 B
U53 C
U52 B
U52 C
U51 B
U51 C
U63 B
U63 C
U62 B
U62 C
U61 B
U61 C
U73 B
U73 C
U72 B
U72 C
U71 B
U71 C

FIG. 32B

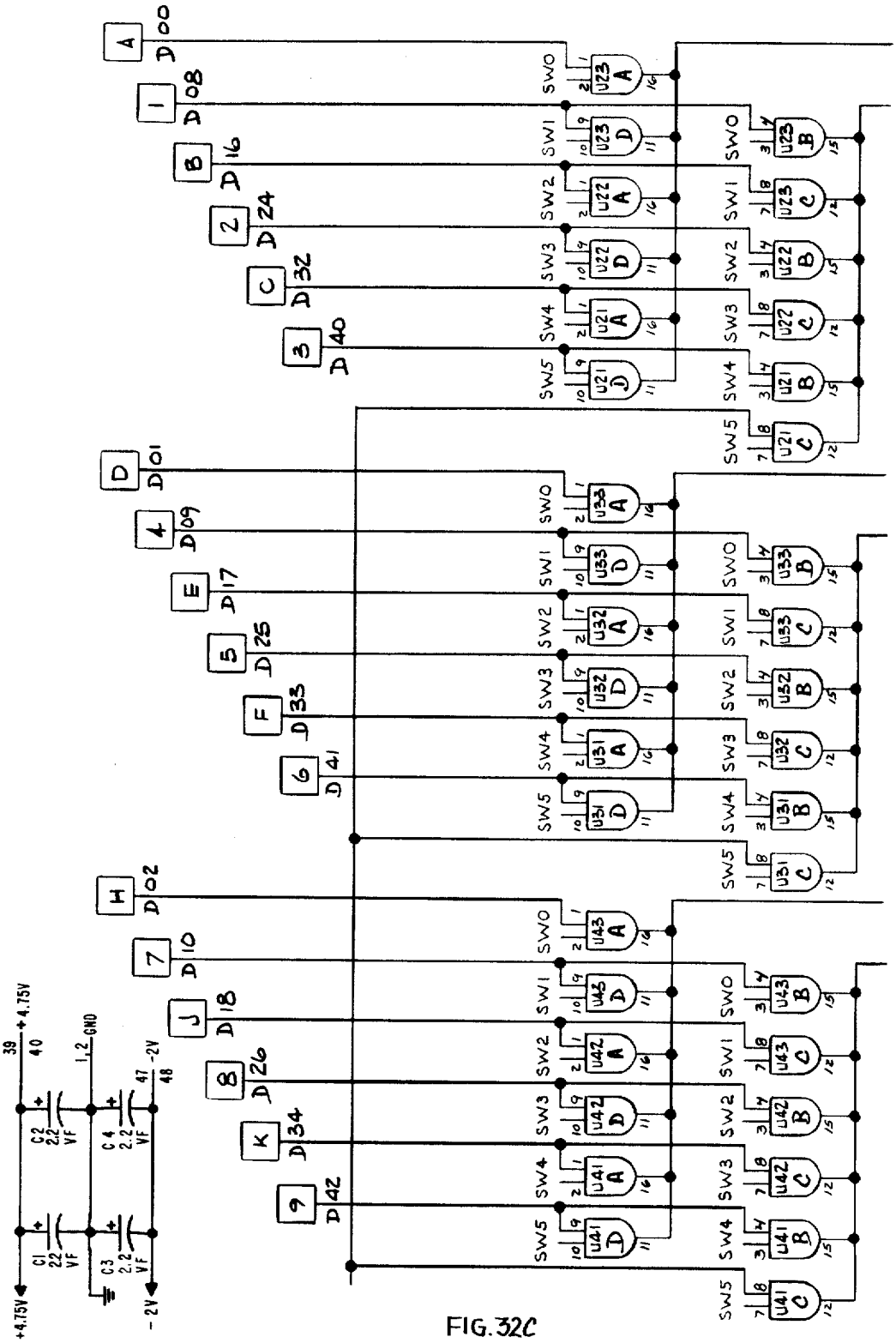
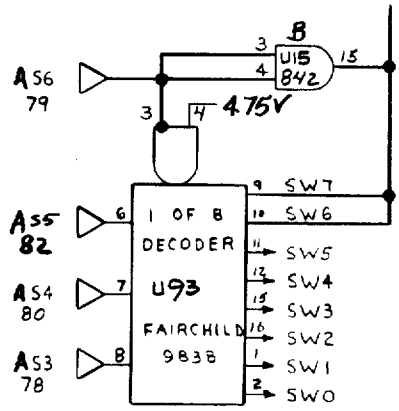

A
D00
I
D08
B
D16
2
D24
C
D32
3
D40
D
D01
4
D09
E
D17
5
D25
F
D33
6
D41
H
D02
7
D10
J
D18
8
D26
K
D34
9
D42
SW0
SW1
SW2
SW3
SW4
SW5
U23 A
U23 D
U22 A
U22 D
U21 A
U21 D
U23 B
U23 C
U22 B
U22 C
U21 B
U21 C
U33 A
U33 D
U32 A
U32 D
U31 A
U31 D
U33 B
U33 C
U32 B
U32 C
U31 B
U31 C
U43 A
U43 D
U42 A
U42 D
U41 A
U41 D
U43 B
U43 C
U42 B
U42 C
U41 B
U41 C
+4.75V
GND
-2V
C1 2.2 VF
C2 2.2 VF
C3 2.2 VF
C4 2.2 VF

FIG. 32C


B
U15
842
AS6
79
4.75V
AS5
82
AS4
80
AS3
78
1 OF 8
DECODER
U93
FAIRCHILD
9838
SW7
SW6
SW5
SW4
SW3
SW2
SW1
SW0

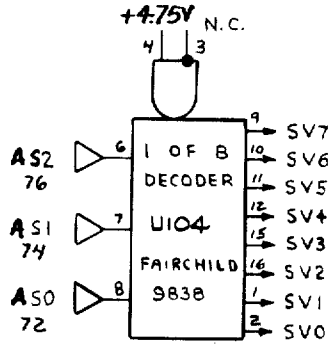

+4.75V
N.C.
AS2
76
AS1
74
AS0
72
1 OF 8
DECODER
U104
FAIRCHILD
9838
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0

FIG. 32D

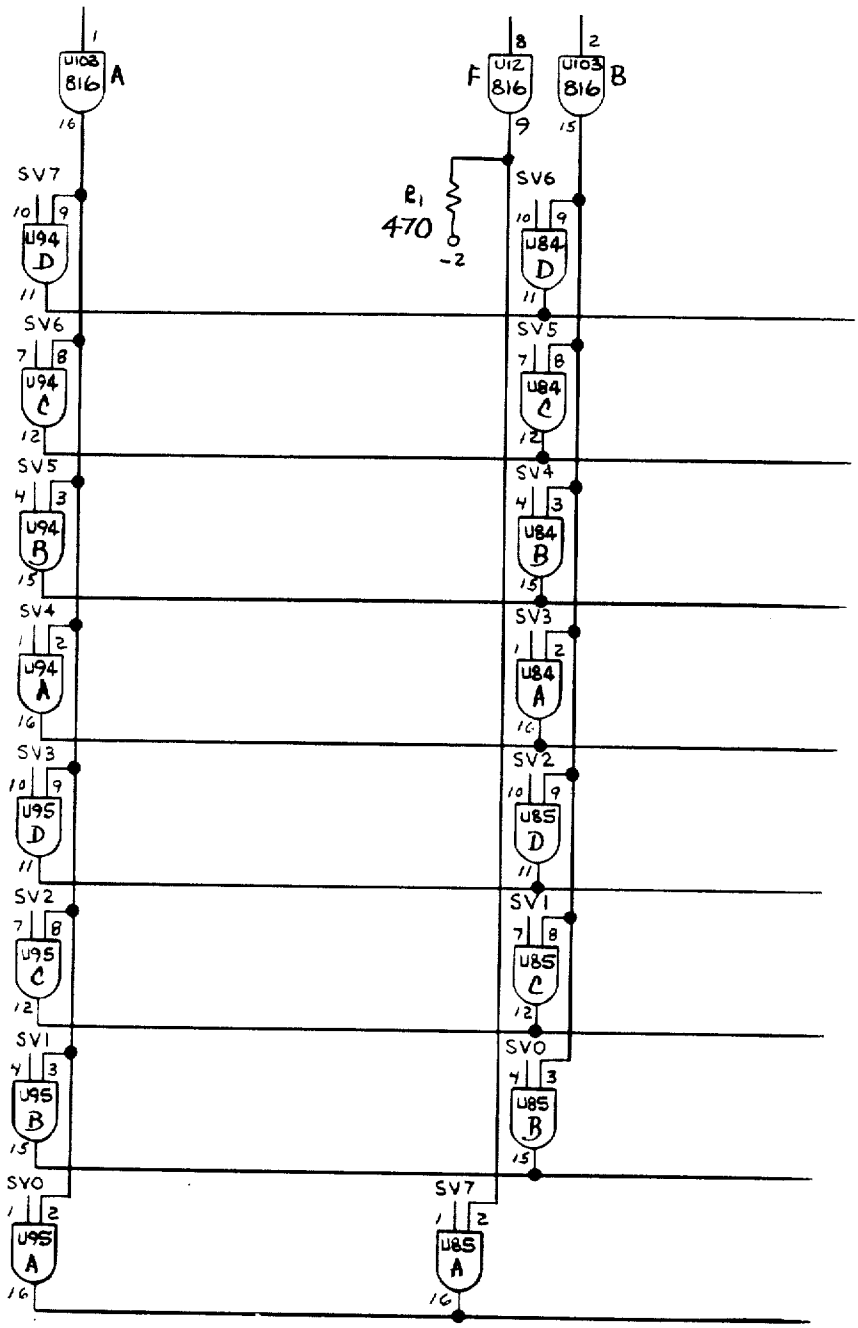

U103 816 A
U12 816 F
U103 816 B
R1 470
-2
SV7
SV6
SV5
SV4
SV3
SV2
SV1
SV0
U94 D
U94 C
U94 B
U94 A
U95 D
U95 C
U95 B
U95 A
U84 D
U84 C
U84 B
U84 A
U85 D
U85 C
U85 B
U85 A

FIG.32E

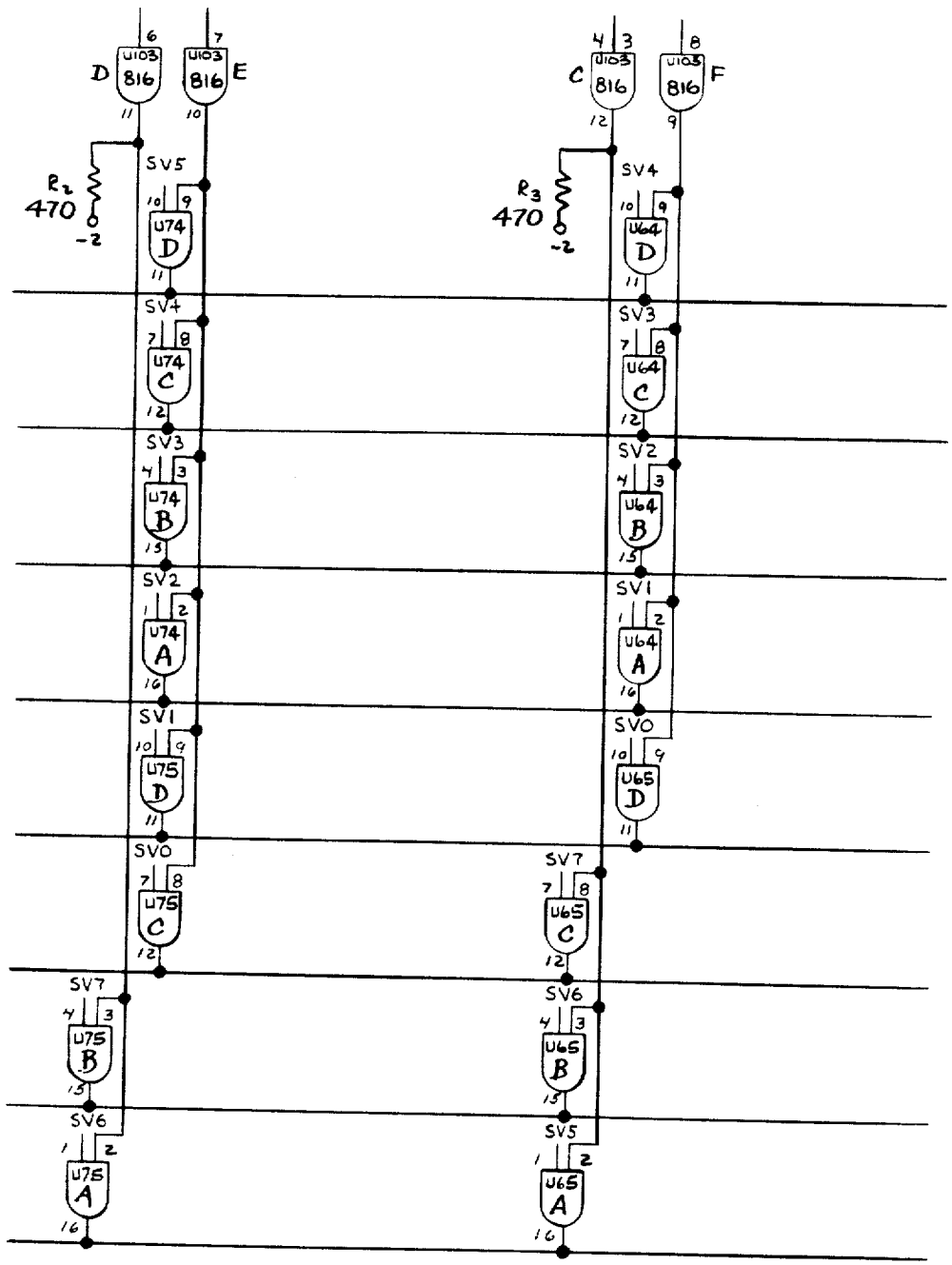

D
U103
816
E
C
F
R2
470
-2
R3
SV5
SV4
SV3
SV2
SV1
SV0
SV7
SV6
U74
U64
U75
U65
A
B
C
D

FIG.32F

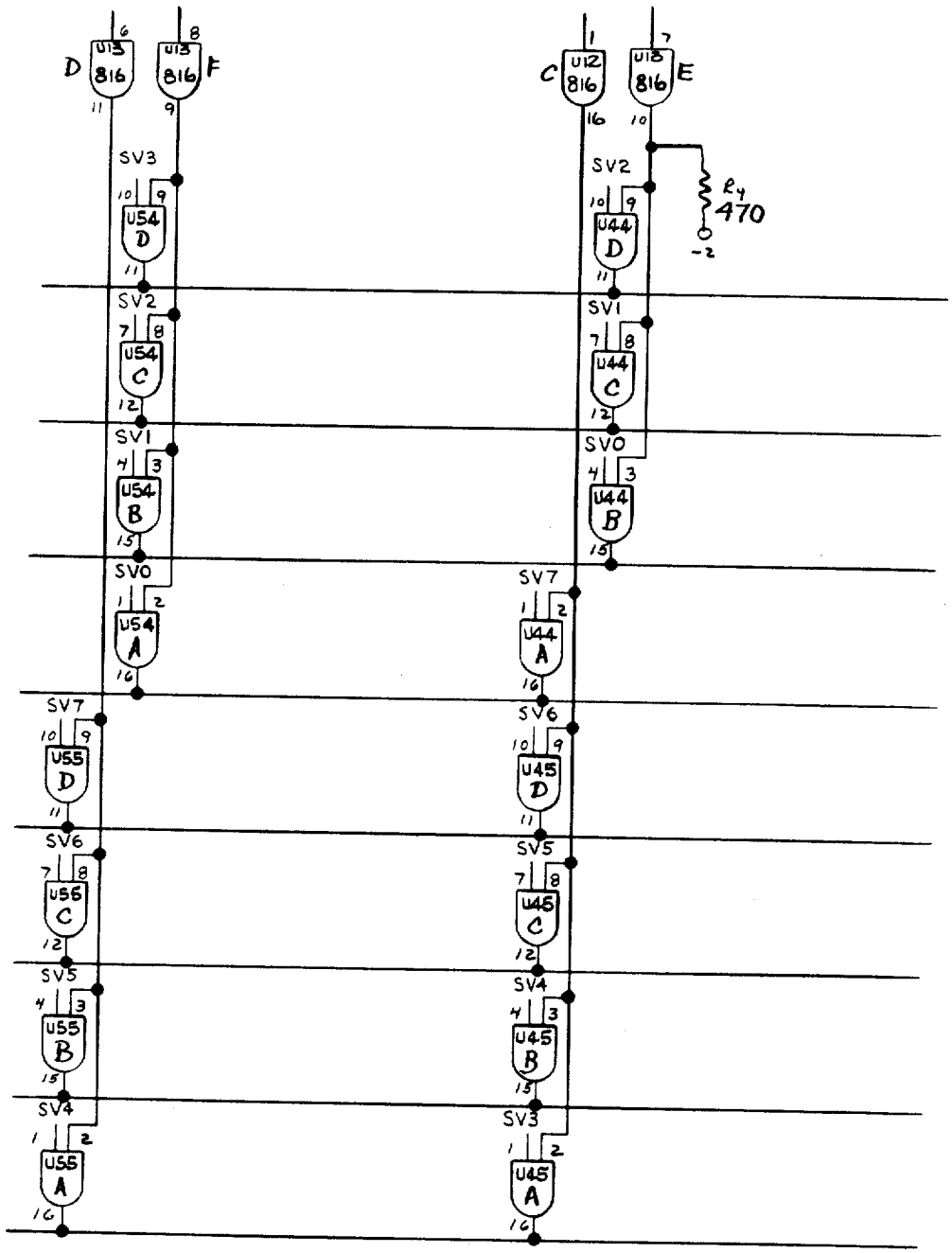

D
U13
816
F
C
U12
816
E
SV3
SV2
SV1
SV0
SV7
SV6
SV5
SV4
U54
U44
U55
U56
U45
A
B
C
D
R4
470
-2

FIG. 32G

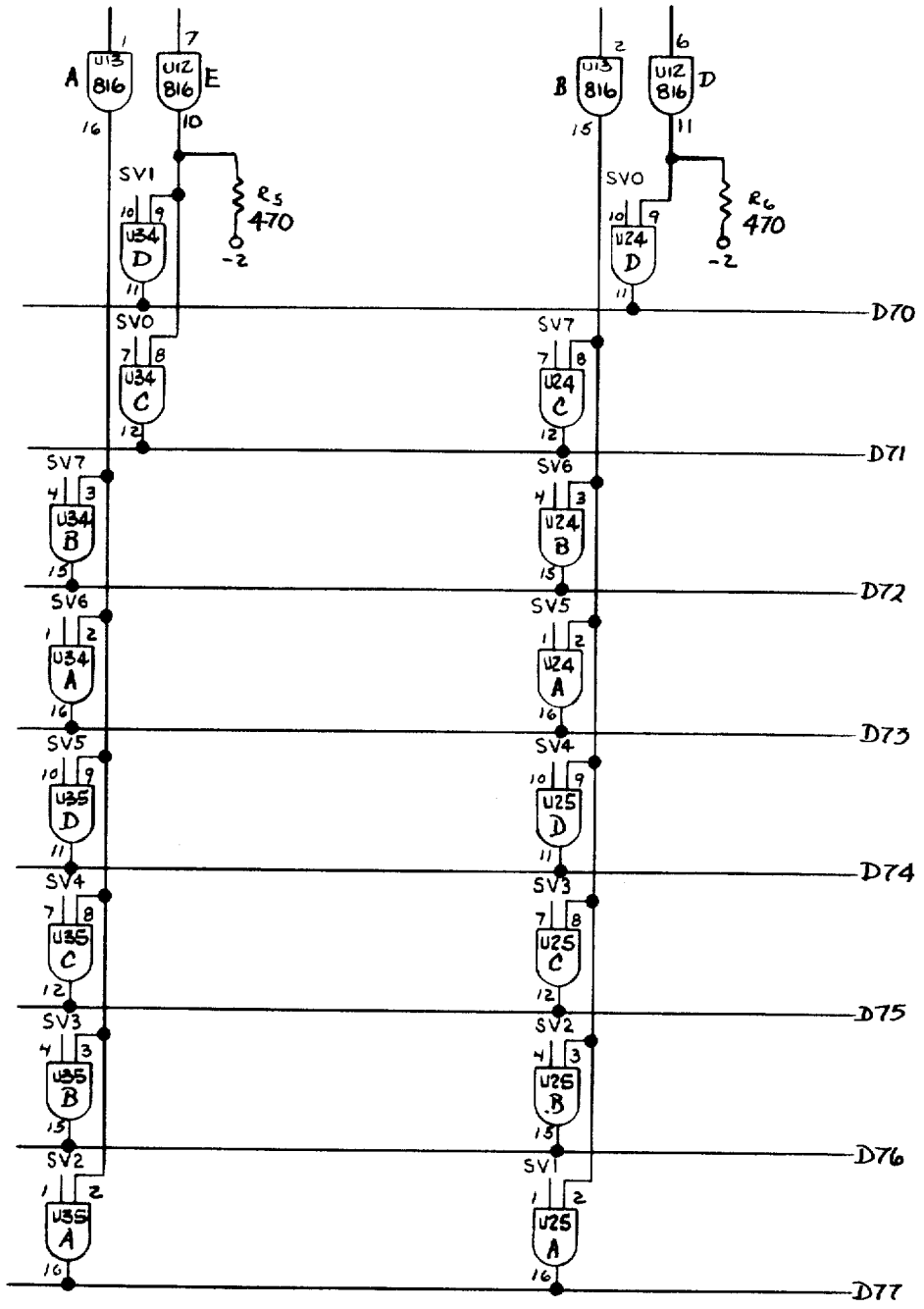

A
U13 816
E
U12 816
SV1
U34 D
R5 470
-2
SV0
U34 C
SV7
U34 B
SV6
U34 A
SV5
U35 D
SV4
U35 C
SV3
U35 B
SV2
U35 A
B
U13 816
D
U12 816
SV0
U24 D
R6 470
-2
SV7
U24 C
SV6
U24 B
SV5
U24 A
SV4
U25 D
SV3
U25 C
SV2
U26 B
SV1
U25 A
D70
D71
D72
D73
D74
D75
D76
D77

FIG. 32H

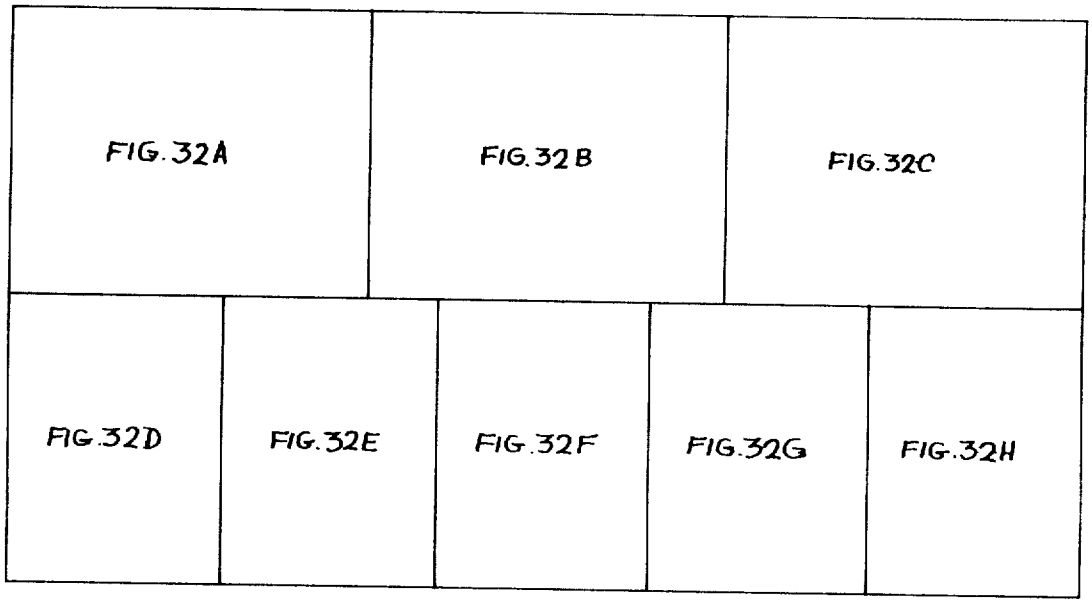



FIG. 33

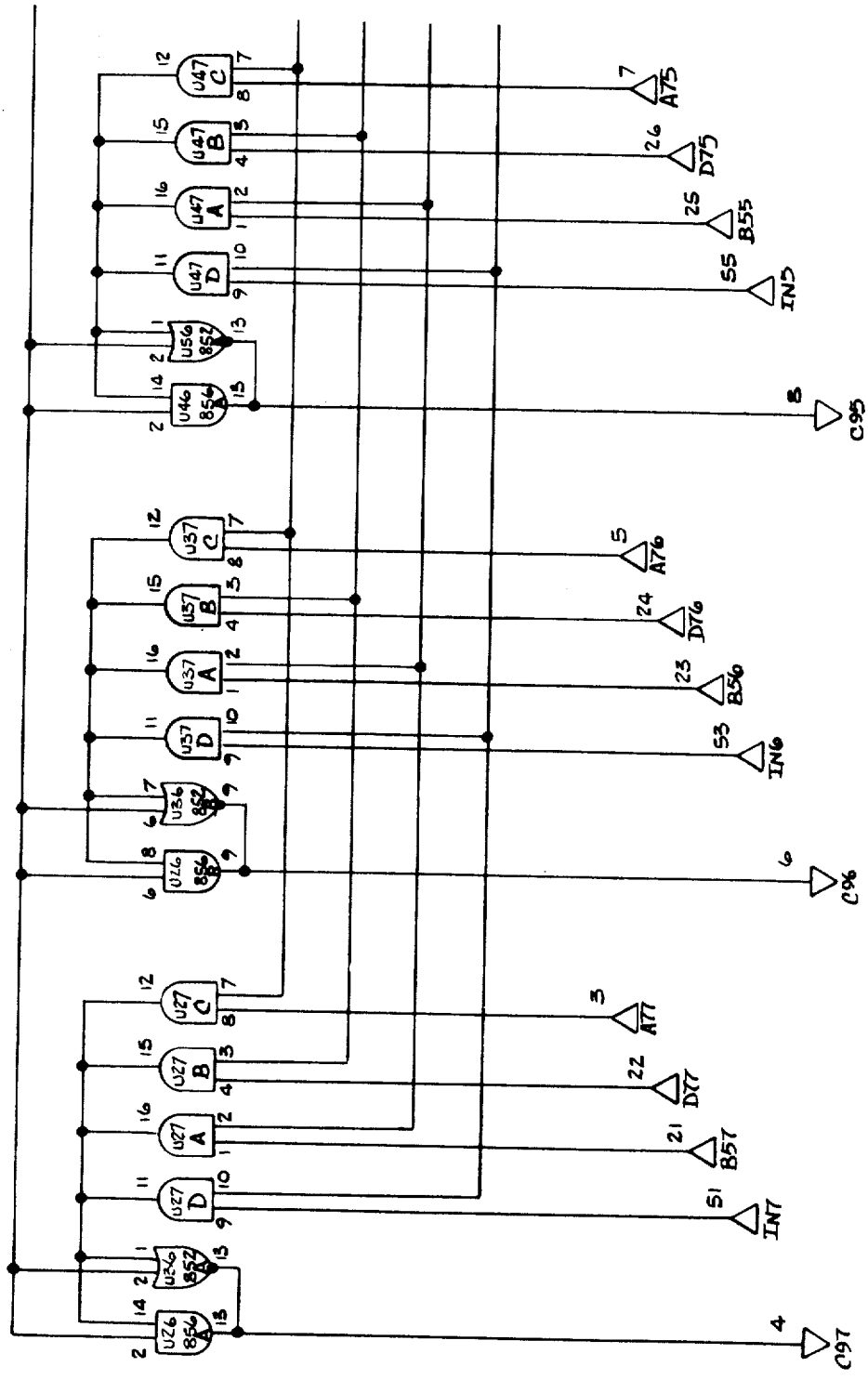

U47 C
U47 B
U47 A
U47 D
U56 852
U46 856
U37 C
U37 B
U37 A
U37 D
U36 852
U26 856
U27 C
U27 B
U27 A
U27 D
U36 852
U26 856
A75
D75
B55
IN5
C95
A76
D76
B56
IN6
C96
A77
D77
B57
INT
C97

FIG. 34A

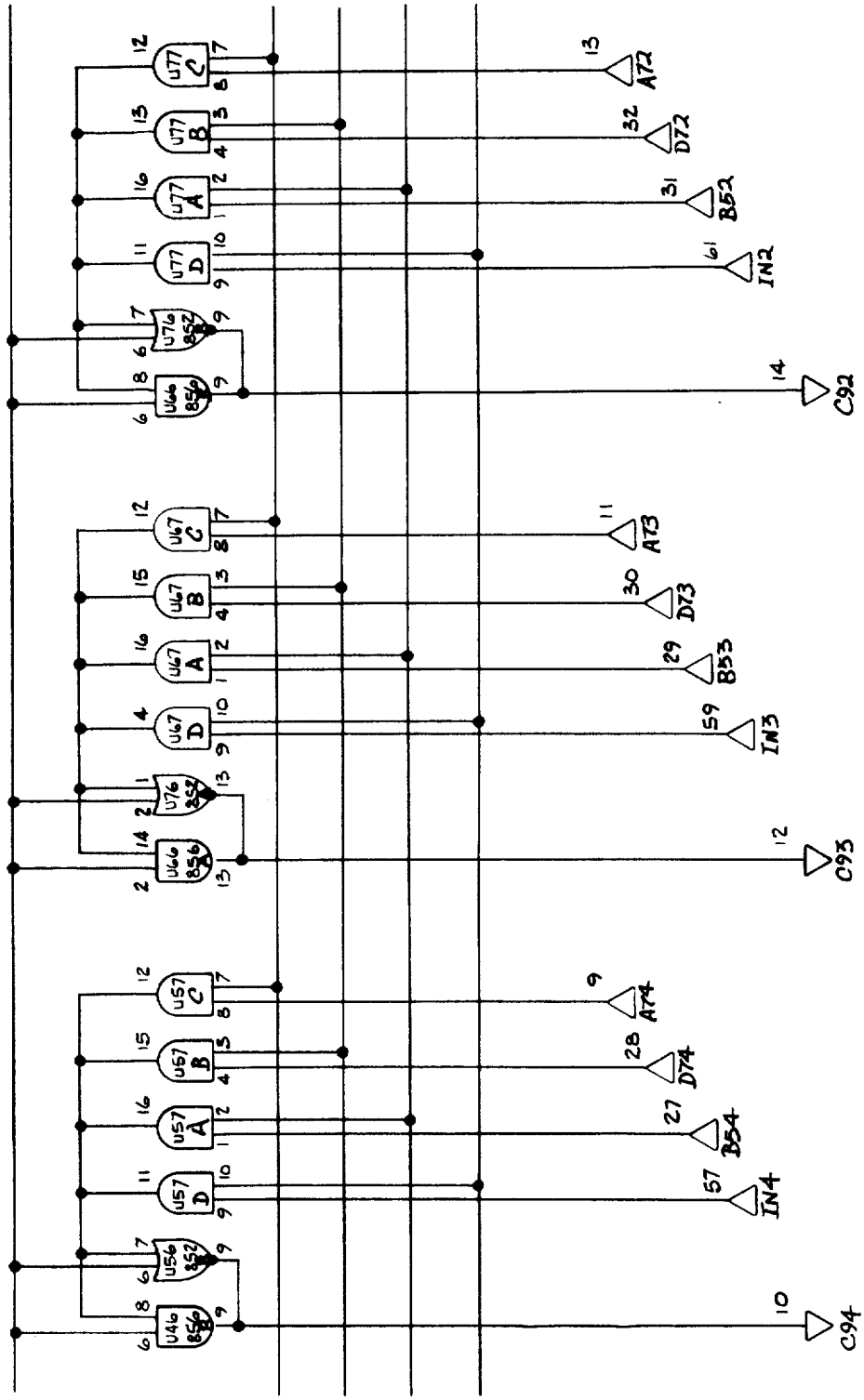

U77 C
U77 B
U77 A
U77 D
U76 852
U66 856
U67 C
U67 B
U67 A
U67 D
U76 852
U66 856
U57 C
U57 B
U57 A
U57 D
U56 852
U46 856
A72
D72
B52
IN2
C92
A73
D73
B53
IN3
C93
A74
D74
B54
IN4
C94

FIG. 34B

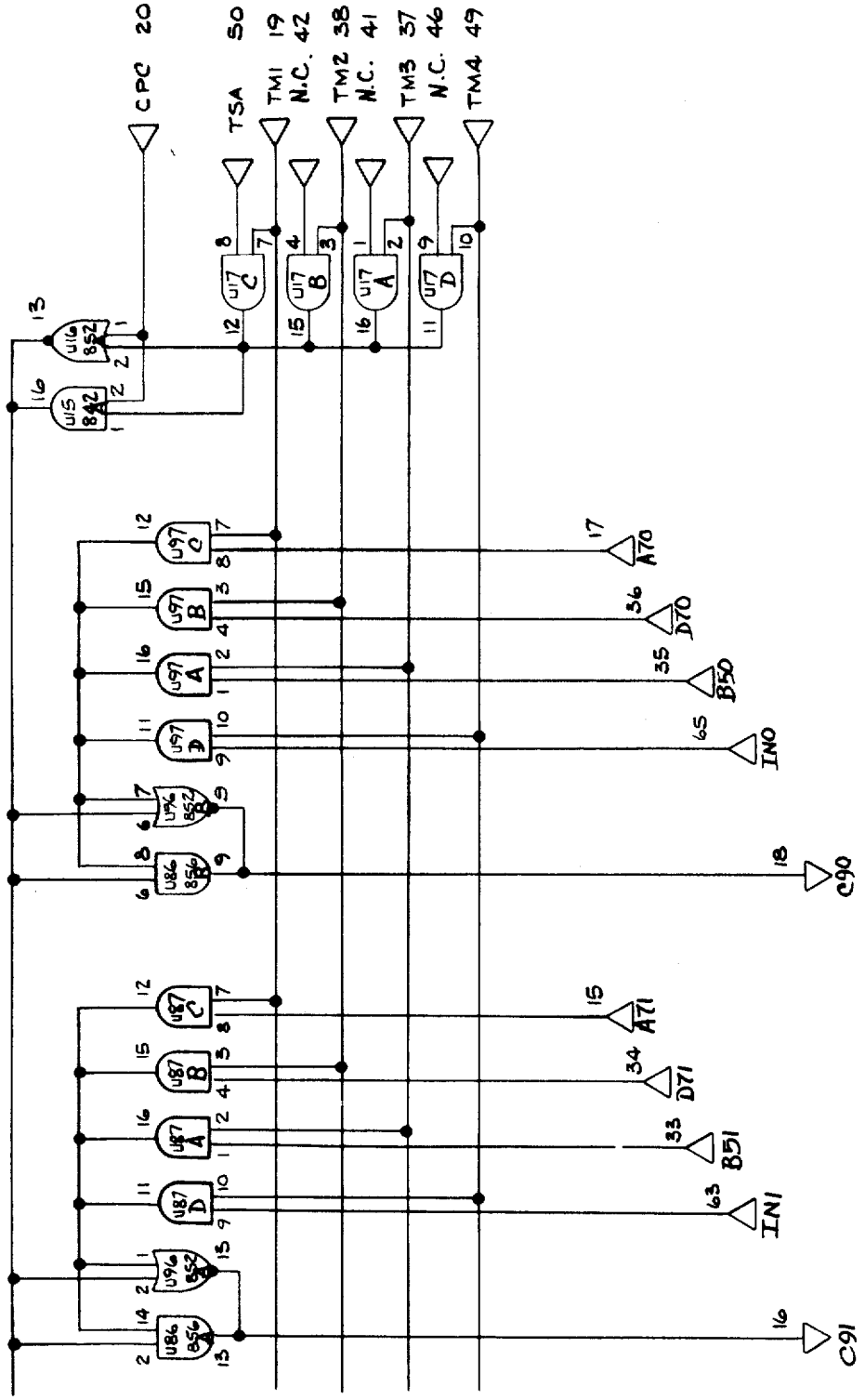

CPC 20
TSA 50
TM1 19
N.C. 42
TM2 38
N.C. 41
TM3 37
N.C. 46
TM4 49
U17 C
U17 B
U17 A
U17 D
U116 852
U115 842
U97 C
U97 B
U97 A
U97 D
U96 852
U86 856
U87 C
U87 B
U87 A
U87 D
U96 852
U86 856
17 A70
36 D70
35 B50
65 IN0
18 C90
15 A71
34 D71
33 B51
63 IN1
16 C91

FIG. 34C

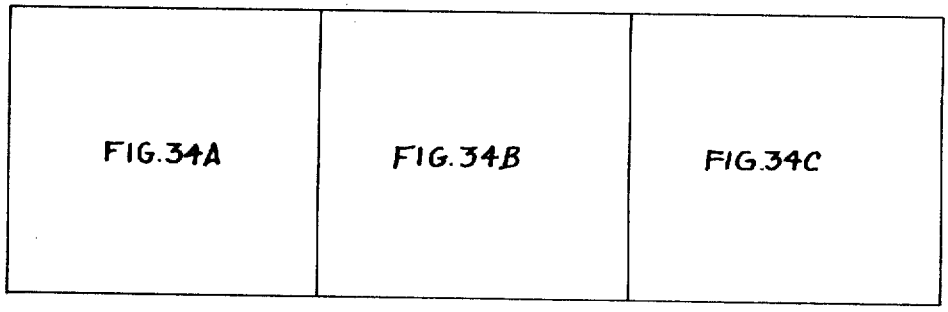



FIG.35

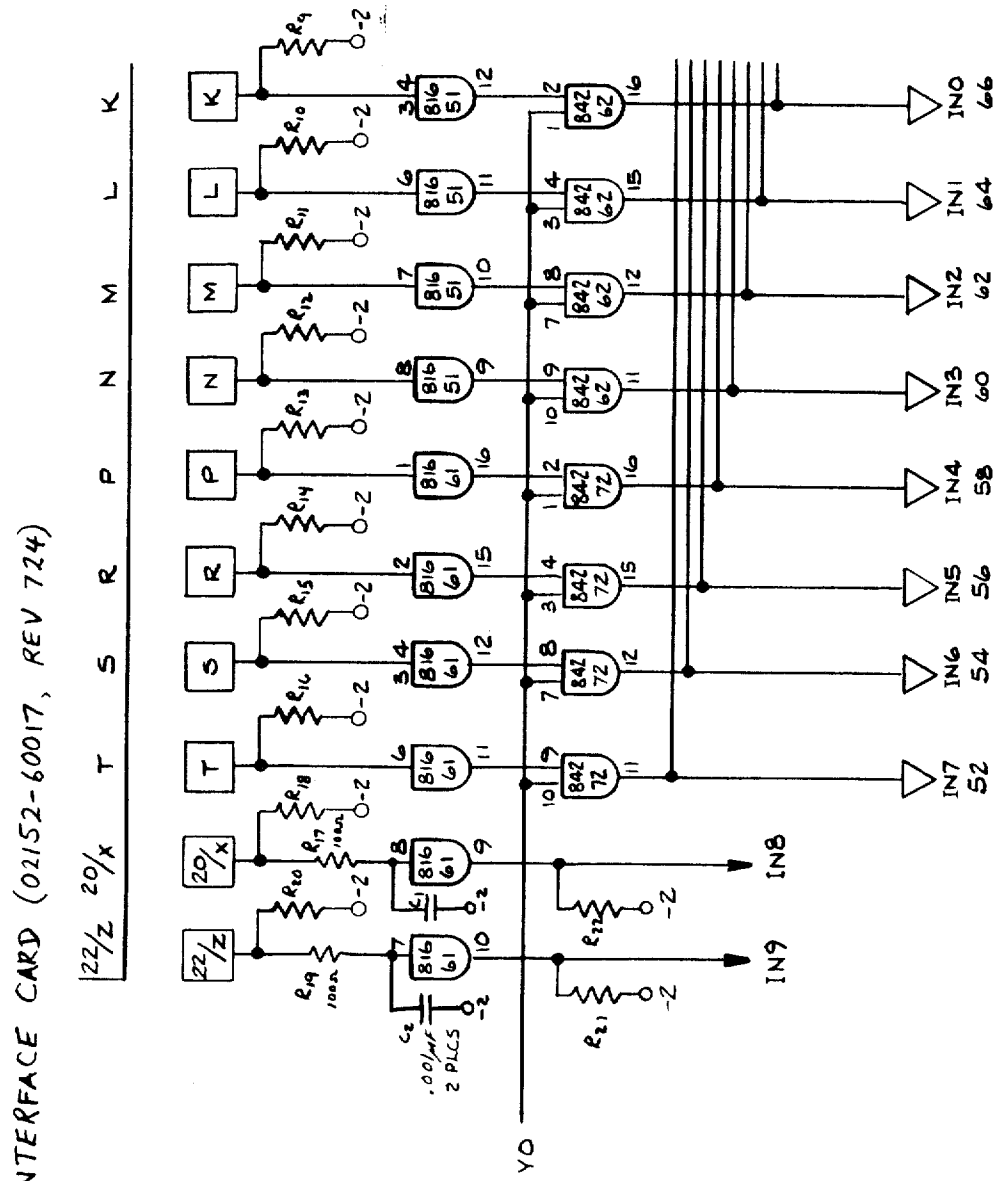

FPP INTERFACE CARD (02152-60017, REV 724)
22/Z 20/X T S R P N M L K
22/Z
20/X
K
L
M
N
P
R
S
T
816 51
816 61
842 62
842 72
IN0 66
IN1 64
IN2 62
IN3 60
IN4 58
IN5 56
IN6 54
IN7 52
IN8
IN9
YO

FIG. 36A

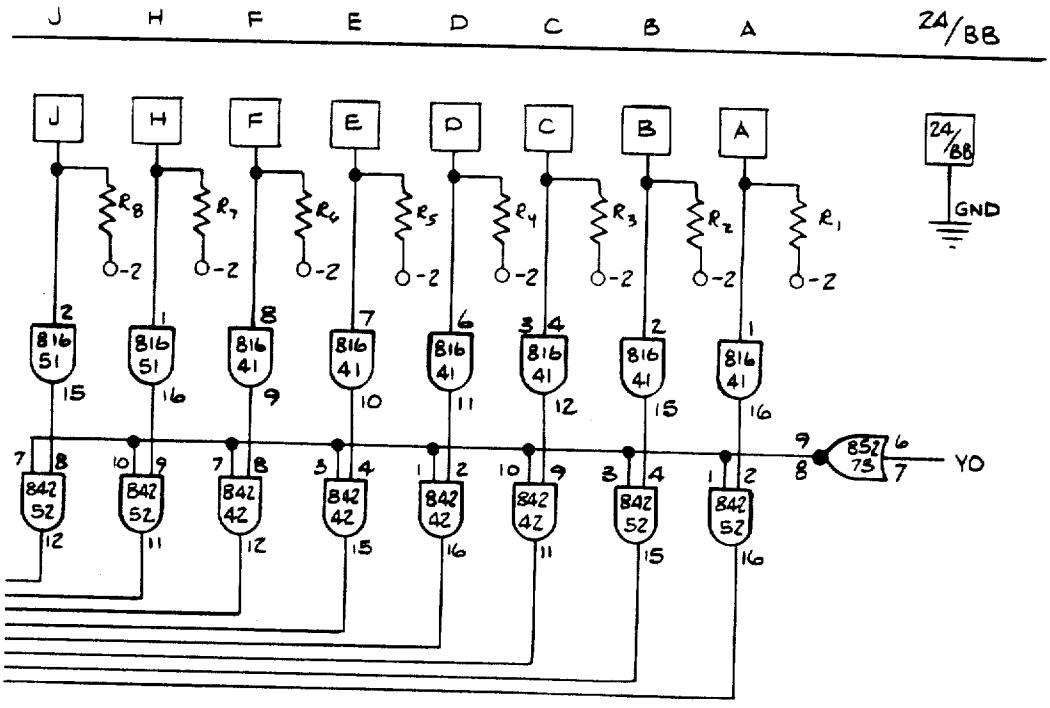



FIG. 36B

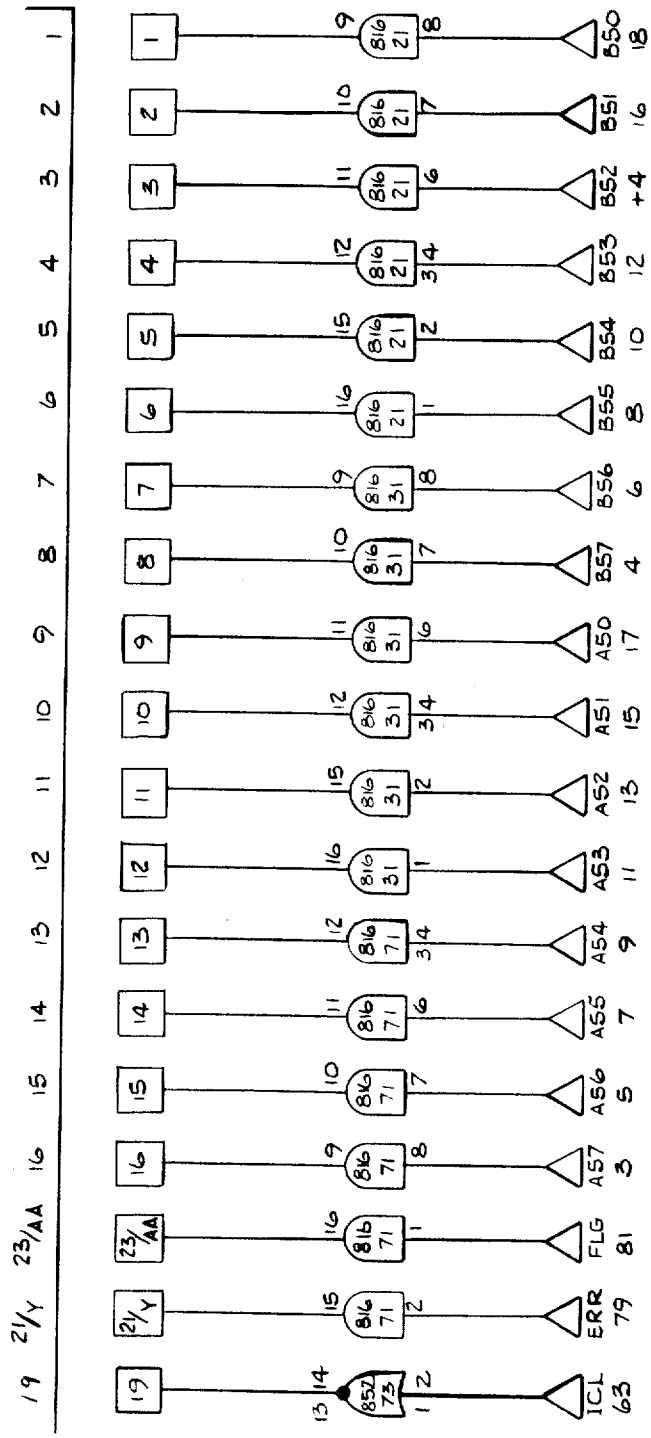
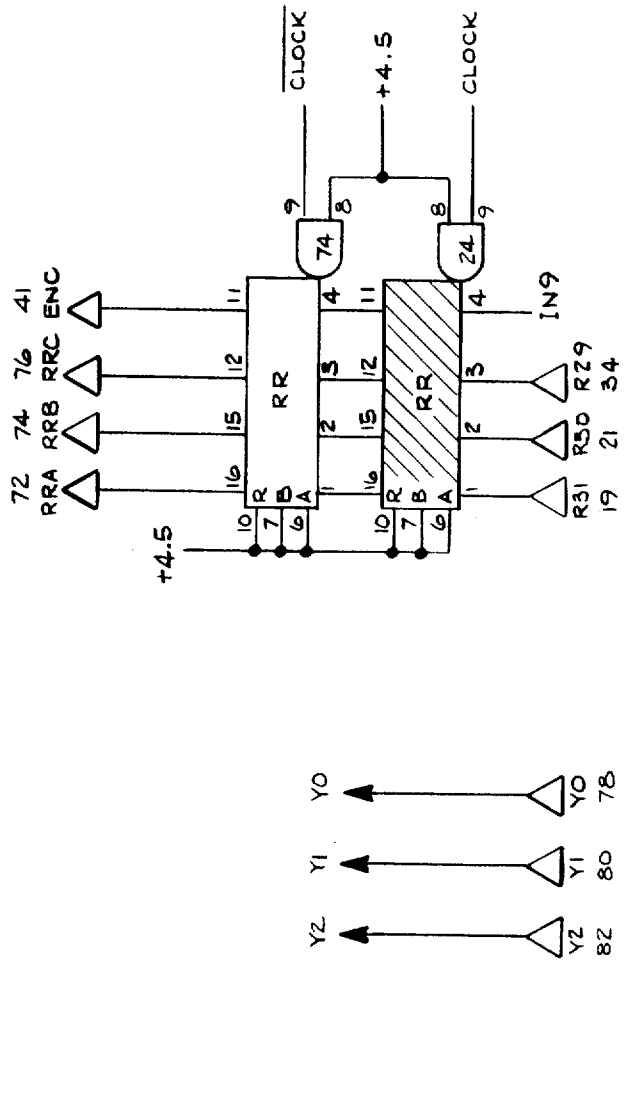

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 23/AA 21/Y 19
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 23/AA 21/Y 19
816 21
816 31
816 71
852 73
B50 B51 B52 B53 B54 B55 B56 B57 A50 A51 A52 A53 A54 A55 A56 A57 FLG ERR ICL
18 16 +4 12 10 8 6 4 17 15 13 11 9 7 5 3 81 79 63

FIG.36C




NOTE

1. ALL RESISTORS RATED 1/4 WATT, 470Ω, ±5% EXCEPT AS NOTED
2. ALL QUAD LATCHES HAVE EXTERNAL PULL DOWN RESISTOR ON OUTPUT

FIG. 36D

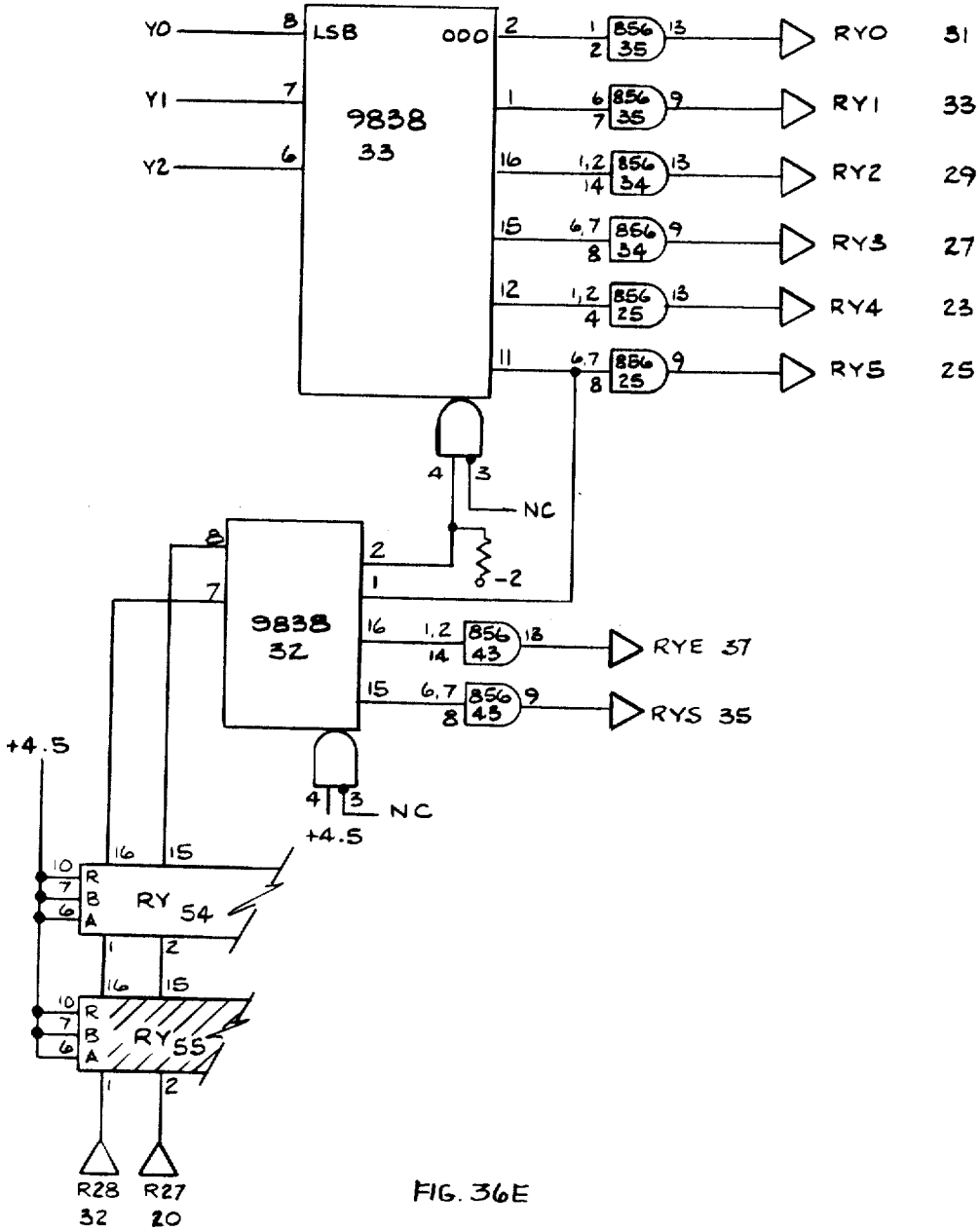

Y0
Y1
Y2
LSB
000
9838
33
RY0 31
RY1 33
RY2 29
RY3 27
RY4 23
RY5 25
856 35
856 34
856 25
NC
-2
9838
32
RYE 37
RYS 35
856 43
856 45
+4.5
RY 54
RY 55
R
B
A
R28
32
R27
20

FIG. 36E

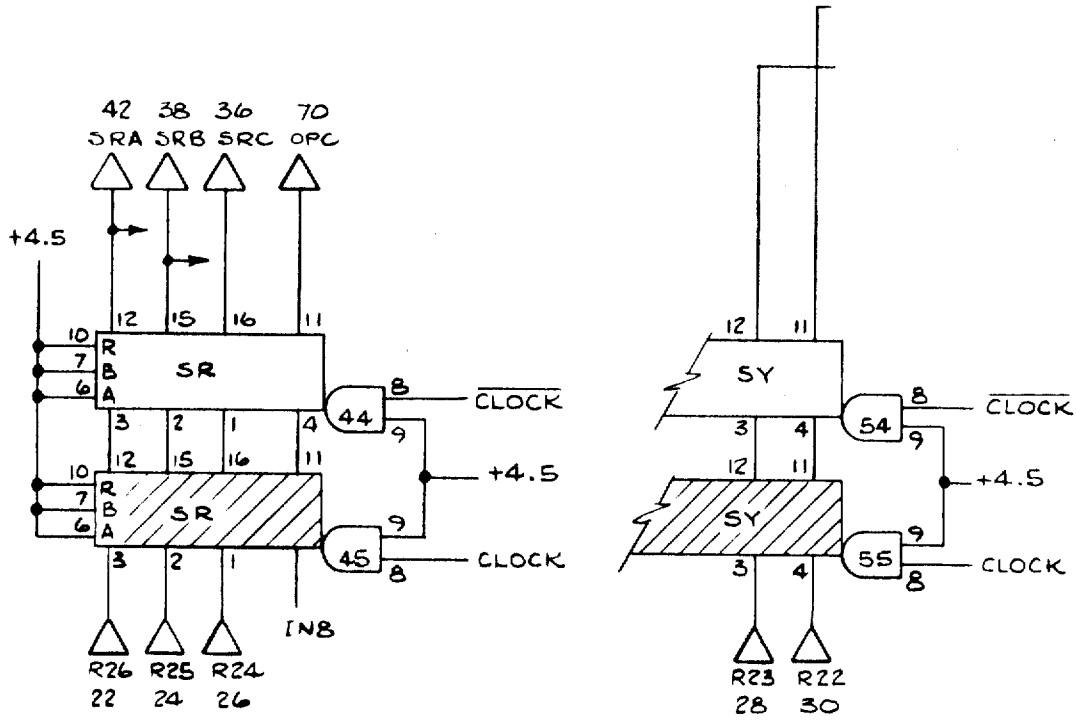

42 SRA
38 SRB
36 SRC
70 OPC
+4.5
12
15
16
11
10
7
6
R
B
A
SR
8
44
9
CLOCK
3
2
1
4
+4.5
45
CLOCK
IN8
R26
22
R25
24
R24
26
SY
54
55
R23
28
R22
30

FIG. 36F

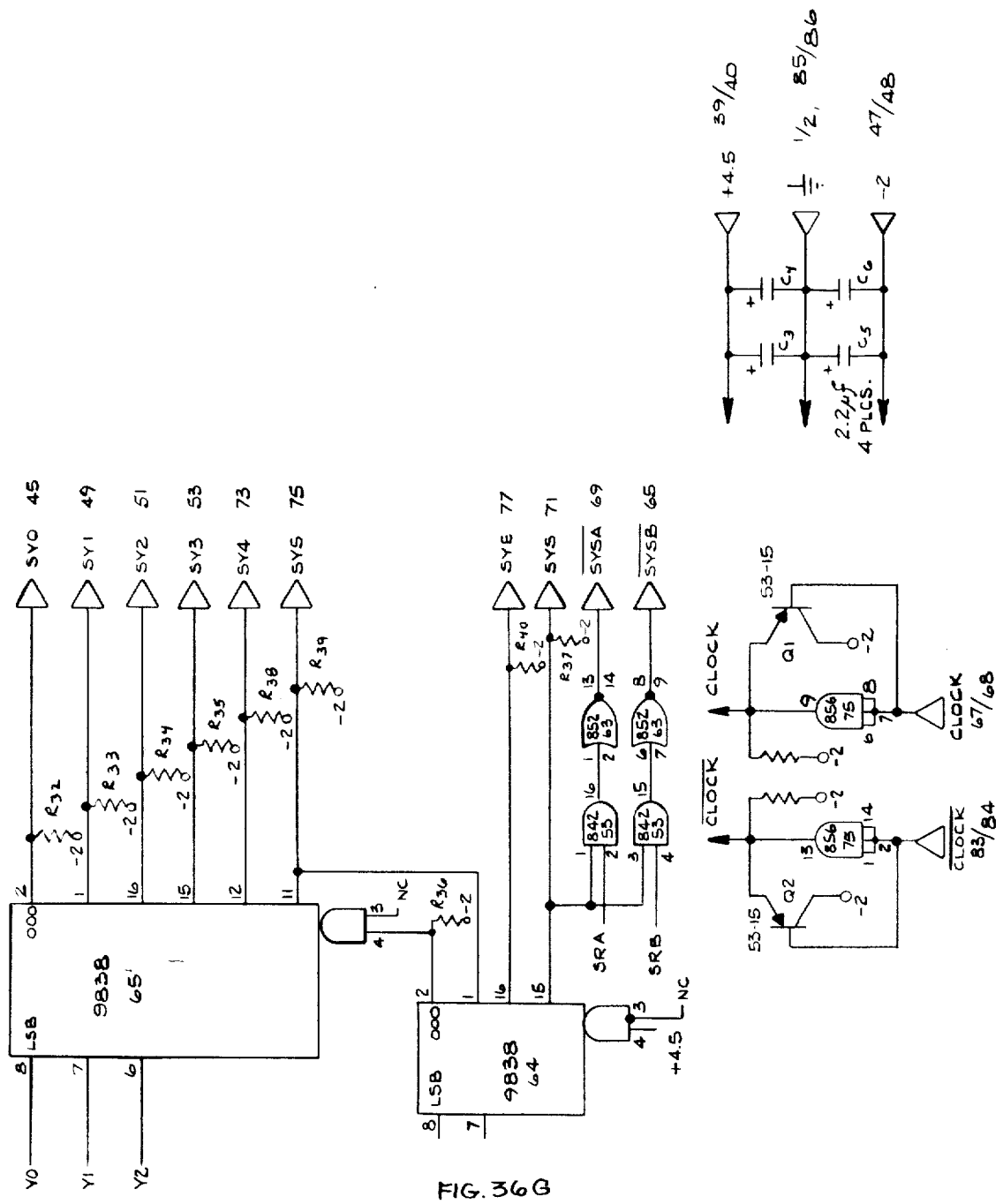

SY0
SY1
SY2
SY3
SY4
SY5
SYE
SYS
SYSA
SYSB
9838
65
64
LSB
NC
SRA
SRB
+4.5
CLOCK
Q1
Q2
Y0
Y1
Y2

FIG. 36B

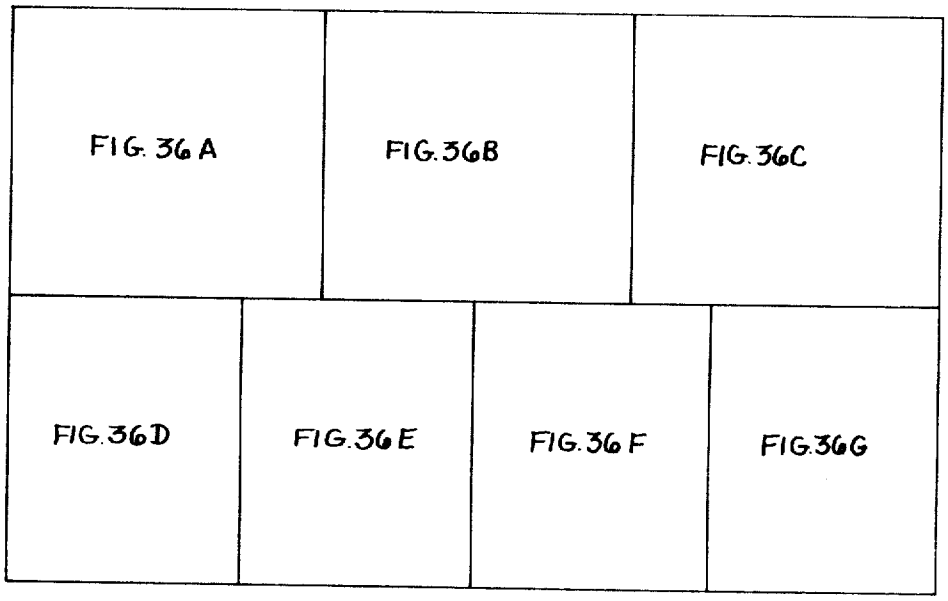



FIG. 37

0 0 0 = L0
0 0 1 = L1
0 1 0 = L2
0 1 1 = L3
1 0 0 = EX1
1 0 1 = EXM
1 1 0 = EXB
1 1 1 = HLT

1 = ROL

TEST CARD (02152-60008, REV 725)

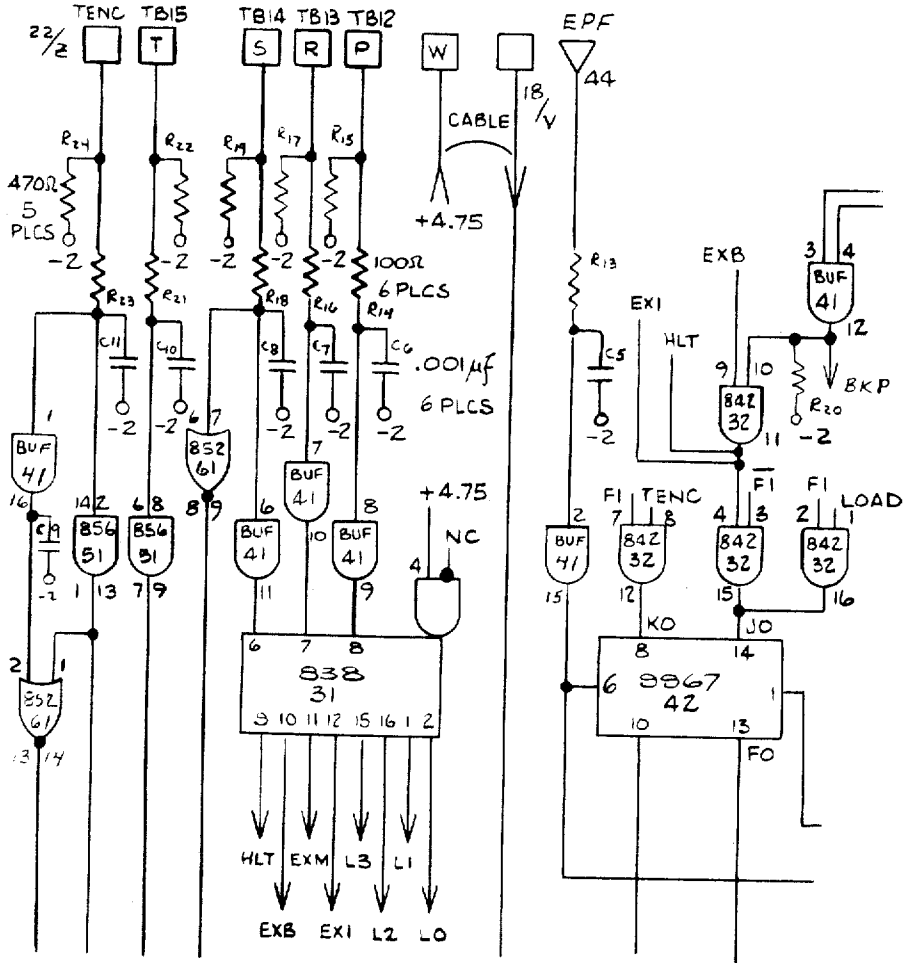



FIG. 38A

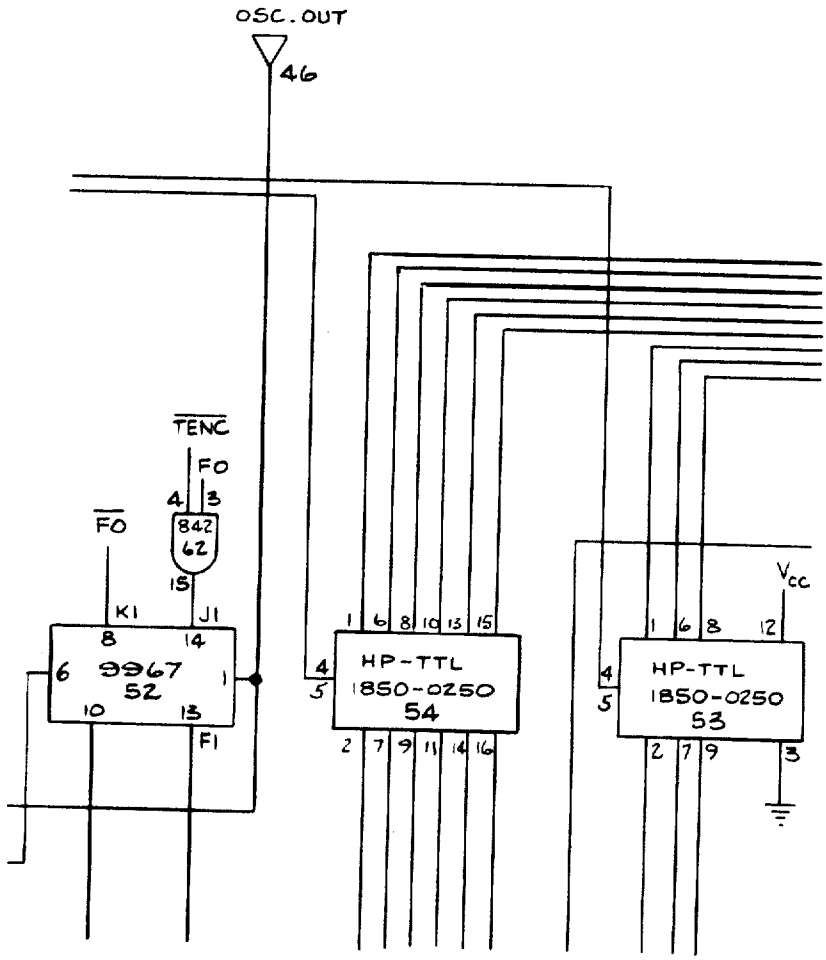

OSC. OUT
46
TENC
FO
4
3
842
62
15
FO
K1
J1
8
14
6
9967
52
10
13
1
F1
4
5
HP-TTL
1850-0250
54
1 6 8 10 13 15
2 7 9 11 14 16
Vcc
1 6 8 12
4
5
HP-TTL
1850-0250
53
2 7 9
3

FIG. 38B

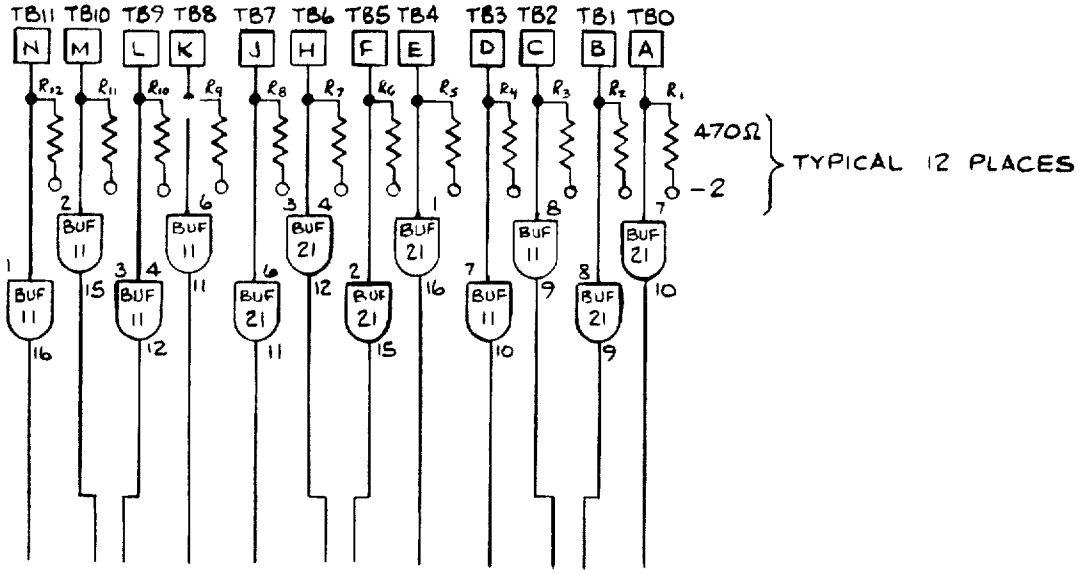

TB11 TB10 TB9 TB8 TB7 TB6 TB5 TB4 TB3 TB2 TB1 TB0
N M L K J H F E D C B A
R12 R11 R10 R9 R8 R7 R6 R5 R4 R3 R2 R1
470Ω
-2
TYPICAL 12 PLACES
BUF 11
BUF 21

FIG. 38C

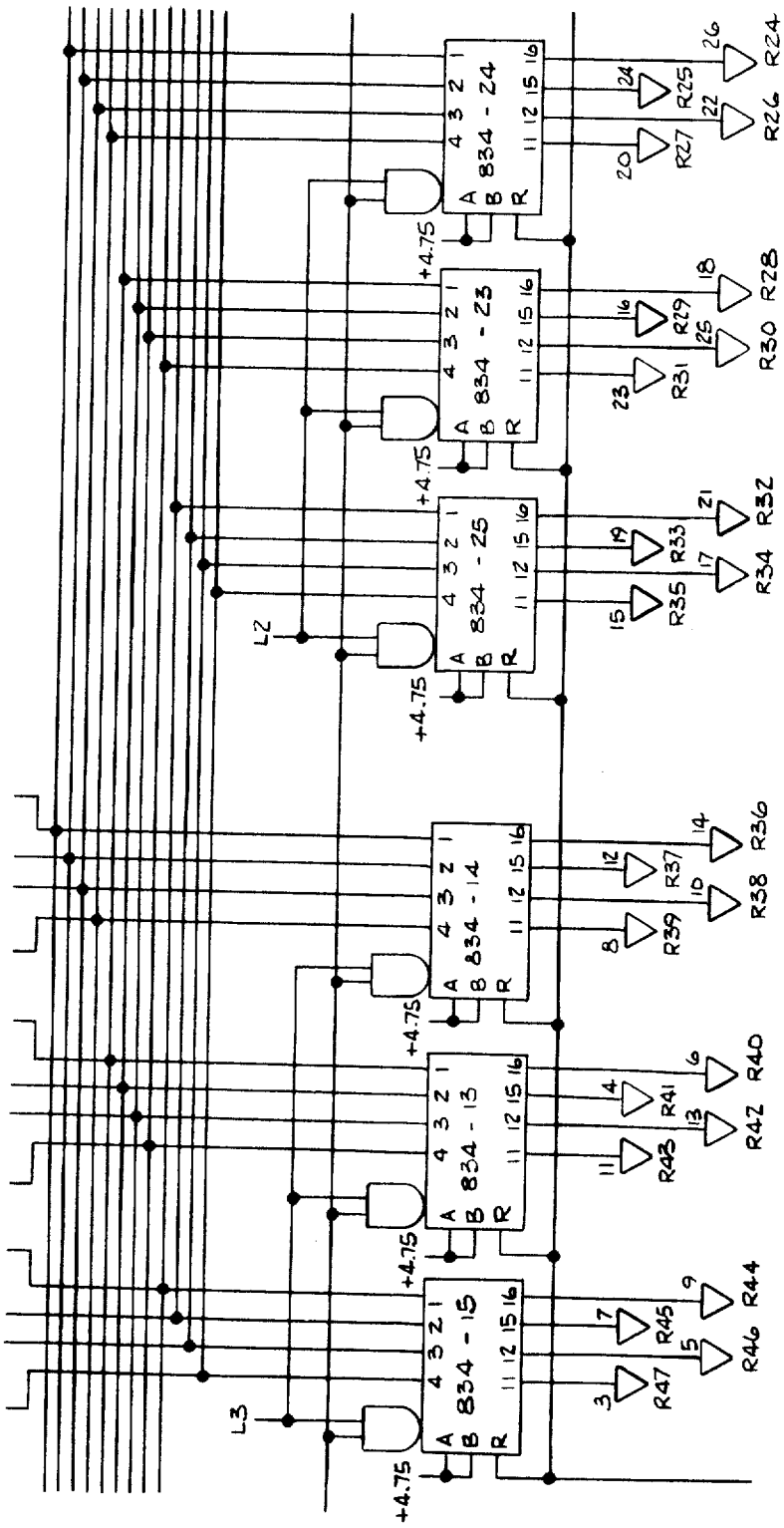

834 - Z4
834 - Z3
834 - 25
834 - 14
834 - 13
834 - 15
+4.75
L2
L3
R24
R25
R26
R27
R28
R29
R30
R31
R32
R33
R34
R35
R36
R37
R38
R39
R40
R41
R42
R43
R44
R45
R46
R47

FIG. 38D

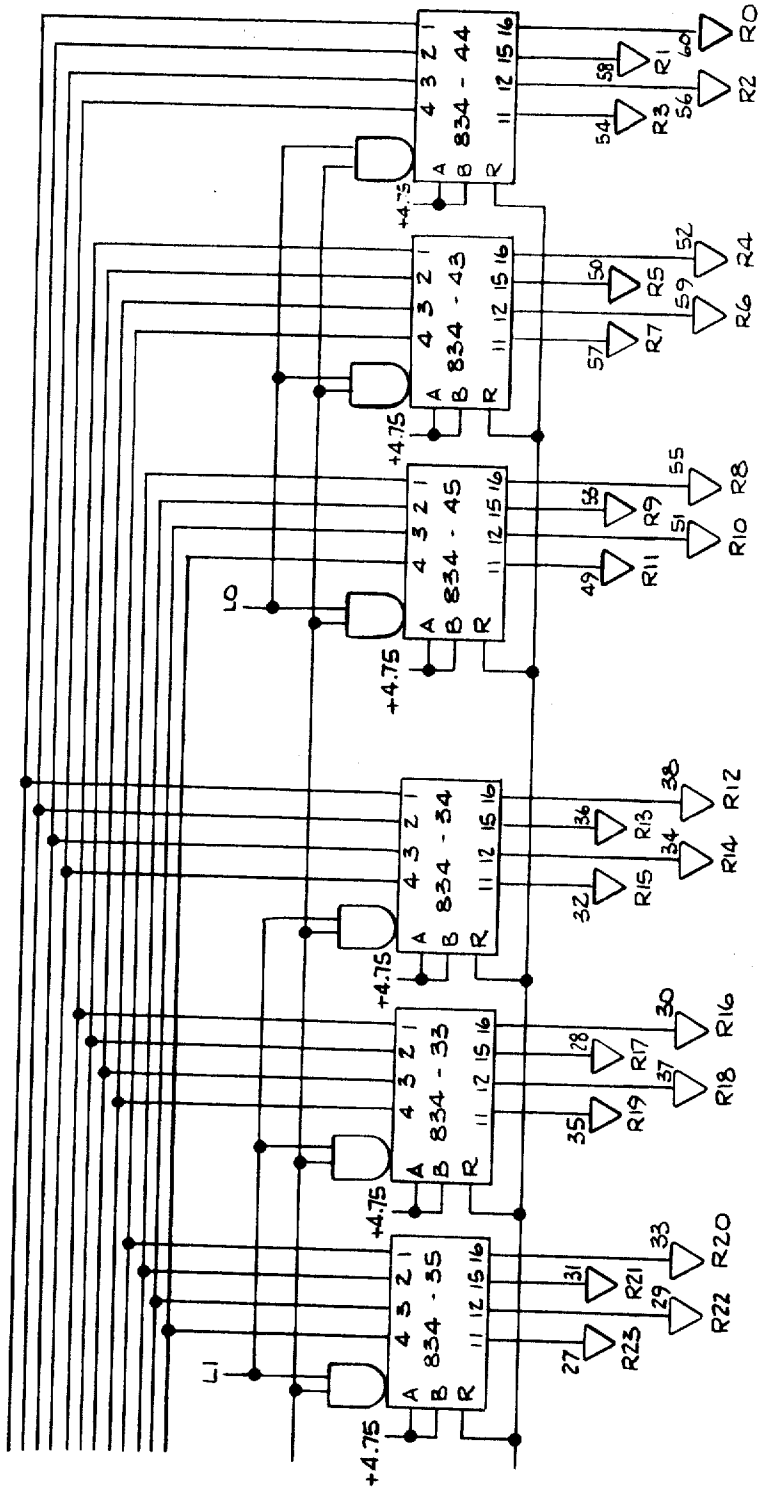
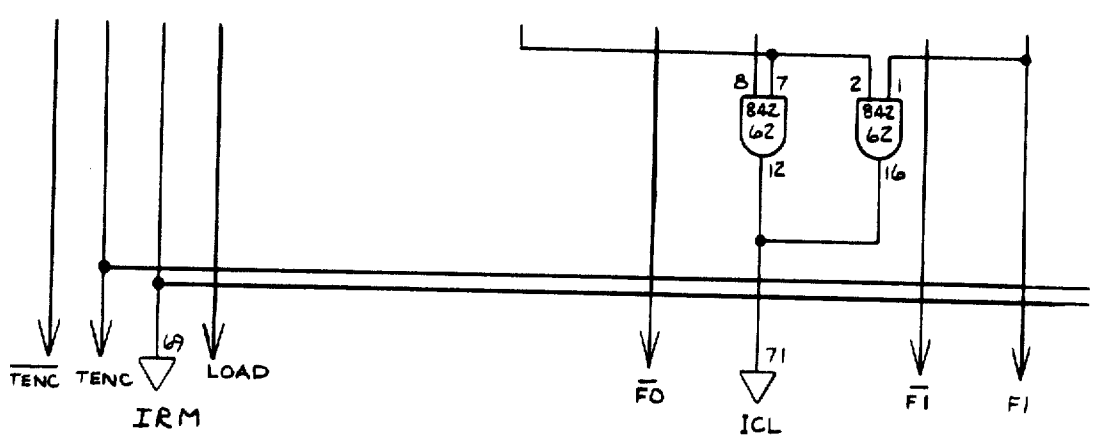

834 - 44
834 - 43
834 - 45
834 - 34
834 - 33
834 - 35
A
B
R
+4.75
L0
L1
R0
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R15
R16
R17
R18
R19
R20
R21
R22
R23

FIG. 38E




NOTE

1. ☐ DENOTES 48 PIN CONNECTOR
2. △ DENOTES 86 PIN CONNECTOR
3. ALL QUAD LATCHES HAVE AN EXTERNAL PULL DOWN RESISTOR ADJACENT TO OUTPUT

FIG. 38F

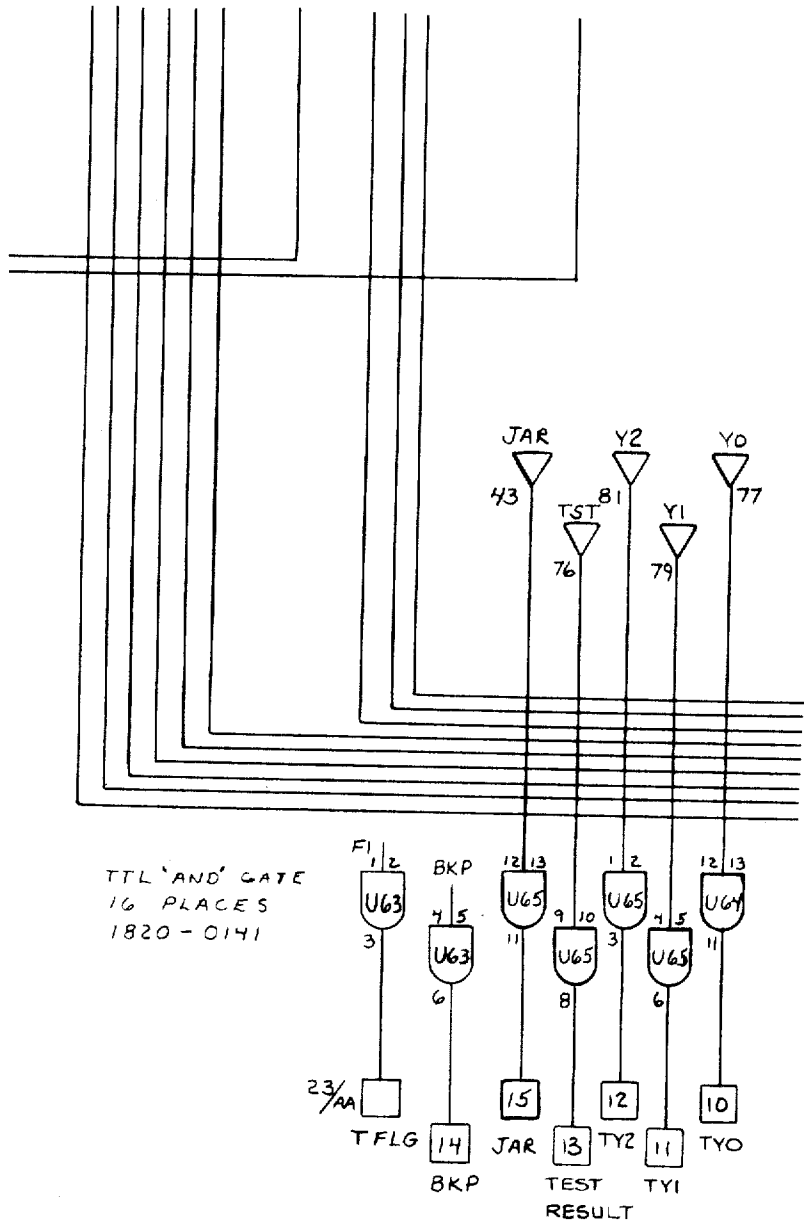

JAR
43
TST
76
Y2
81
Y1
79
YO
77
TTL 'AND' GATE
16 PLACES
1820-0141
F1
BKP
U63
U65
U64
23/AA
TFLG
14
BKP
15
JAR
13
TEST RESULT
12
TY2
11
TY1
10
TYO

FIG. 38 G

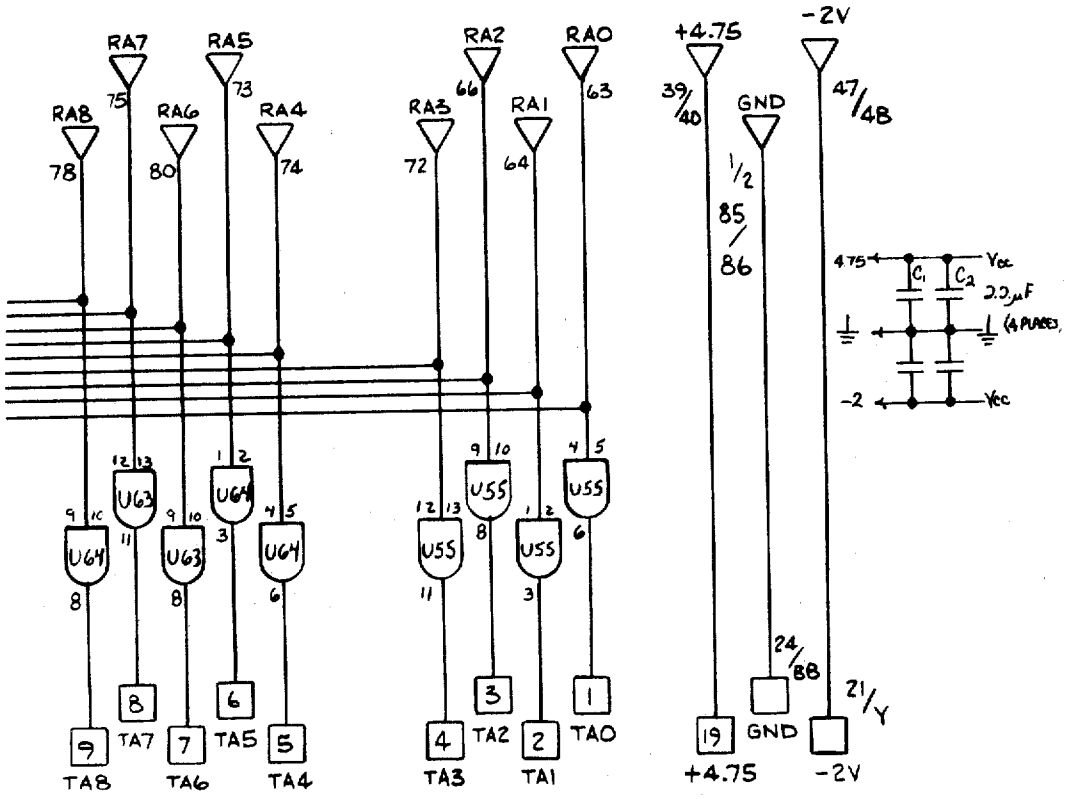
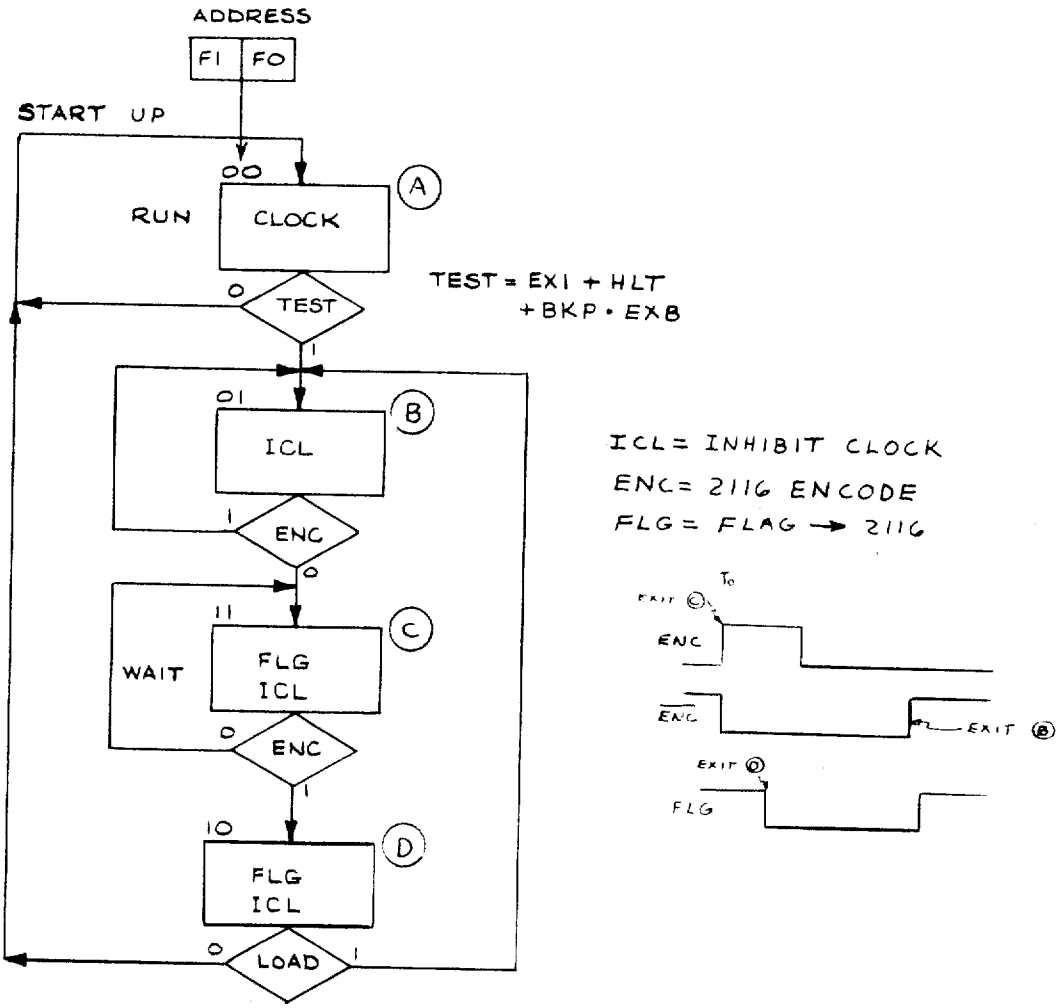

RA7
RA5
RA8
RA6
RA4
RA2
RA0
RA3
RA1
+4.75
-2V
GND
U64
U63
U55
TA8
TA7
TA6
TA5
TA4
TA3
TA2
TA1
TA0
+4.75
GND
-2V
Vcc
C1
C2
2.2 μF
(4 PLACES)

FIG. 38H




FLOW CHART FOR TEST BOARD CONTROLLER

FIG. 38I

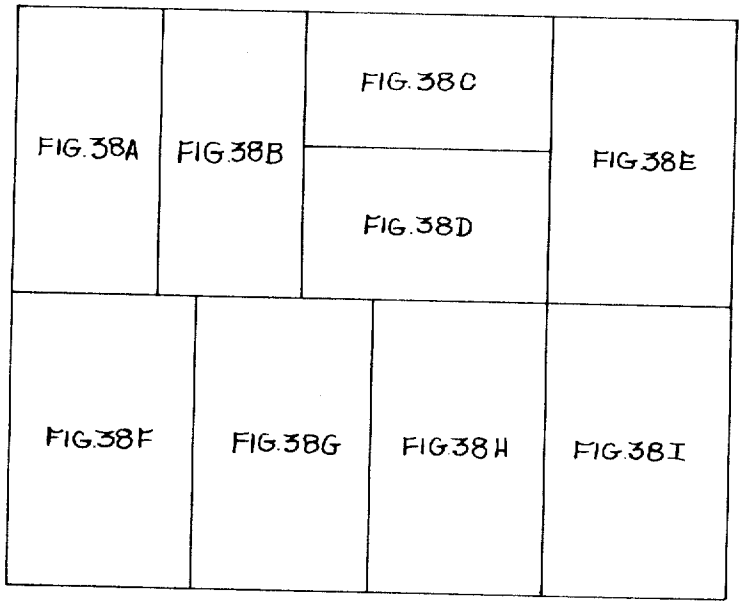


FIG. 38A
FIG. 38B
FIG. 38C
FIG. 38D
FIG. 38E
FIG. 38F
FIG. 38G
FIG. 38H
FIG. 38I

FIG. 39

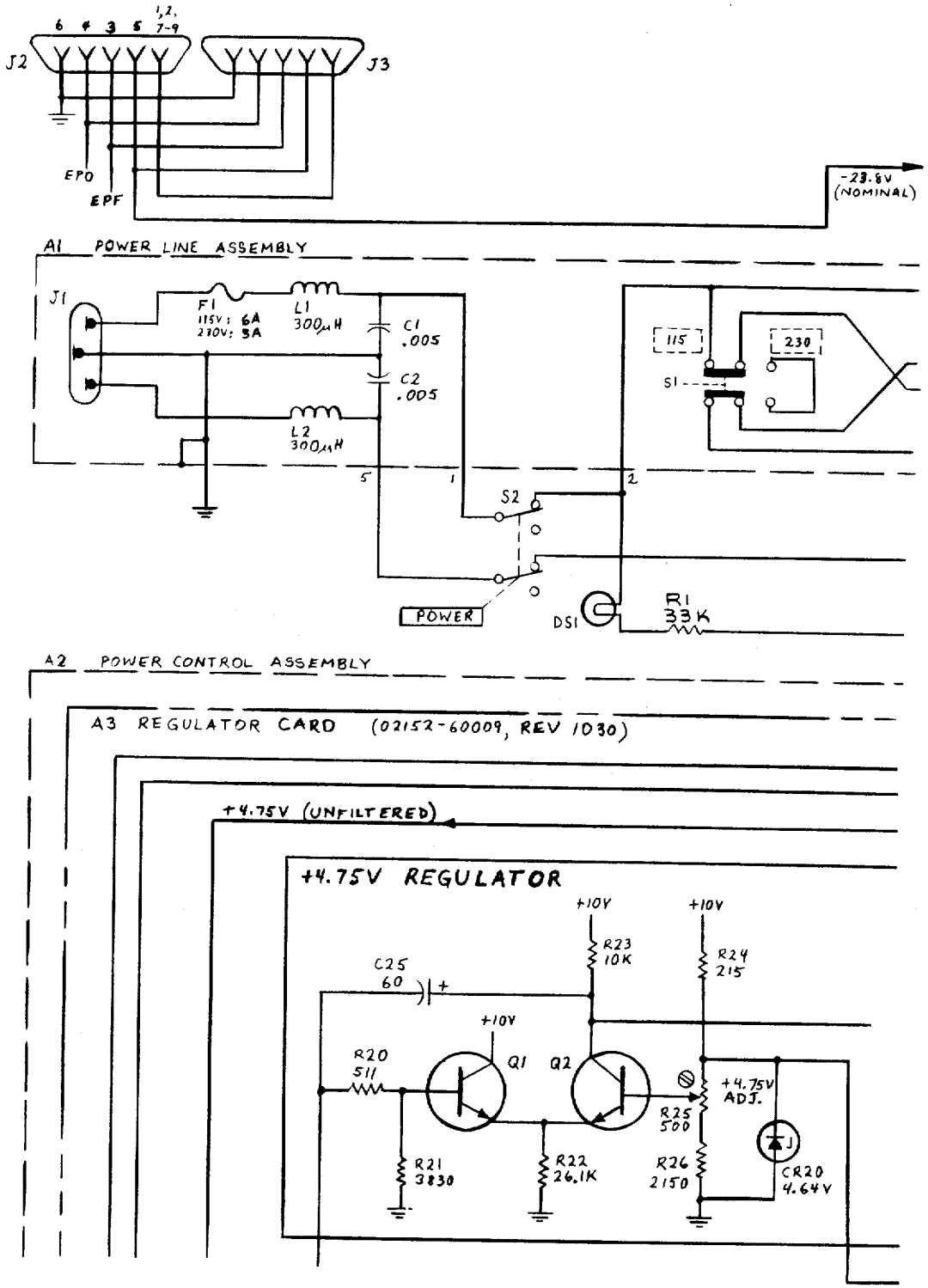

J2
6 4 3 5 1,2, 7-9
J3
EPO
EPF
-23.8V (NOMINAL)
A1 POWER LINE ASSEMBLY
J1
F1
115V: 6A
230V: 5A
L1
300μH
C1
.005
C2
.005
L2
300μH
115
230
S1
5
1
2
S2
POWER
DS1
R1
33K
A2 POWER CONTROL ASSEMBLY
A3 REGULATOR CARD (02152-60009, REV 1030)
+4.75V (UNFILTERED)
+4.75V REGULATOR
+10V
+10V
R23
10K
R24
215
C25
60
+10V
R20
511
Q1
Q2
+4.75V
ADJ.
R25
500
R21
3830
R22
26.1K
R26
2150
CR20
4.64V

FIG. 40A

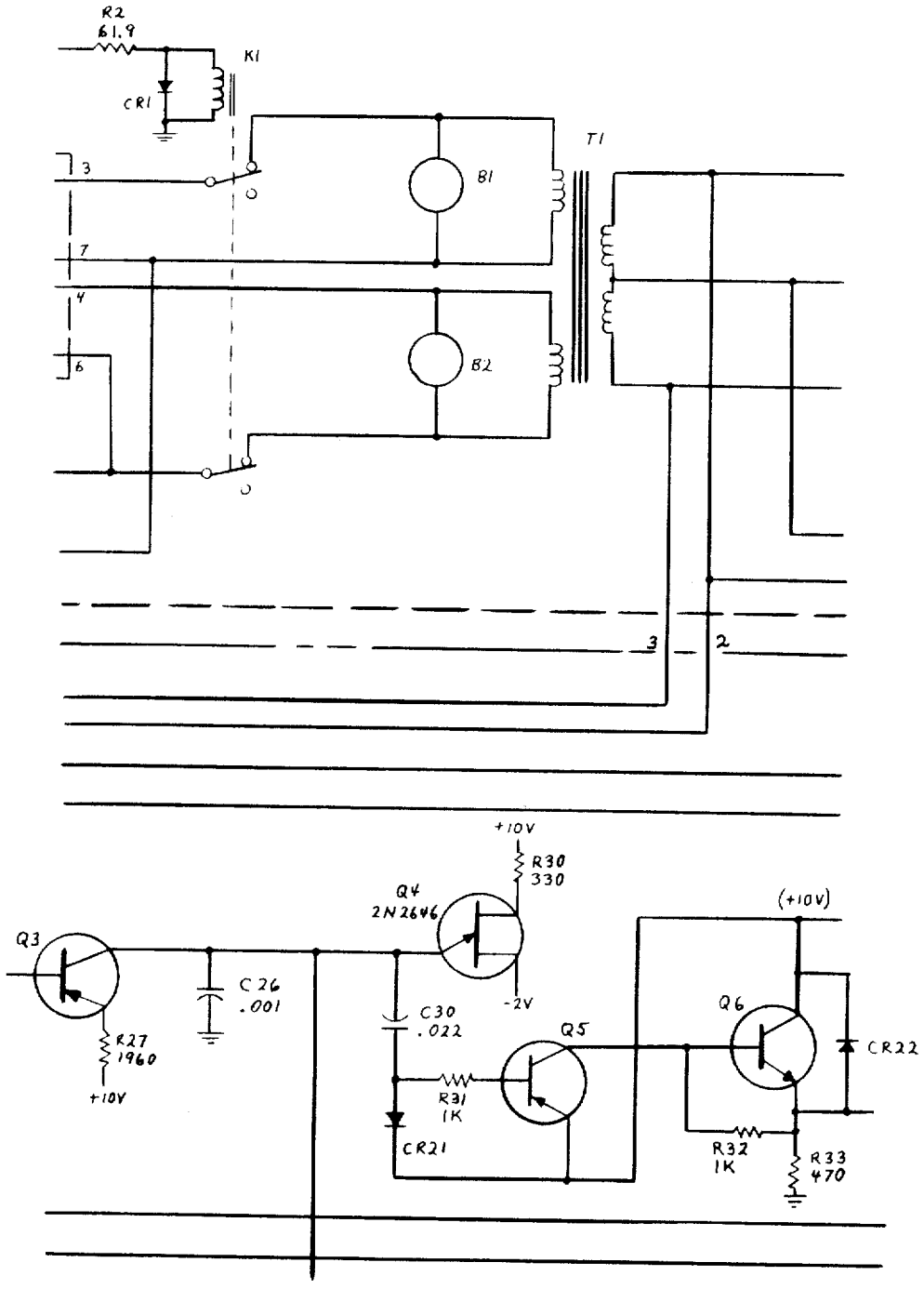

R2
61.9
K1
CR1
T1
B1
B2
3
7
4
6
3
2
+10V
R30
330
Q4
2N2646
-2V
Q3
C26
.001
C30
.022
Q5
Q6
(+10V)
CR22
R27
1960
+10V
R31
1K
CR21
R32
1K
R33
470

FIG. 40B

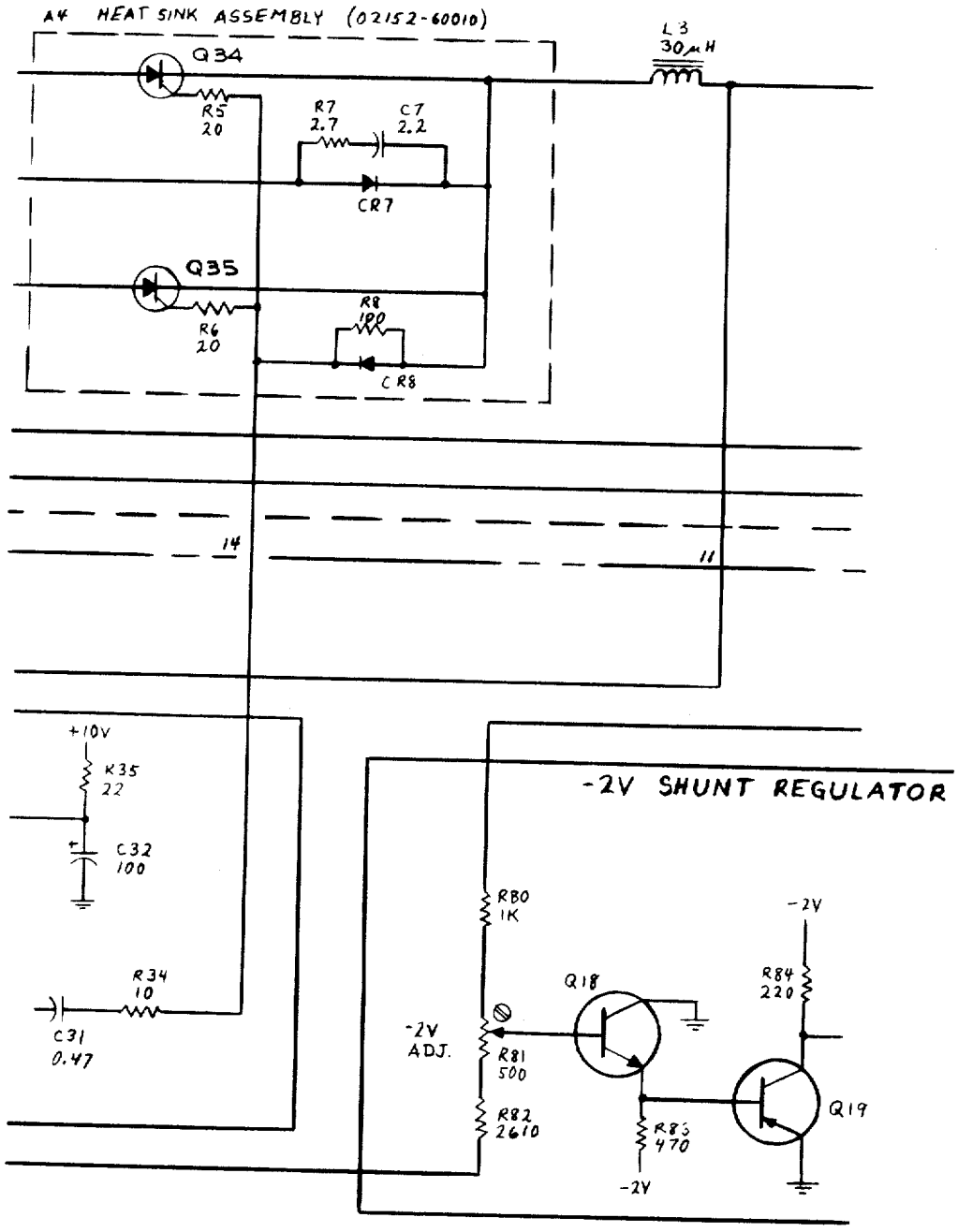

A4 HEAT SINK ASSEMBLY (02152-60010)
Q34
R5
20
R7
2.7
C7
2.2
CR7
L3
30μH
Q35
R6
20
R8
100
CR8
14
11
+10V
R35
22
C32
100
R34
10
C31
0.47
-2V SHUNT REGULATOR
R80
1K
-2V
ADJ.
R81
500
R82
2610
Q18
R84
220
R83
470
Q19
-2V

FIG. 40C

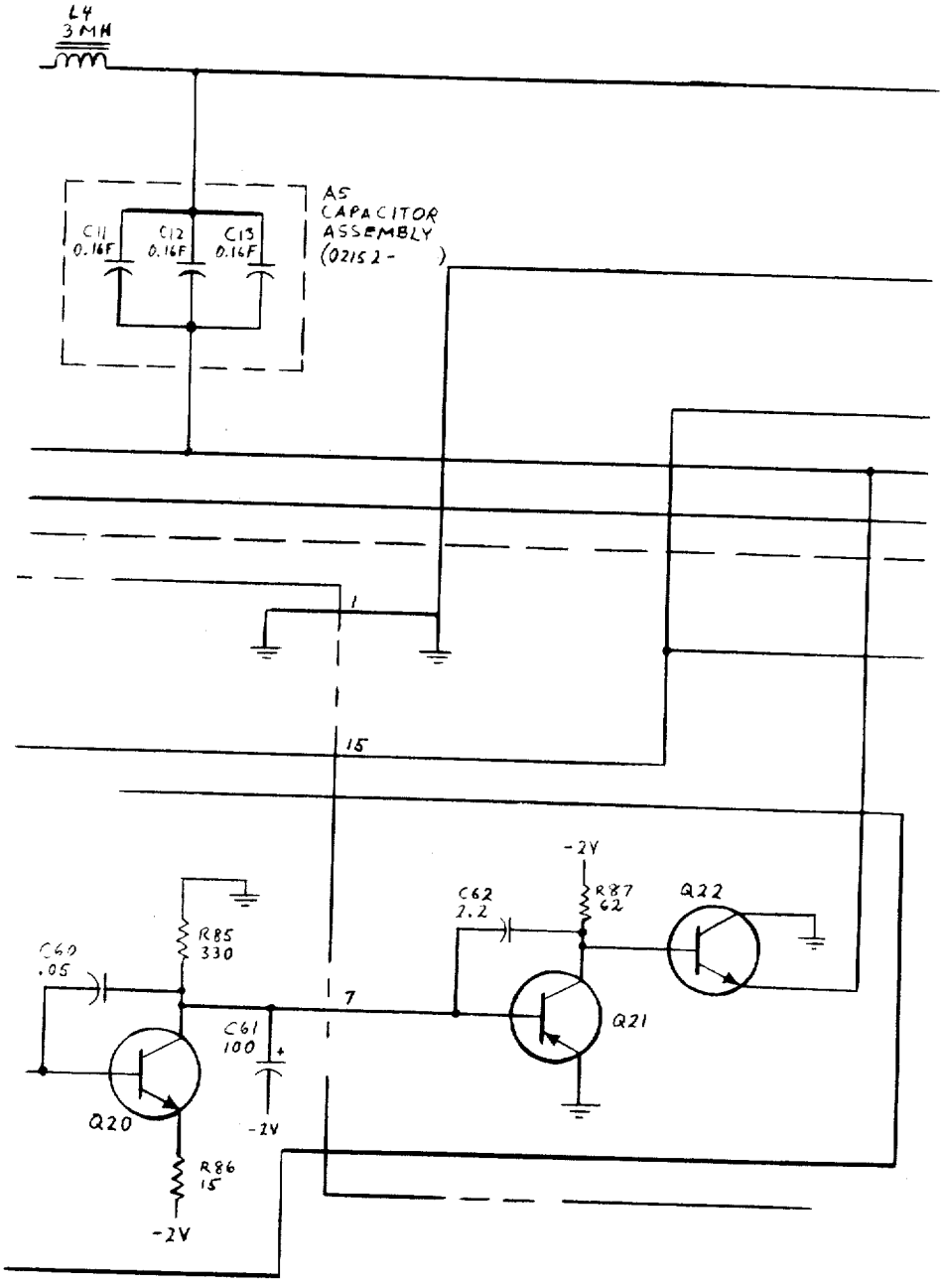

L4
3 MH
A5
CAPACITOR
ASSEMBLY
(02152-
)
C11
0.16F
C12
0.16F
C13
0.16F
1
15
7
-2V
R85
330
C60
.05
C61
100
Q20
R86
15
-2V
C62
2.2
R87
62
-2V
Q22
Q21

FIG. 40D

NOTE: CAPACITOR VALUES IN MICROFARADS UNLESS OTHERWISE INDICATED

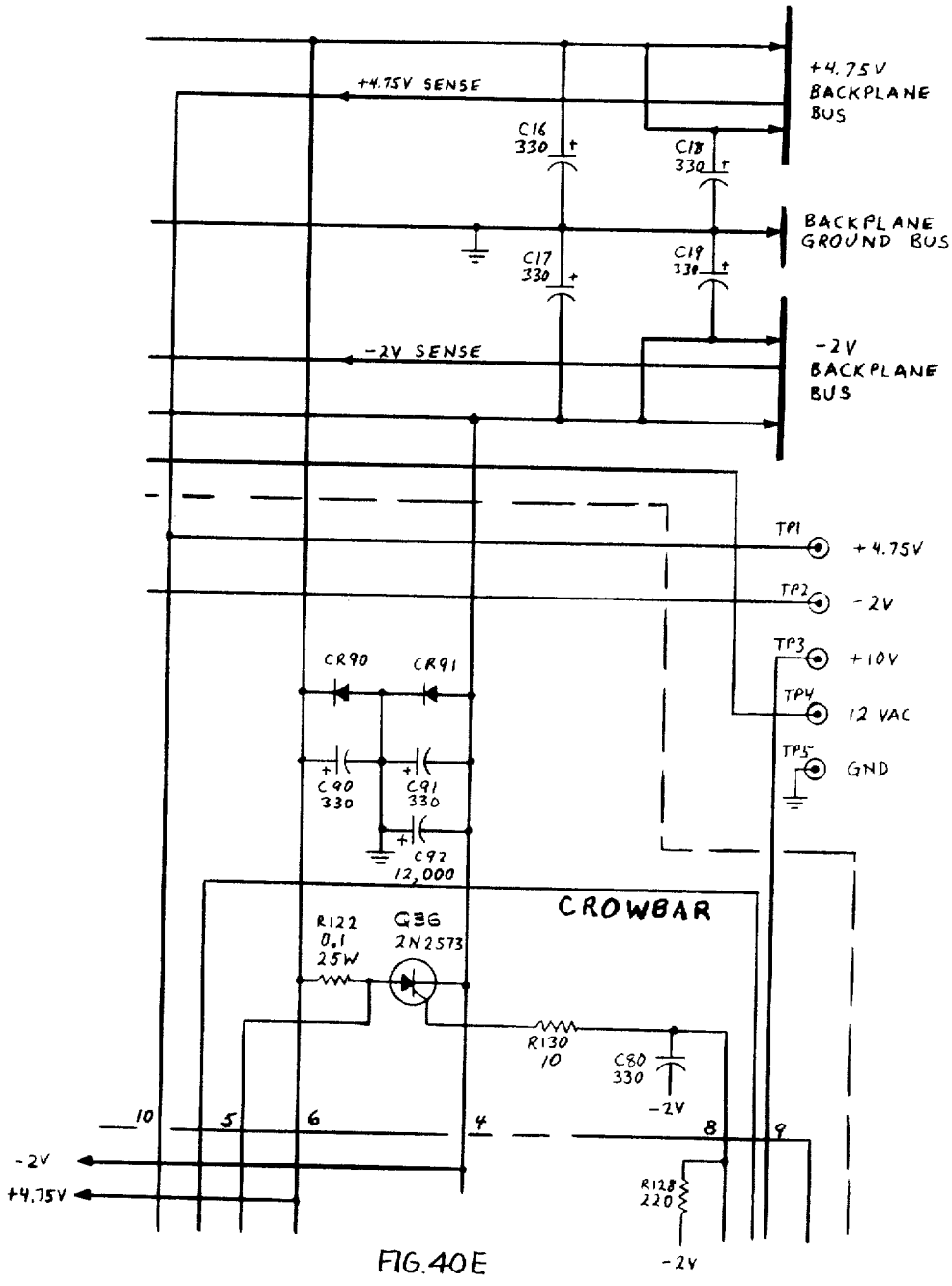



FIG. 40E

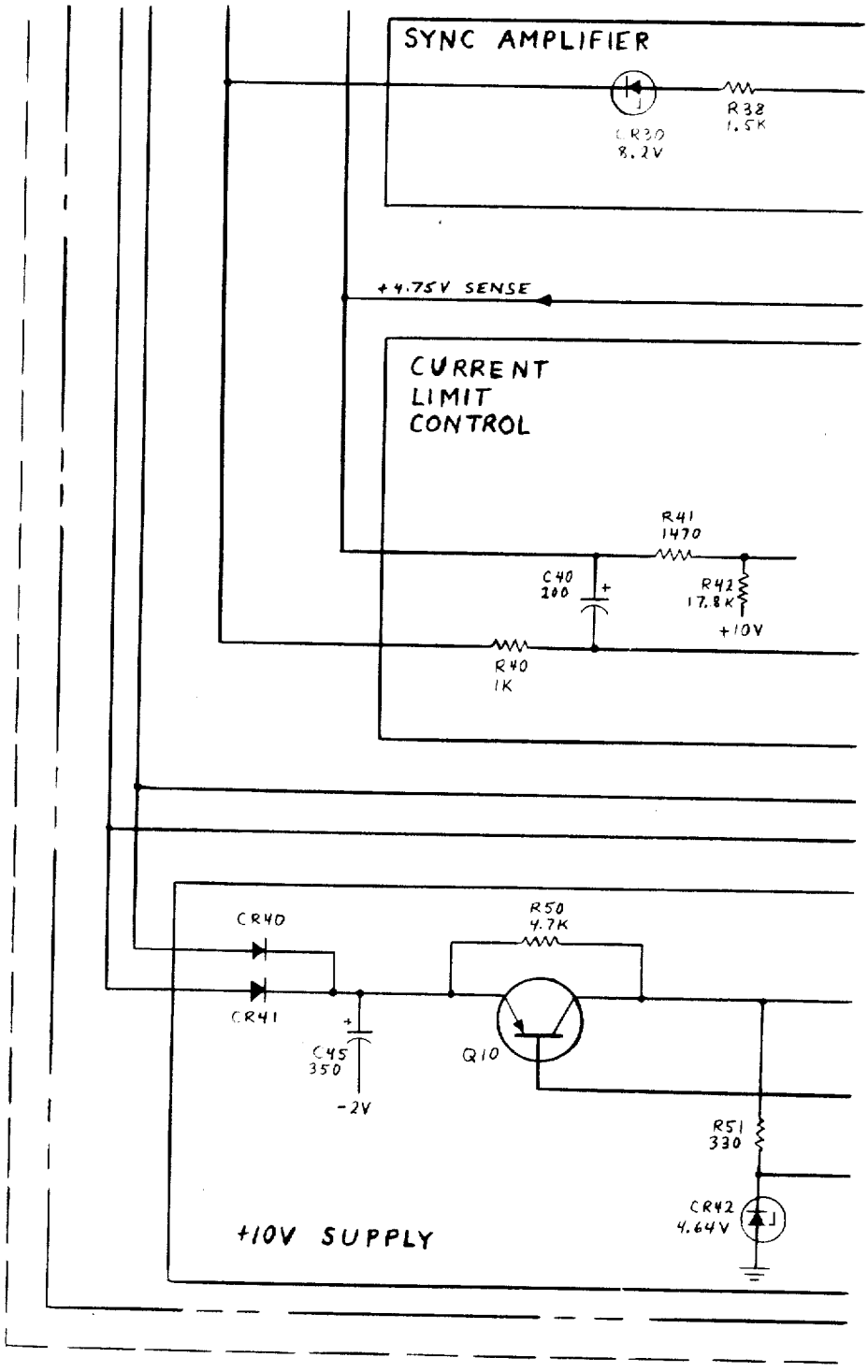

SYNC AMPLIFIER
CR30
8.2V
R38
1.5K
+4.75V SENSE
CURRENT
LIMIT
CONTROL
R41
1470
C40
200
R42
17.8K
+10V
R40
1K
CR40
CR41
R50
4.7K
C45
350
-2V
Q10
R51
330
CR42
4.64V
+10V SUPPLY

FIG. 40F

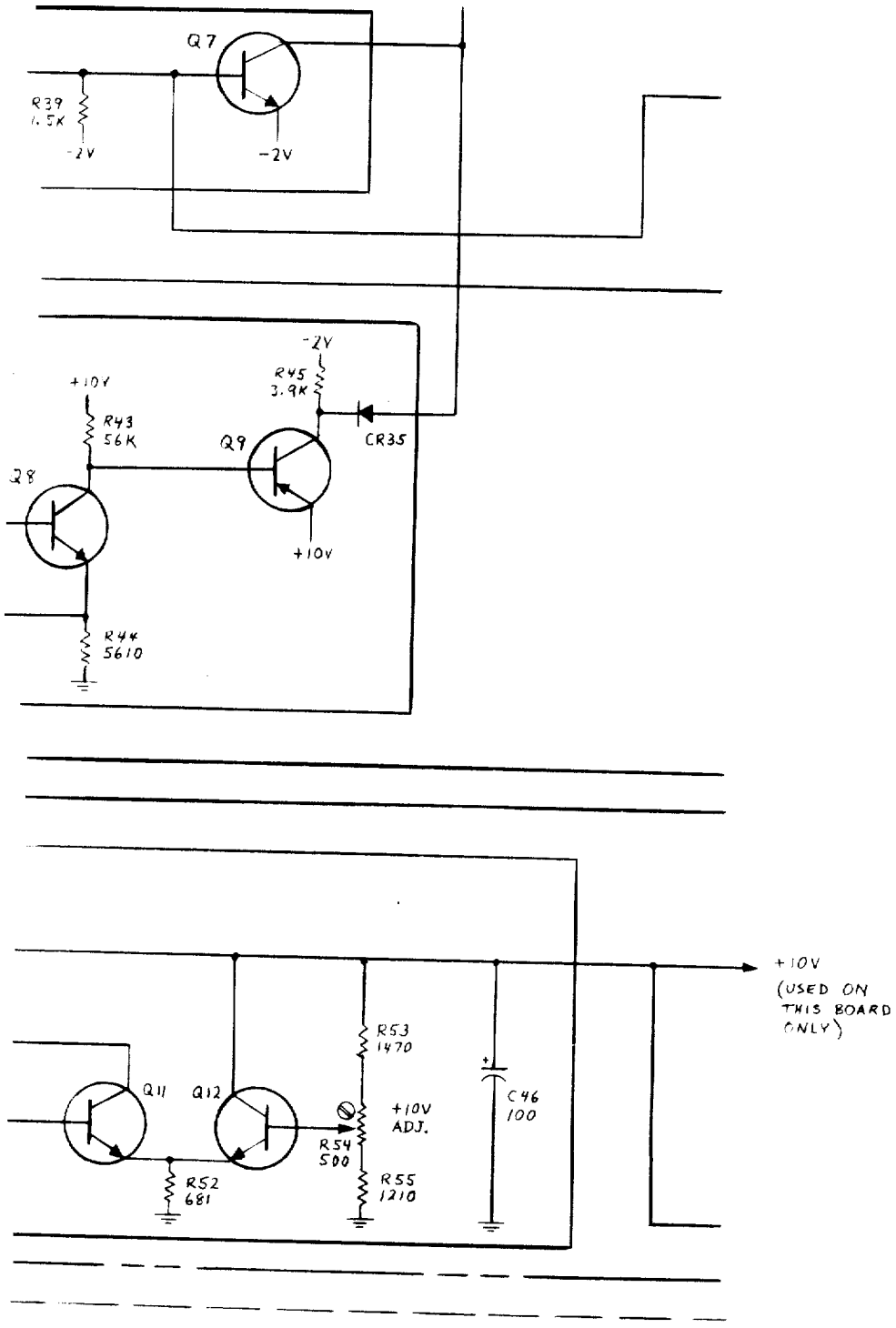

Q7
R39
1.5K
-2V
-2V
-2V
R45
3.9K
+10V
R43
56K
Q9
CR35
Q8
+10V
R44
5610
+10V
(USED ON
THIS BOARD
ONLY)
R53
1470
Q11
Q12
+10V
ADJ.
C46
100
R54
500
R52
681
R55
1210

FIG. 40G

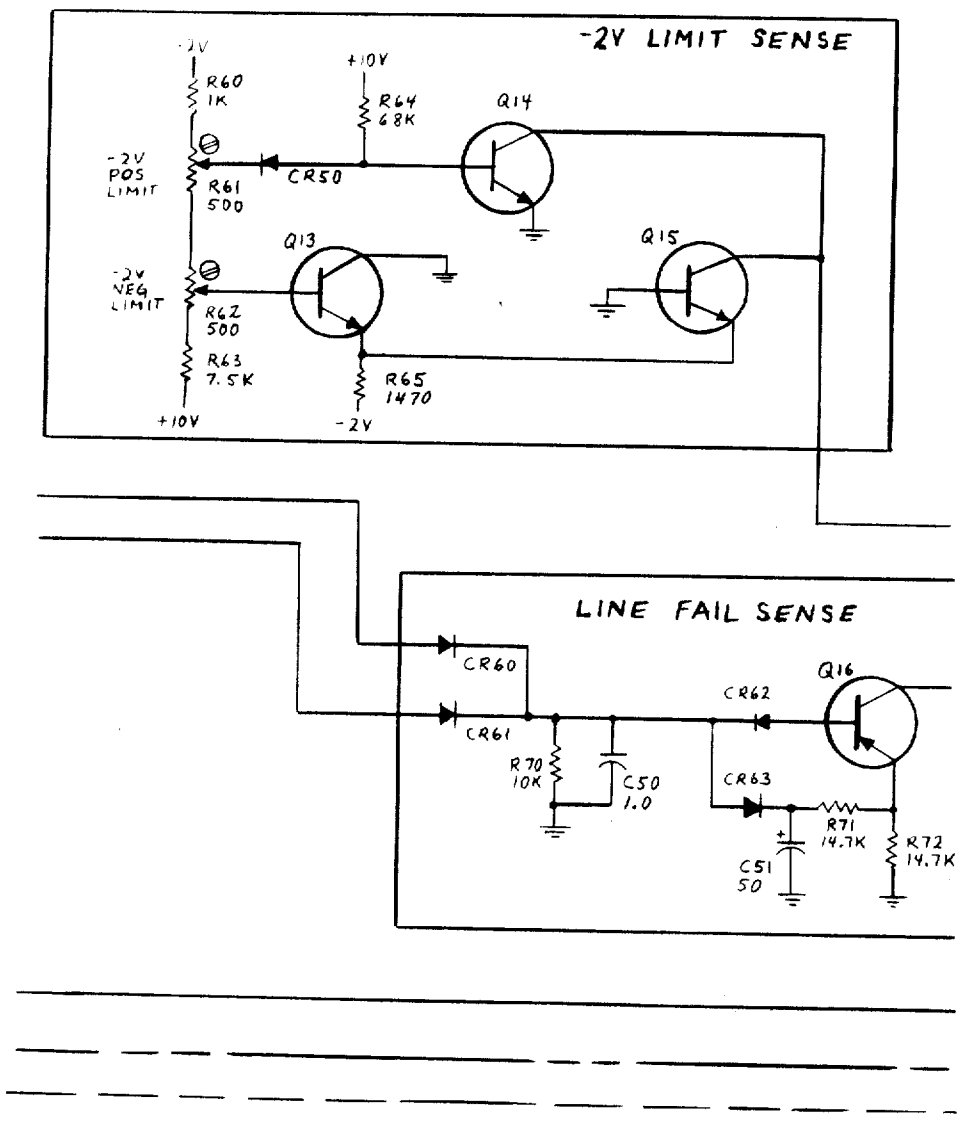

-2V LIMIT SENSE
-2V
+10V
R60
1K
R64
68K
Q14
-2V
POS
LIMIT
R61
500
CR50
Q13
Q15
-2V
NEG
LIMIT
R62
500
R63
7.5K
R65
1470
+10V
-2V
LINE FAIL SENSE
CR60
CR61
Q16
CR62
R70
10K
C50
1.0
CR63
R71
14.7K
R72
14.7K
C51
50

FIG. 40H

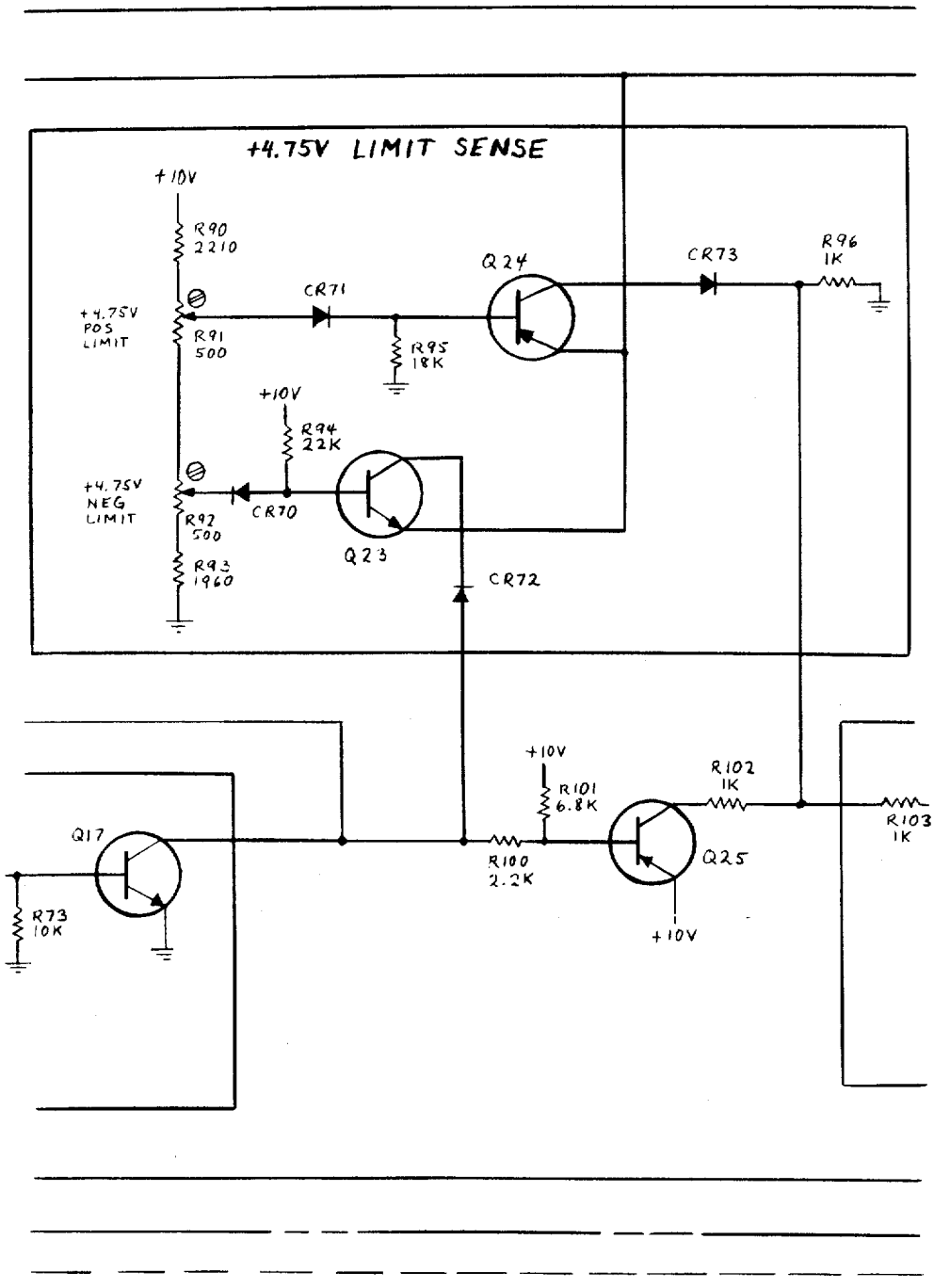

+4.75V LIMIT SENSE
+10V
R90
221Ω
+4.75V
POS
LIMIT
R91
500
CR71
R95
18K
Q24
CR73
R96
1K
+10V
R94
22K
+4.75V
NEG
LIMIT
R92
500
CR70
Q23
R93
1960
CR72
+10V
R101
6.8K
R102
1K
R103
1K
Q17
R100
2.2K
Q25
R73
10K
+10V

FIG. 40I

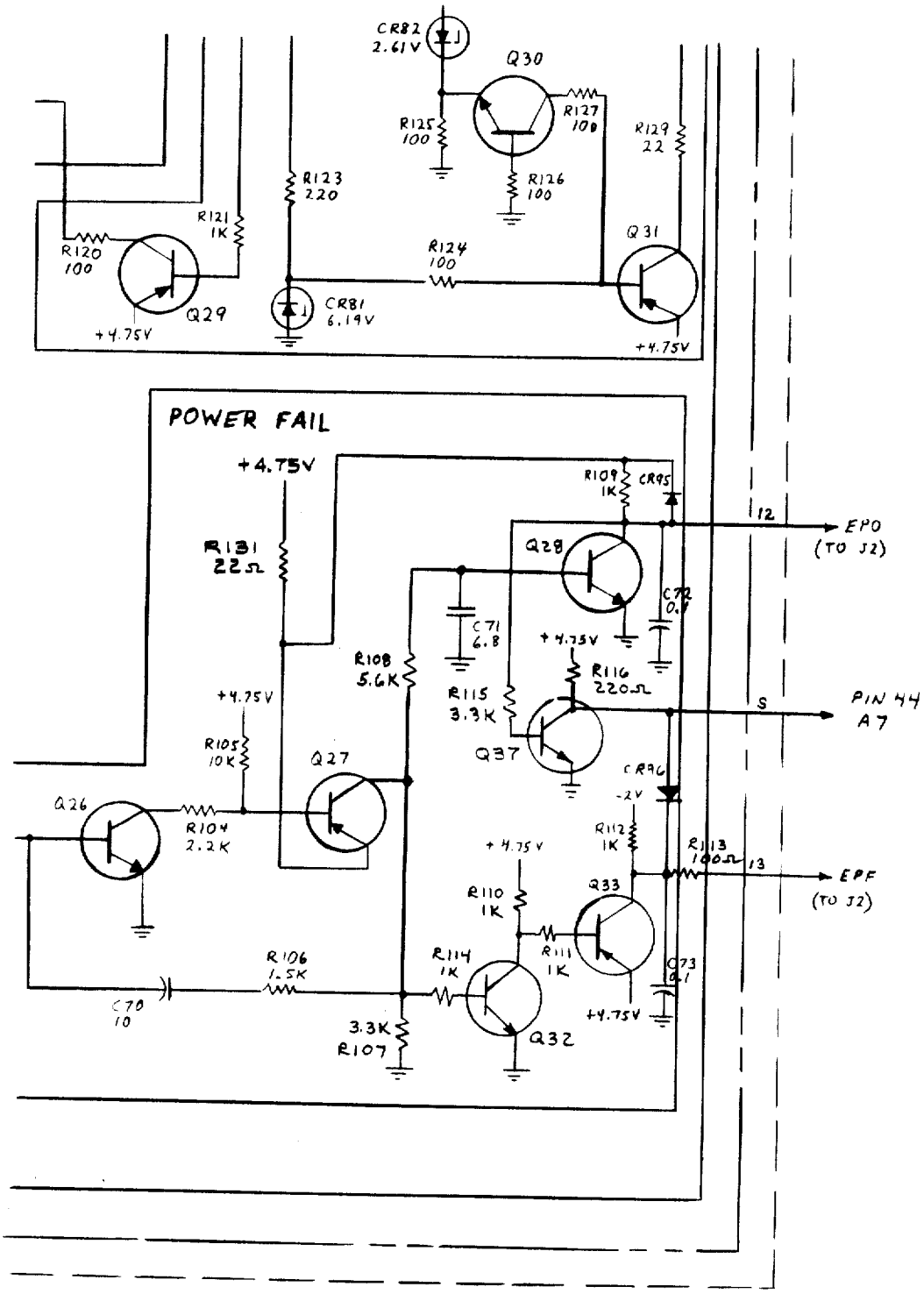

CR82
2.61V
Q30
R125
100
R127
100
R129
22
R126
100
R123
220
R121
1K
R120
100
Q29
+4.75V
CR81
6.19V
R124
100
Q31
+4.75V
POWER FAIL
+4.75V
R109
1K
CR95
12
EPO
(TO J2)
R131
22Ω
Q28
C72
0.1
C71
6.8
+4.75V
R108
5.6K
R116
220Ω
R115
3.3K
S
PIN 44
A7
Q37
+4.75V
R105
10K
Q27
CR96
-2V
Q26
R104
2.2K
R112
1K
+4.75V
R113
100Ω
13
EPF
(TO J2)
R110
1K
Q33
R106
1.5K
R114
1K
R111
1K
C73
0.1
C70
10
+4.75V
3.3K
R107
Q32

FIG. 40J

| | | | | |
|---|---|---|---|---|
| FIG. 40A | FIG. 40B | FIG. 40C | FIG. 40D | FIG. 40E |
| FIG. 40F | FIG. 40G | FIG. 40H | FIG. 40I | FIG. 40J |

FIG. 41

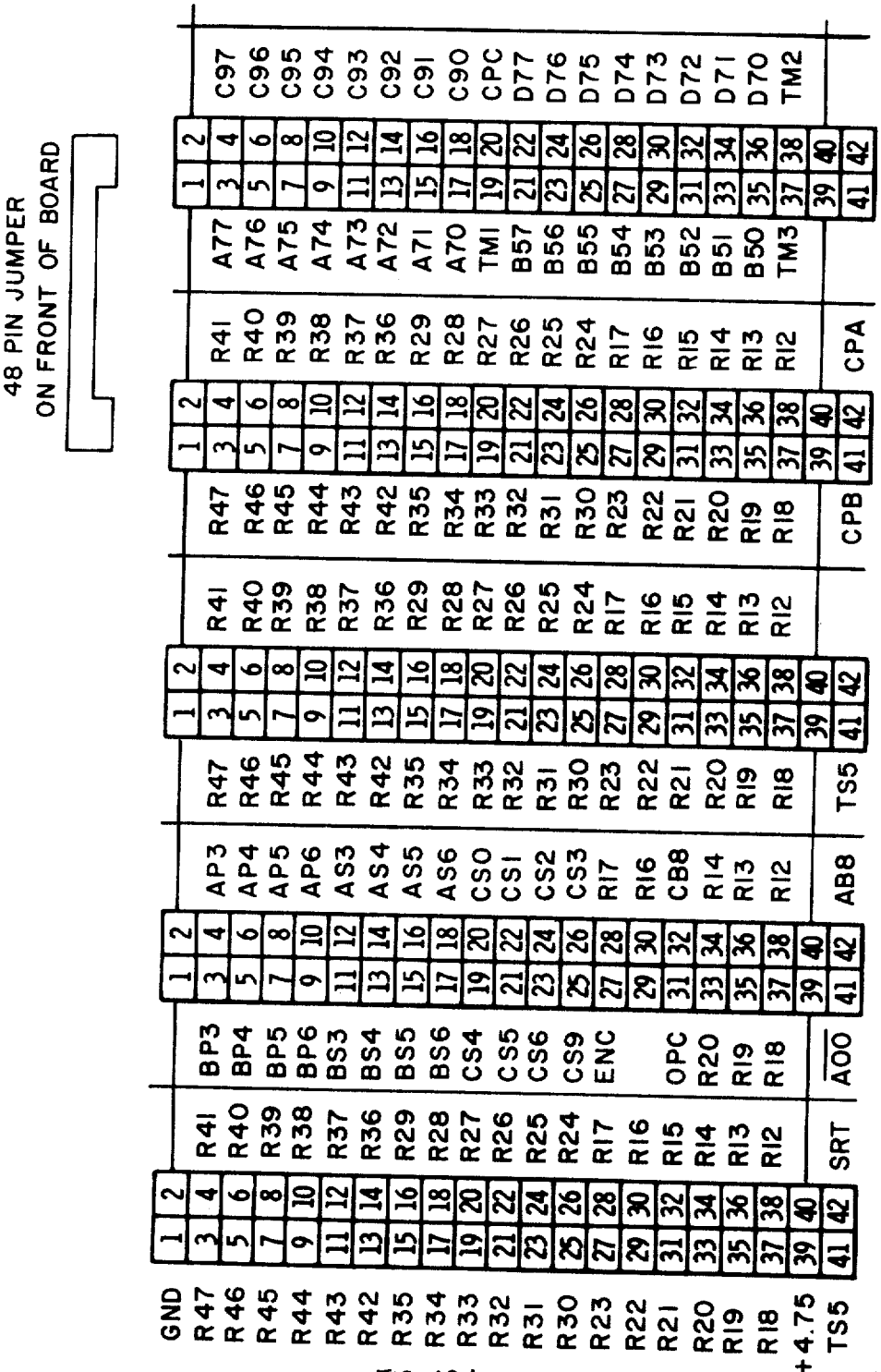

48 PIN JUMPER
ON FRONT OF BOARD
GND
R47
R46
R45
R44
R43
R42
R35
R34
R33
R32
R31
R30
R23
R22
R21
R20
R19
R18
+4.75
TS5
R41 R40 R39 R38 R37 R36 R29 R28 R27 R26 R25 R24 R17 R16 R15 R14 R13 R12 SRT
BP3 BP4 BP5 BP6 BS3 BS4 BS5 BS6 CS4 CS5 CS6 CS9 ENC OPC R20 R19 R18 A00
AP3 AP4 AP5 AP6 AS3 AS4 AS5 AS6 CS0 CS1 CS2 CS3 R17 R16 CB8 R14 R13 R12 AB8
R47 R46 R45 R44 R43 R42 R35 R34 R33 R32 R31 R30 R23 R22 R21 R20 R19 R18 TS5
R41 R40 R39 R38 R37 R36 R29 R28 R27 R26 R25 R24 R17 R16 R15 R14 R13 R12
R47 R46 R45 R44 R43 R42 R35 R34 R33 R32 R31 R30 R23 R22 R21 R20 R19 R18 CPB
R41 R40 R39 R38 R37 R36 R29 R28 R27 R26 R25 R24 R17 R16 R15 R14 R13 R12 CPA
A77 A76 A75 A74 A73 A72 A71 A70 TM1 B57 B56 B55 B54 B53 B52 B51 B50 TM3
C97 C96 C95 C94 C93 C92 C91 C90 CPC D77 D76 D75 D74 D73 D72 D71 D70 TM2

FIG. 42A

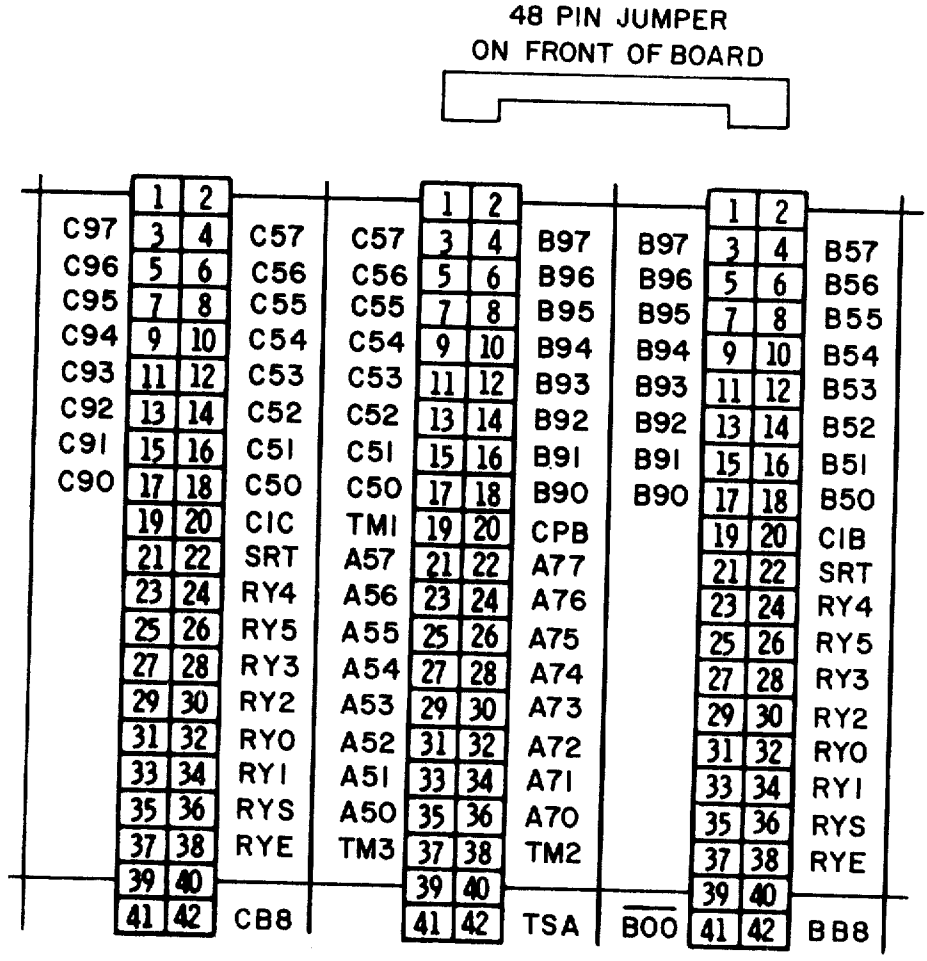

48 PIN JUMPER
ON FRONT OF BOARD
C97 C96 C95 C94 C93 C92 C91 C90
C57 C56 C55 C54 C53 C52 C51 C50 CIC SRT RY4 RY5 RY3 RY2 RYO RYI RYS RYE CB8
C57 C56 C55 C54 C53 C52 C51 C50 TMI A57 A56 A55 A54 A53 A52 A51 A50 TM3
B97 B96 B95 B94 B93 B92 B91 B90 CPB A77 A76 A75 A74 A73 A72 A71 A70 TM2 TSA
B97 B96 B95 B94 B93 B92 B91 B90 BOO
B57 B56 B55 B54 B53 B52 B51 B50 CIB SRT RY4 RY5 RY3 RY2 RYO RYI RYS RYE BB8

FIG. 42B

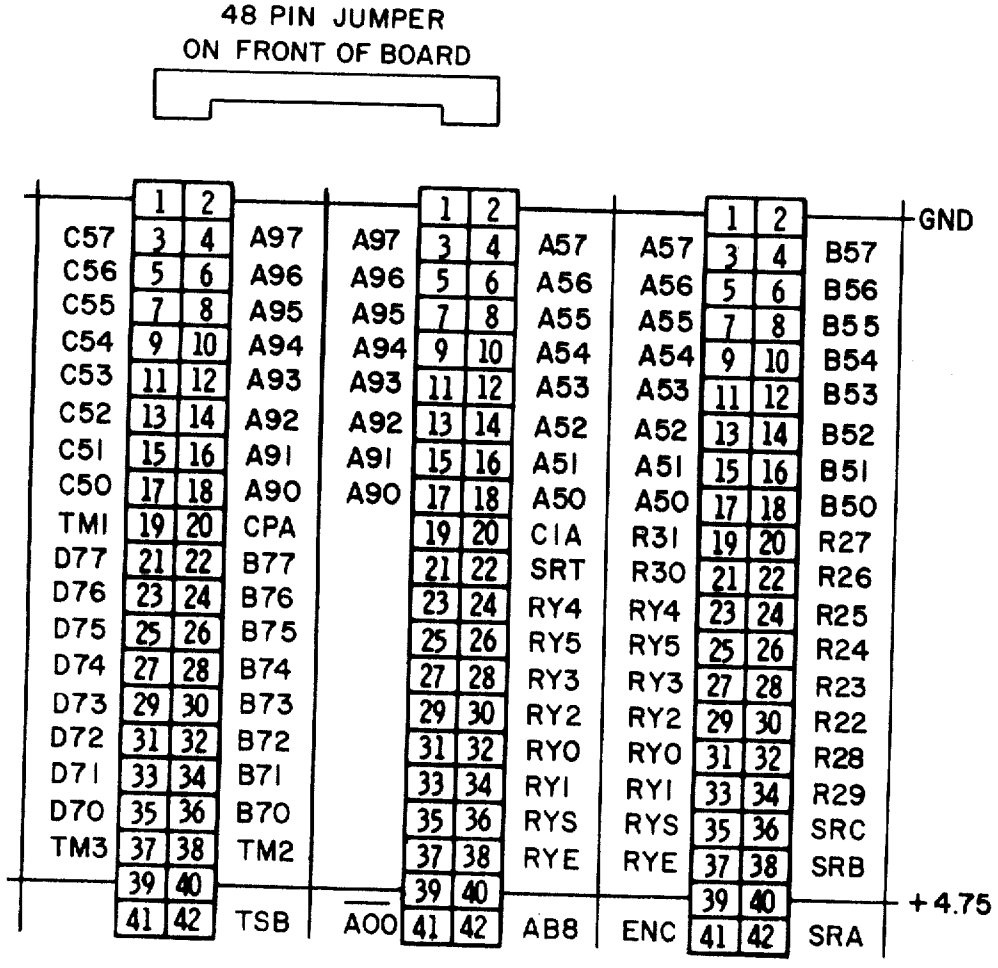

48 PIN JUMPER
ON FRONT OF BOARD
C57 C56 C55 C54 C53 C52 C51 C50 TMI D77 D76 D75 D74 D73 D72 D71 D70 TM3
A97 A96 A95 A94 A93 A92 A91 A90 CPA B77 B76 B75 B74 B73 B72 B71 B70 TM2 TSB
A97 A96 A95 A94 A93 A92 A91 A90 AOO
A57 A56 A55 A54 A53 A52 A51 A50 CIA SRT RY4 RY5 RY3 RY2 RYO RYI RYS RYE AB8
A57 A56 A55 A54 A53 A52 A51 A50 R31 R30 RY4 RY5 RY3 RY2 RYO RYI RYS RYE ENC
GND
B57 B56 B55 B54 B53 B52 B51 B50 R27 R26 R25 R24 R23 R22 R28 R29 SRC SRB SRA
+4.75

FIG. 42C

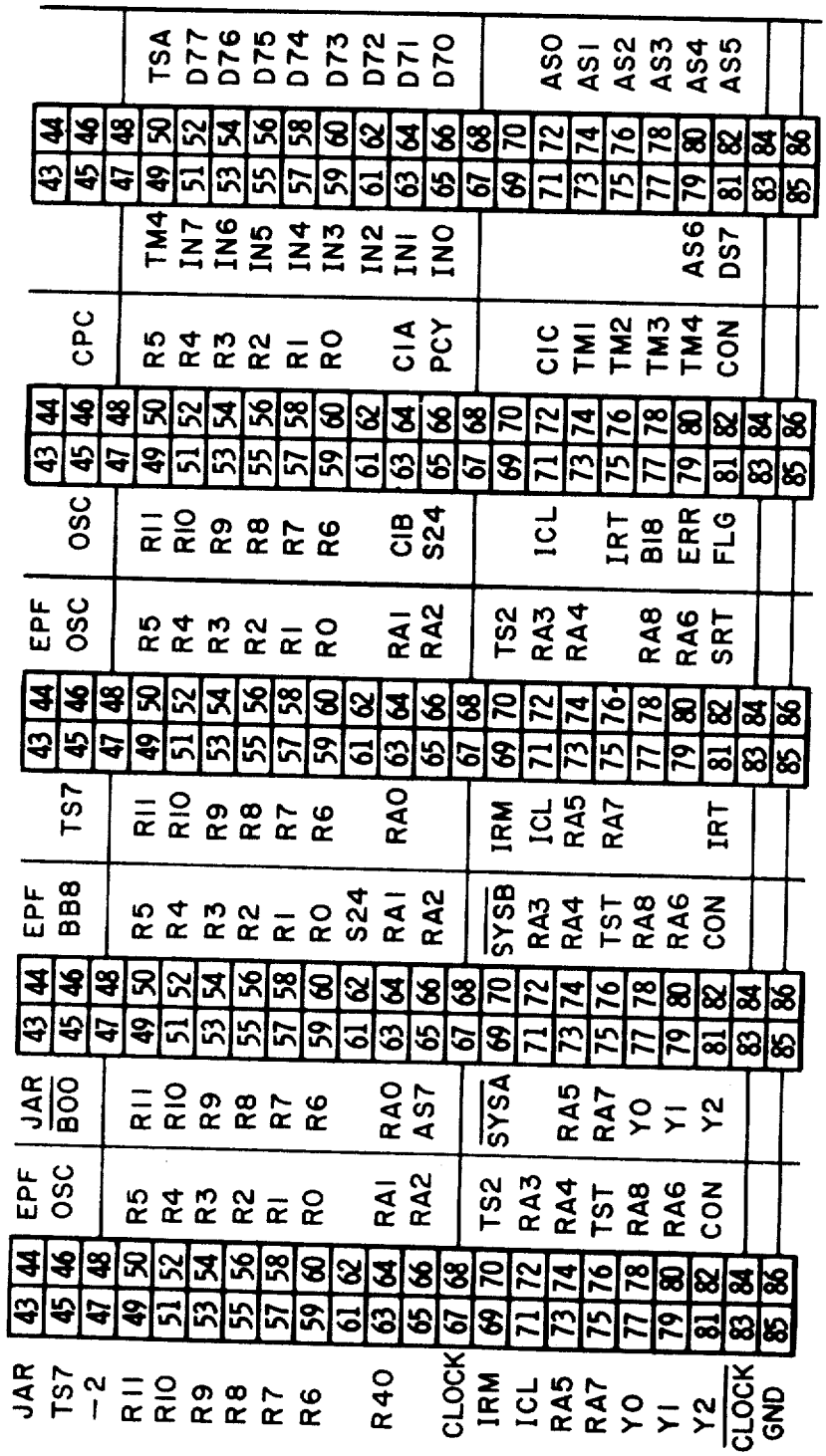



FIG. 42D

NOTE:

DIAGRAM VIEWED FROM BACK PLANE

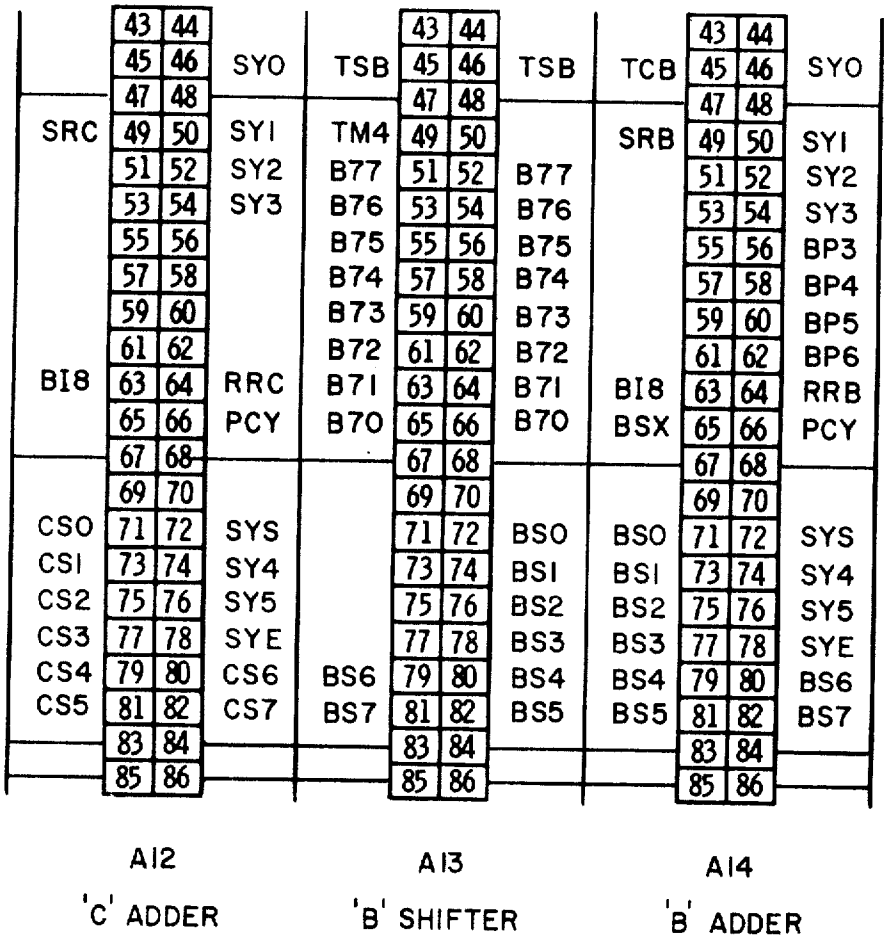

43 44
45 46
47 48
49 50
51 52
53 54
55 56
57 58
59 60
61 62
63 64
65 66
67 68
69 70
71 72
73 74
75 76
77 78
79 80
81 82
83 84
85 86
SYO
SYI
SY2
SY3
RRC
PCY
SYS
SY4
SY5
SYE
CS6
CS7
SRC
BI8
CSO
CSI
CS2
CS3
CS4
CS5
TSB
TM4
B77
B76
B75
B74
B73
B72
B71
B70
BS6
BS7
TSB
B77
B76
B75
B74
B73
B72
B71
B70
BSO
BSI
BS2
BS3
BS4
BS5
TCB
SRB
BI8
BSX
SYO
SYI
SY2
SY3
BP3
BP4
BP5
BP6
RRB
PCY
SYS
SY4
SY5
SYE
BS6
BS7
BSO
BSI
BS2
BS3
BS4
BS5
A12
'C' ADDER
A13
'B' SHIFTER
A14
'B' ADDER

FIG. 42E

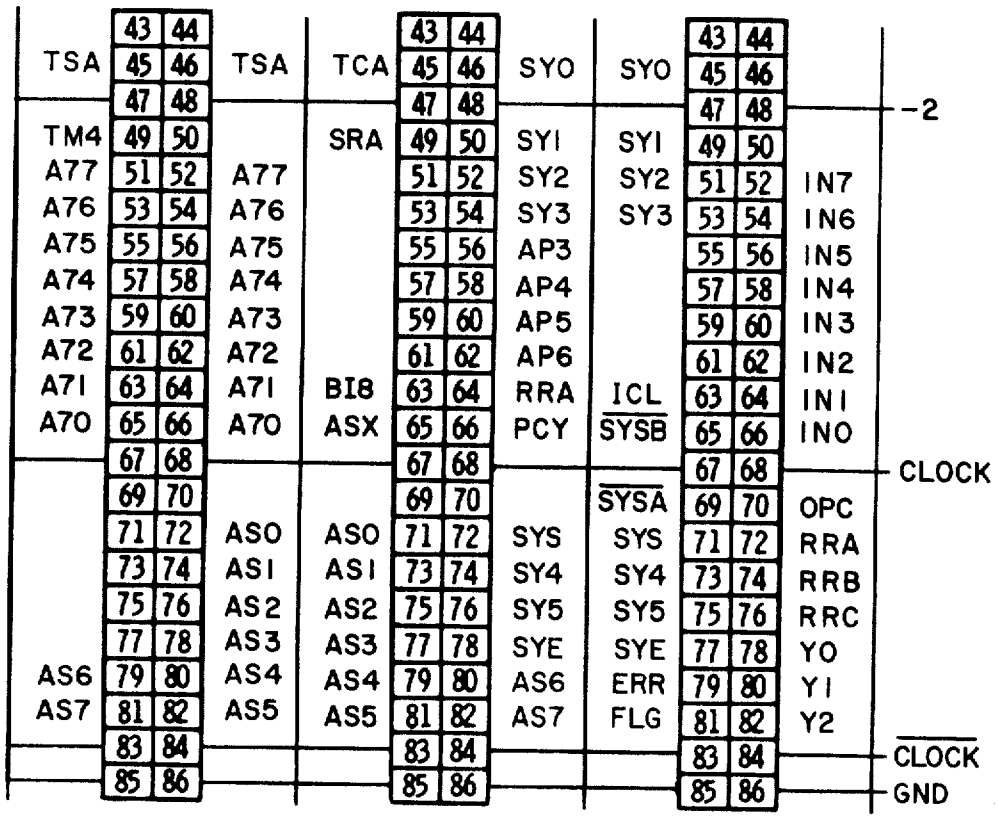

TSA
TM4
A77
A76
A75
A74
A73
A72
A71
A70
AS6
AS7
TSA
A77
A76
A75
A74
A73
A72
A71
A70
AS0
AS1
AS2
AS3
AS4
AS5
TCA
SRA
BI8
ASX
AS0
AS1
AS2
AS3
AS4
AS5
SYO
SYI
SY2
SY3
AP3
AP4
AP5
AP6
RRA
PCY
SYS
SY4
SY5
SYE
AS6
AS7
SYO
SYI
SY2
SY3
ICL
SYSB
SYSA
SYS
SY4
SY5
SYE
ERR
FLG
IN7
IN6
IN5
IN4
IN3
IN2
IN1
IN0
OPC
RRA
RRB
RRC
YO
Y1
Y2
-2
CLOCK
CLOCK
GND
AI5
'A' SHIFTER
AI6
'A' ADDER
AI7
INPUT / OUTPUT

FIG. 42F

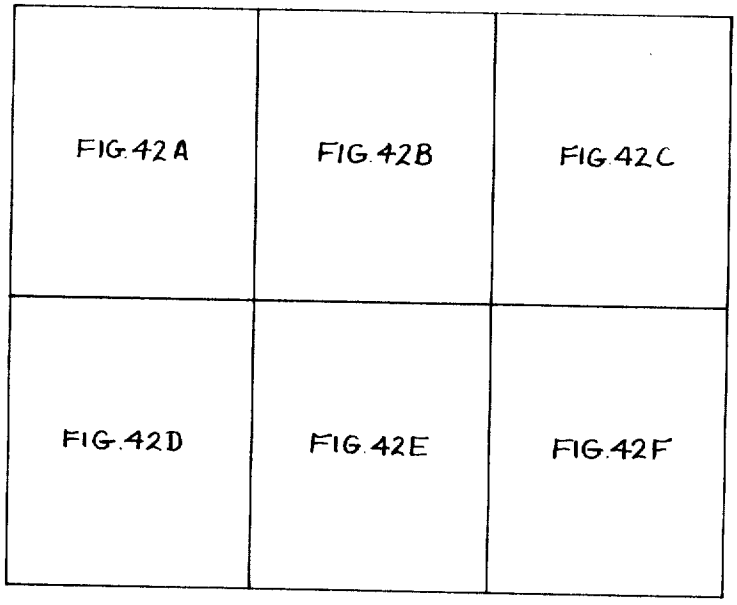



FIG. 43

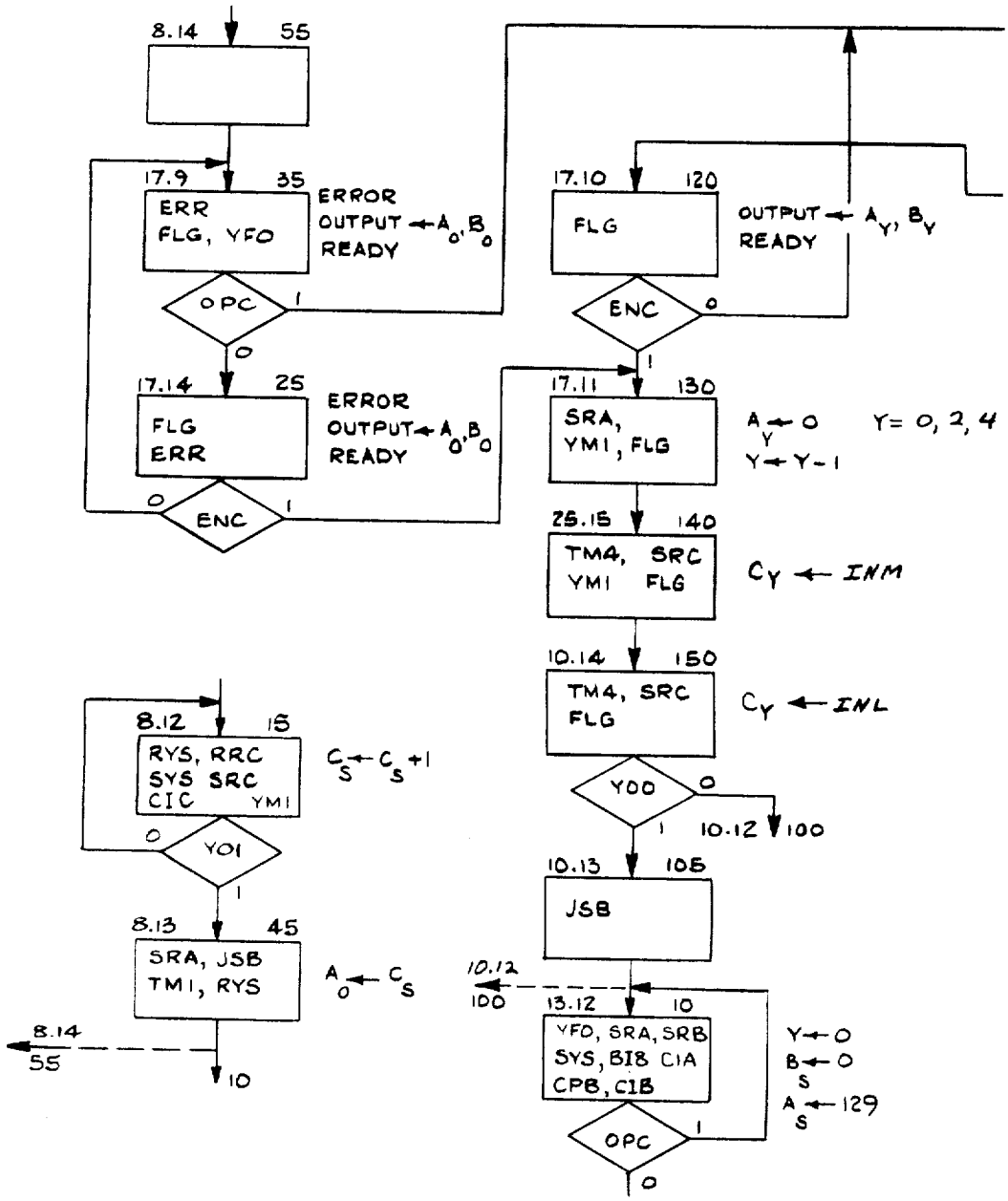

8.14
55
17.9
35
ERR
FLG, YFO
ERROR
OUTPUT ← A0, B0
READY
OPC
1
0
17.14
25
FLG
ERR
ERROR
OUTPUT ← A0, B0
READY
0
ENC
1
17.10
120
FLG
OUTPUT ← AY, BY
READY
ENC
0
1
17.11
130
SRA,
YM1, FLG
AY ← 0
Y ← Y-1
Y= 0, 2, 4
25.15
140
TM4, SRC
YM1 FLG
CY ← INM
10.14
150
TM4, SRC
FLG
CY ← INL
YOO
0
1
10.12
100
10.13
105
JSB
8.12
15
RYS, RRC
SYS SRC
CIC YM1
CS ← CS +1
0
YO1
1
8.13
45
SRA, JSB
TM1, RYS
A0 ← CS
8.14
55
10
10.12
100
13.12
10
YFO, SRA, SRB
SYS, BI8 CIA
CP8, C18
Y ← 0
BS ← 0
AS ← 129
OPC
1
0

FIG. 44A

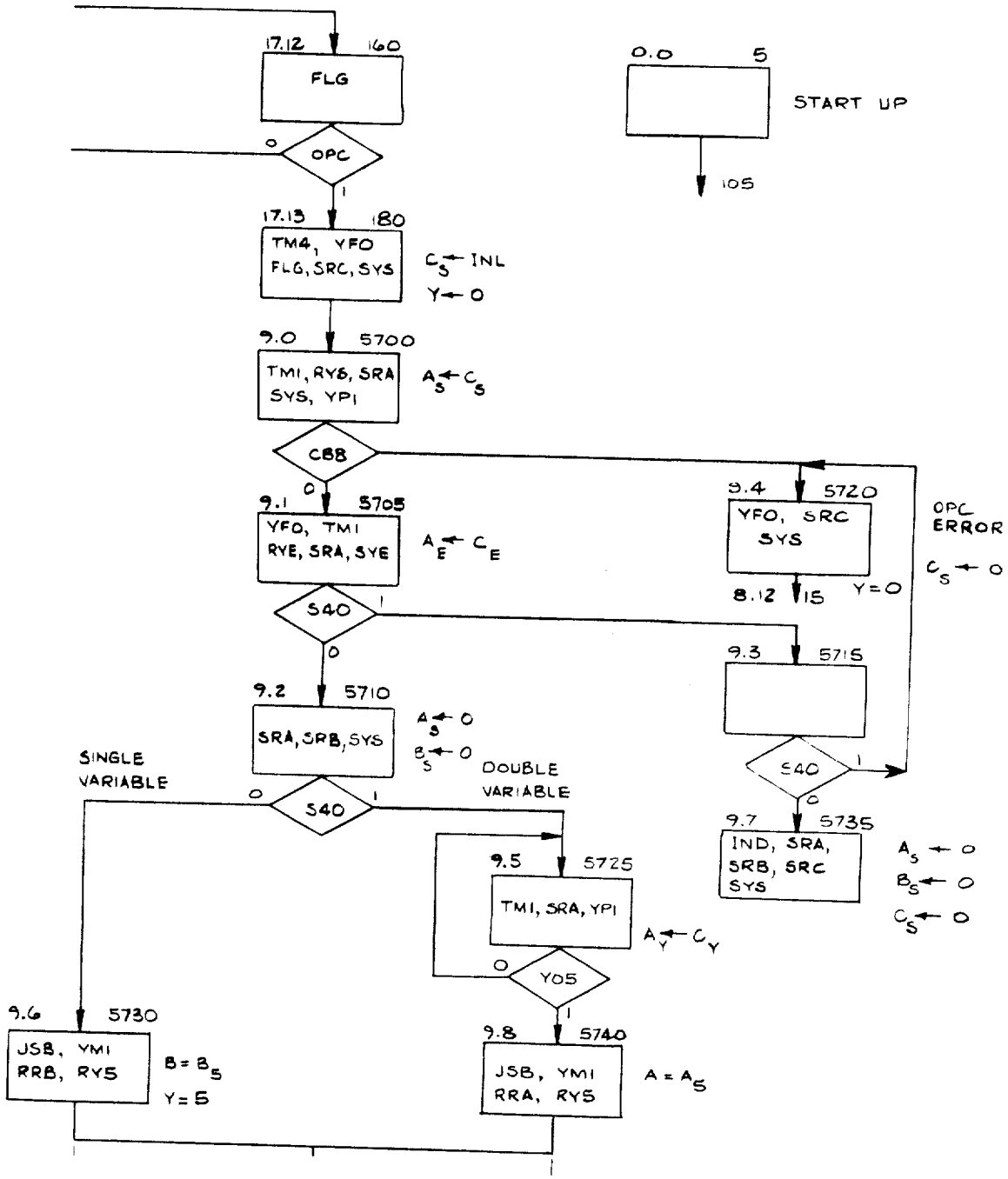

17.12
160
FLG
OPC
0
1
17.13
180
TM4, YFO
FLG, SRC, SYS
$C_S \leftarrow$ INL
$Y \leftarrow 0$
9.0
5700
TM1, RY6, SRA
SYS, YP1
$A_S \leftarrow C_S$
CBB
0
9.1
5705
YFO, TM1
RYE, SRA, SYE
$A_E \leftarrow C_E$
S40
1
0
9.2
5710
SRA, SRB, SYS
$A_S \leftarrow 0$
$B_S \leftarrow 0$
SINGLE VARIABLE
DOUBLE VARIABLE
0
S40
1
9.6
5730
JSB, YM1
RRB, RY5
$B = B_S$
$Y = 5$
9.5
5725
TM1, SRA, YP1
$A_Y \leftarrow C_Y$
0
Y05
1
9.8
5740
JSB, YM1
RRA, RY5
$A = A_S$
0.0
5
START UP
105
9.4
5720
YFO, SRC
SYS
OPC ERROR
$C_S \leftarrow 0$
8.12
15
Y=0
9.3
5715
S40
1
0
9.7
5735
IND, SRA,
SRB, SRC
SYS
$A_S \leftarrow 0$
$B_S \leftarrow 0$
$C_S \leftarrow 0$

FIG. 44B

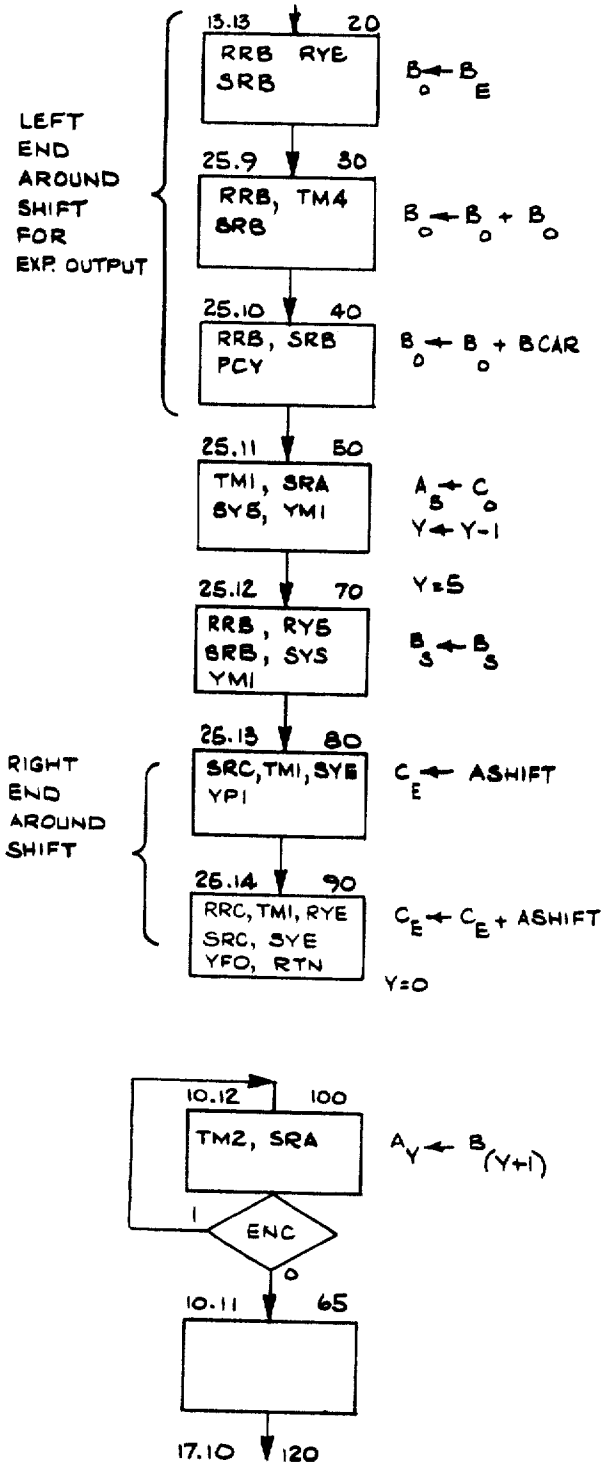

13.13
20
RRB RYE
SRB
B0 ← BE
LEFT END AROUND SHIFT FOR EXP. OUTPUT
25.9
30
RRB, TM4
SRB
B0 ← B0 + B0
25.10
40
RRB, SRB
PCY
B0 ← B0 + BCAR
25.11
50
TM1, SRA
SYS, YM1
AS ← C0
Y ← Y-1
25.12
70
Y=5
RRB, RYS
SRB, SYS
YM1
BS ← BS
25.13
80
RIGHT END AROUND SHIFT
SRC, TM1, SYE
YP1
CE ← ASHIFT
25.14
90
RRC, TM1, RYE
SRC, SYE
YFO, RTN
CE ← CE + ASHIFT
Y=0
10.12
100
TM2, SRA
AY ← B(Y+1)
1
ENC
0
10.11
65
17.10
120

FIG. 44C

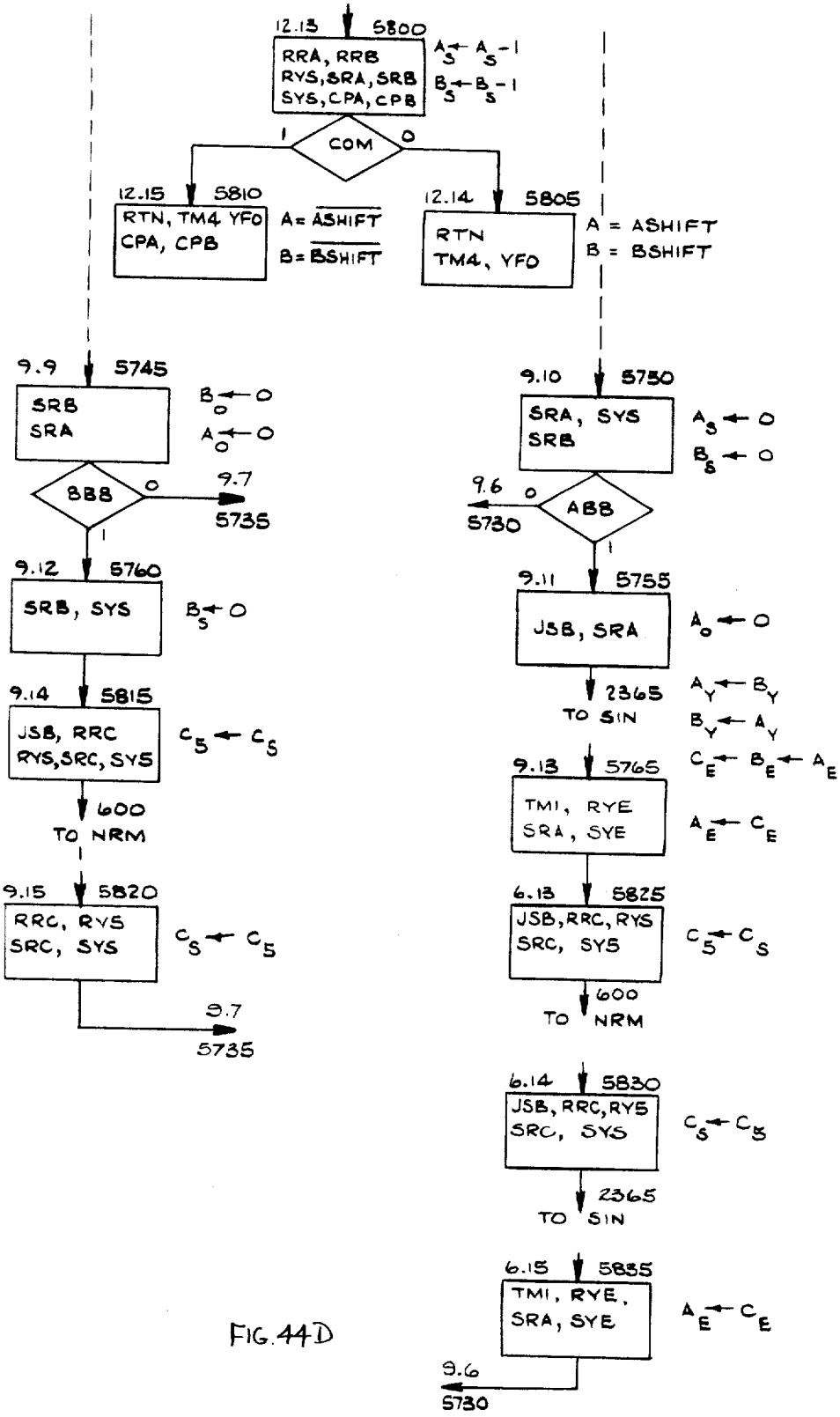

12.13 5800
RRA, RRB
RYS, SRA, SRB
SYS, CPA, CPB
$A_S \leftarrow A_S - 1$
$B_S \leftarrow B_S - 1$
COM
1
0
12.15 5810
RTN, TM4 YF0
CPA, CPB
A = ASHIFT
B = BSHIFT
12.14 5805
RTN
TM4, YF0
A = ASHIFT
B = BSHIFT
9.9 5745
SRB
SRA
$B_0 \leftarrow 0$
$A_0 \leftarrow 0$
BBB
0
9.7
5735
1
9.12 5760
SRB, SYS
$B_S \leftarrow 0$
9.14 5815
JSB, RRC
RYS, SRC, SYS
$C_5 \leftarrow C_5$
600
TO NRM
9.15 5820
RRC, RYS
SRC, SYS
$C_S \leftarrow C_5$
9.7
5735
9.10 5750
SRA, SYS
SRB
$A_S \leftarrow 0$
$B_S \leftarrow 0$
9.6
0
5730
ABB
1
9.11 5755
JSB, SRA
$A_0 \leftarrow 0$
2365
TO SIN
$A_Y \leftarrow B_Y$
$B_Y \leftarrow A_Y$
9.13 5765
TMI, RYE
SRA, SYE
$C_E \leftarrow B_E \leftarrow A_E$
$A_E \leftarrow C_E$
6.13 5825
JSB, RRC, RYS
SRC, SYS
$C_5 \leftarrow C_S$
600
TO NRM
6.14 5830
JSB, RRC, RYS
SRC, SYS
$C_S \leftarrow C_5$
2365
TO SIN
6.15 5835
TMI, RYE,
SRA, SYE
$A_E \leftarrow C_E$
9.6
5730

FIG. 44D

ENTRY ROUTINE FLOWCHART

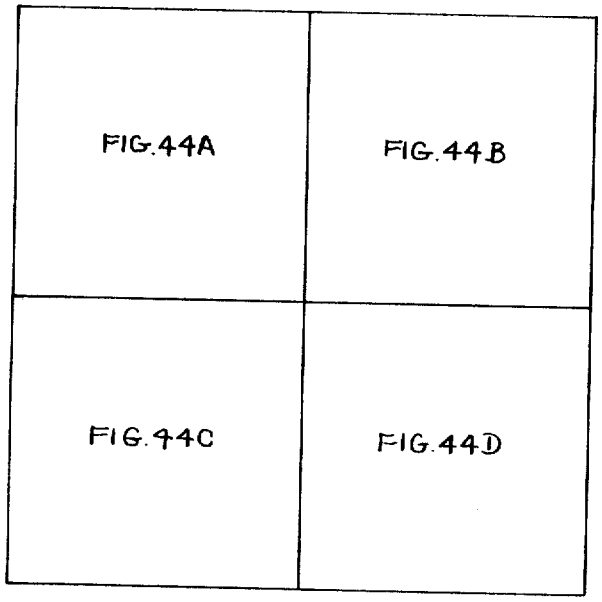



FIG. 45

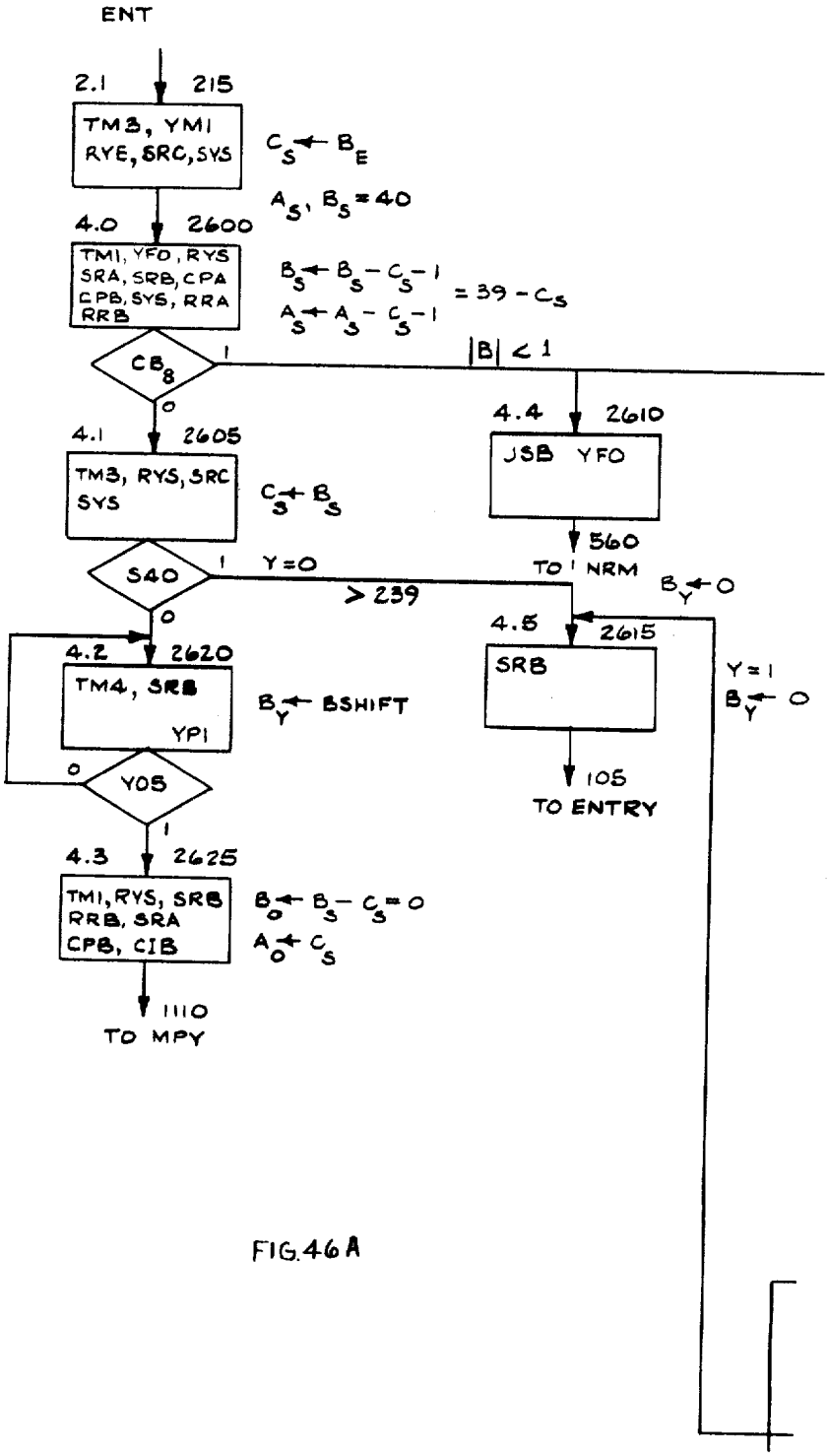

ENT
2.1
215
TM3, YM1
RYE, SRC, SYS
$C_S \leftarrow B_E$
$A_S, B_S = 40$
4.0
2600
TM1, YFO, RYS
SRA, SRB, CPA
CPB, SYS, RRA
RRB
$B_S \leftarrow B_S - C_S - 1$
$A_S \leftarrow A_S - C_S - 1$
$= 39 - C_S$
$CB_8$
1
$|B| < 1$
0
4.1
2605
TM3, RYS, SRC
SYS
$C_S \leftarrow B_S$
S40
1
Y=0
> 239
0
4.2
2620
TM4, SRB
YP1
$B_Y \leftarrow$ BSHIFT
0
Y05
1
4.3
2625
TM1, RYS, SRB
RRB, SRA
CPB, CIB
$B_0 \leftarrow B_S - C_S = 0$
$A_0 \leftarrow C_S$
1110
TO MPY
4.4
2610
JSB YFO
560
TO ! NRM
$B_Y \leftarrow 0$
4.5
2615
SRB
105
TO ENTRY
Y=1
$B_Y \leftarrow 0$

FIG. 46A

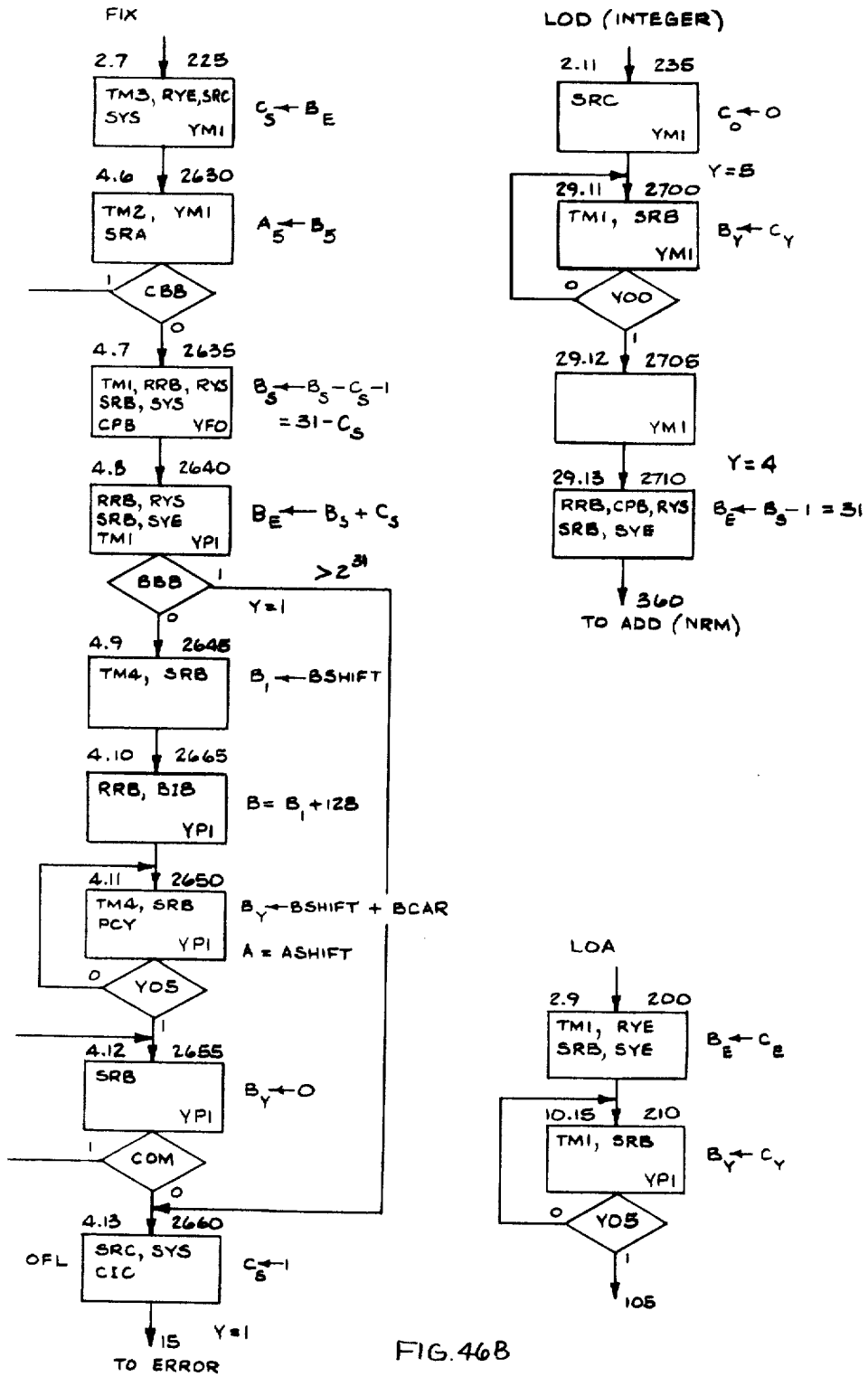

FIX
2.7 225
TM3, RYE, SRC SYS YM1
$C_S \leftarrow B_E$
4.6 2630
TM2, YM1 SRA
$A_5 \leftarrow B_5$
CBB
1
0
4.7 2635
TM1, RRB, RYS SRB, SYS CPB YFO
$B_S \leftarrow B_S - C_S - 1$
$= 31 - C_S$
4.8 2640
RRB, RYS SRB, SYE TM1 YP1
$B_E \leftarrow B_S + C_S$
BBB
1
$> 2^{31}$
Y=1
0
4.9 2645
TM4, SRB
$B_1 \leftarrow$ BSHIFT
4.10 2665
RRB, B1B YP1
$B = B_1 + 128$
4.11 2650
TM4, SRB PCY YP1
$B_Y \leftarrow$ BSHIFT + BCAR
A = ASHIFT
YO5
0
1
4.12 2655
SRB YP1
$B_Y \leftarrow 0$
COM
1
0
4.13 2660
OFL
SRC, SYS CIC
$C_S \leftarrow 1$
15
Y=1
TO ERROR
LOD (INTEGER)
2.11 235
SRC YM1
$C_0 \leftarrow 0$
Y=B
29.11 2700
TM1, SRB YM1
$B_Y \leftarrow C_Y$
YOO
0
1
29.12 2706
YM1
Y=4
29.13 2710
RRB, CPB, RYS SRB, SYE
$B_E \leftarrow B_S - 1 = 31$
360
TO ADD (NRM)
LOA
2.9 200
TM1, RYE SRB, SYE
$B_E \leftarrow C_E$
10.15 210
TM1, SRB YP1
$B_Y \leftarrow C_Y$
YO5
0
1
105

FIG. 46B

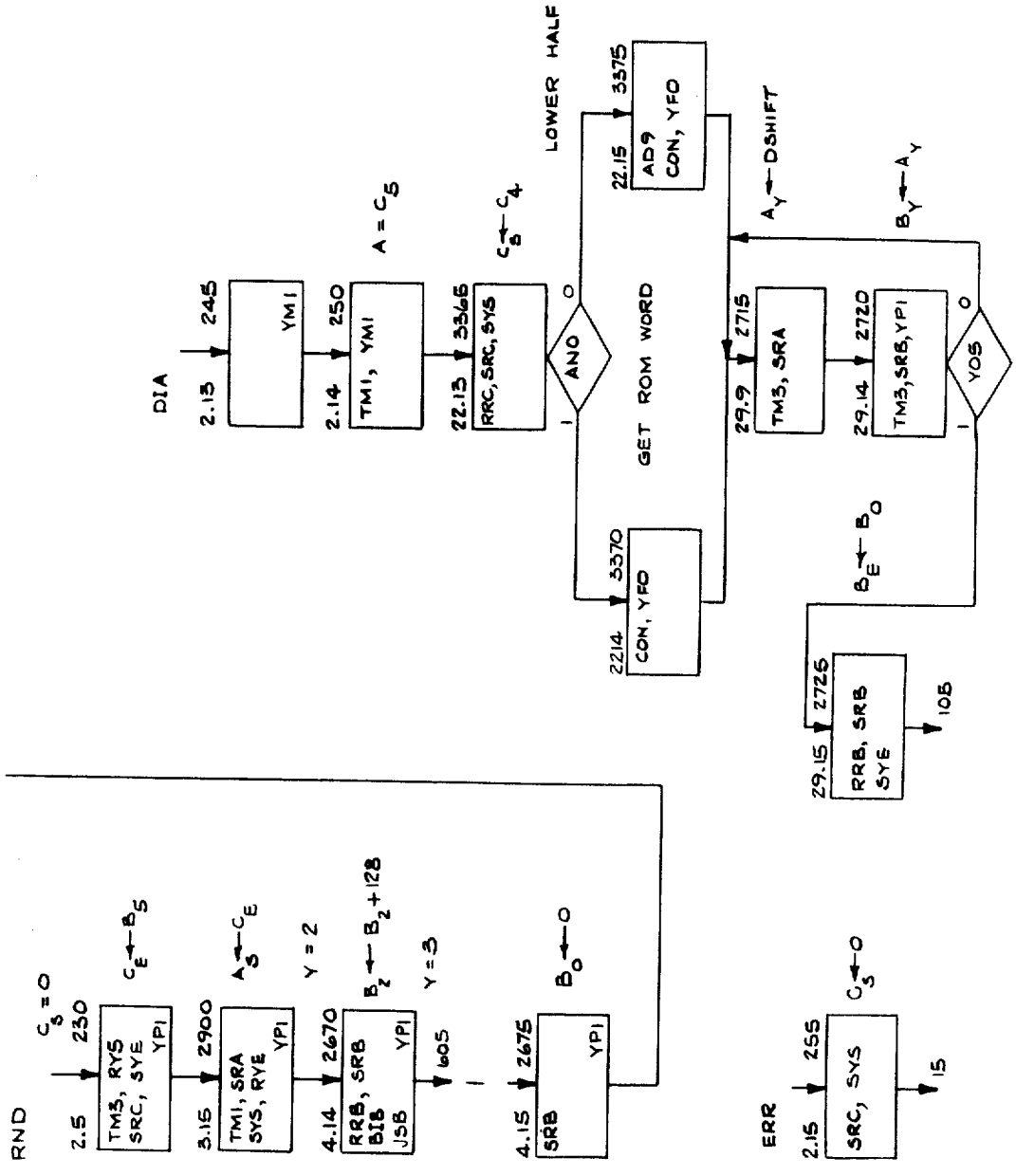

RND
$C_S = 0$
230
2.5
TM3, RYS
SRC, SYE
YP1
$C_E \leftarrow B_S$
2900
3.15
TM1, SRA
SYS, RYE
YP1
$A_S \leftarrow C_E$
Y = 2
2670
4.14
RRB, SRB
BIB
JSB
$B_2 \leftarrow B_2 + 128$
Y = 3
605
2675
4.15
SRB
YP1
$B_0 \leftarrow 0$
ERR
255
2.15
SRC, SYS
$C_S \leftarrow 0$
15
DIA
245
2.13
YM1
250
2.14
TM1, YM1
$A = C_S$
3366
22.13
RRC, SRC, SYS
$C_S \leftarrow C_4$
LOWER HALF
AND
GET ROM WORD
3375
22.15
AD9
CON, YFO
3370
2214
CON, YFO
2715
29.9
TM3, SRA
$A_Y \leftarrow$ DSHIFT
2720
29.14
TM3, SRB, YP1
$B_Y \leftarrow A_Y$
YOS
$B_E \leftarrow B_0$
2725
29.15
RRB, SRB
SYE
105

FIG. 46C

ENTRY/FIX/LOAD/LOAD I/ ROUND FLOWCHARTS

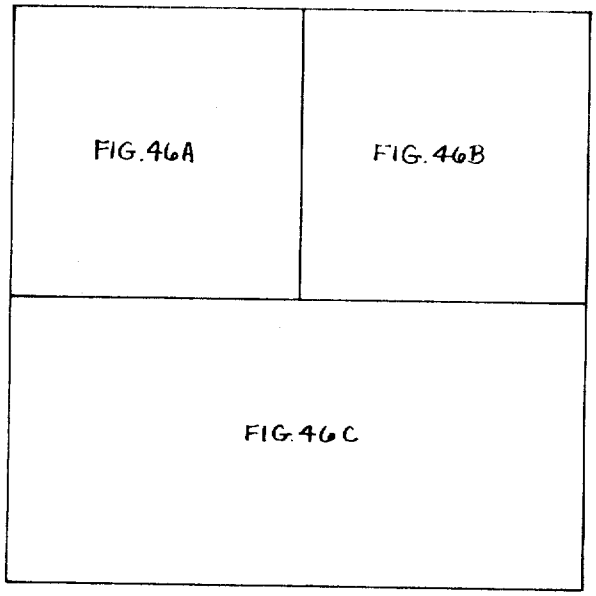



FIG. 47

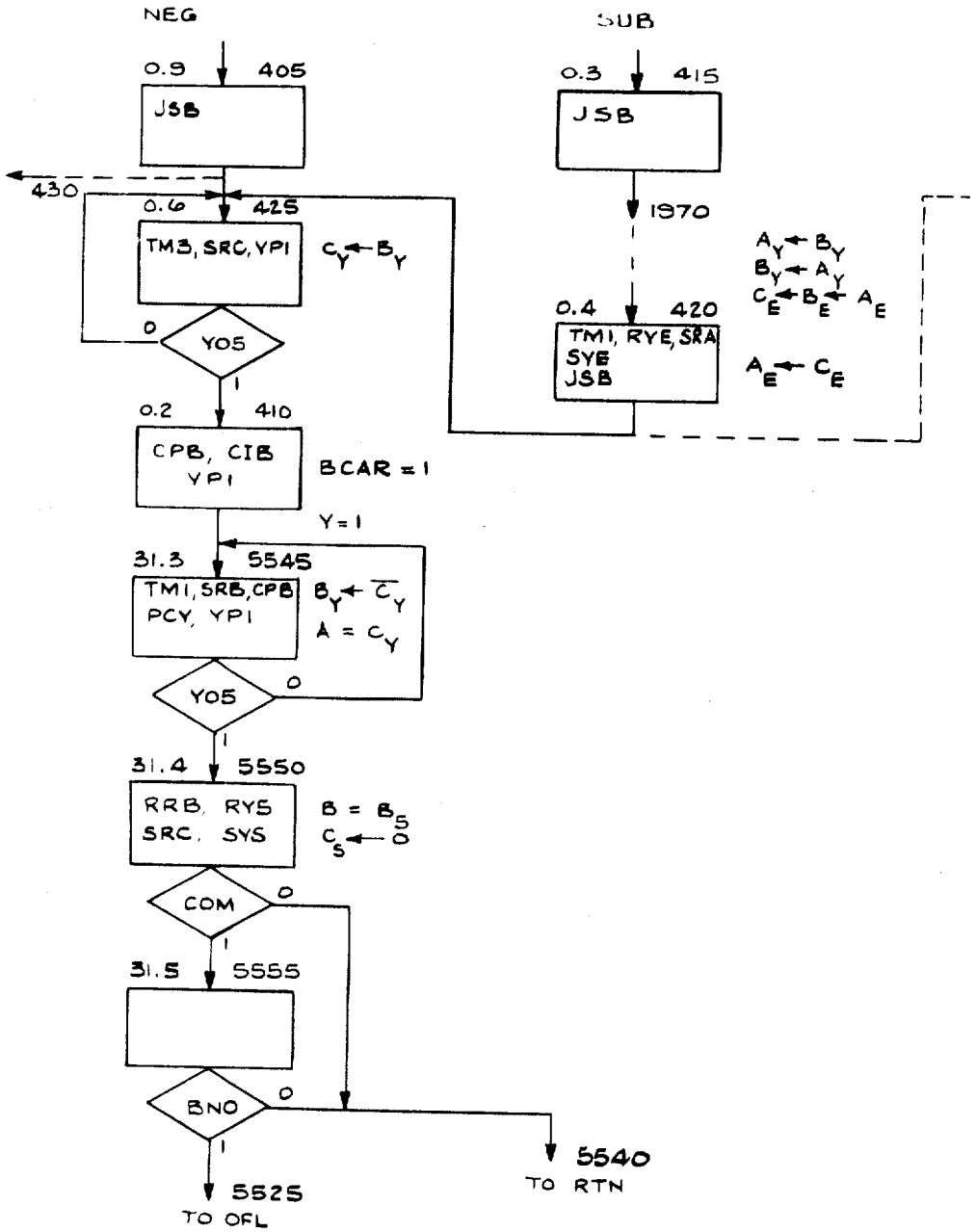

NEG
0.9
405
JSB
SUB
0.3
415
JSB
1970
430
0.6
425
TM3, SRC, YP1
$C_Y \leftarrow B_Y$
$A_Y \leftarrow B_Y$
$B_Y \leftarrow A_Y$
$C_E \leftarrow B_E \leftarrow A_E$
0.4
420
TM1, RYE, SRA
SYE
JSB
$A_E \leftarrow C_E$
0
Y05
1
0.2
410
CPB, CIB
YP1
BCAR = 1
Y = 1
31.3
5545
TM1, SRB, CPB
PCY, YP1
$B_Y \leftarrow \overline{C}_Y$
$A = C_Y$
Y05
0
1
31.4
5550
RRB, RY5
SRC, SY5
$B = B_5$
$C_5 \leftarrow 0$
COM
0
1
31.5
5555
BNO
0
1
5540
TO RTN
5525
TO OFL

FIG. 48A

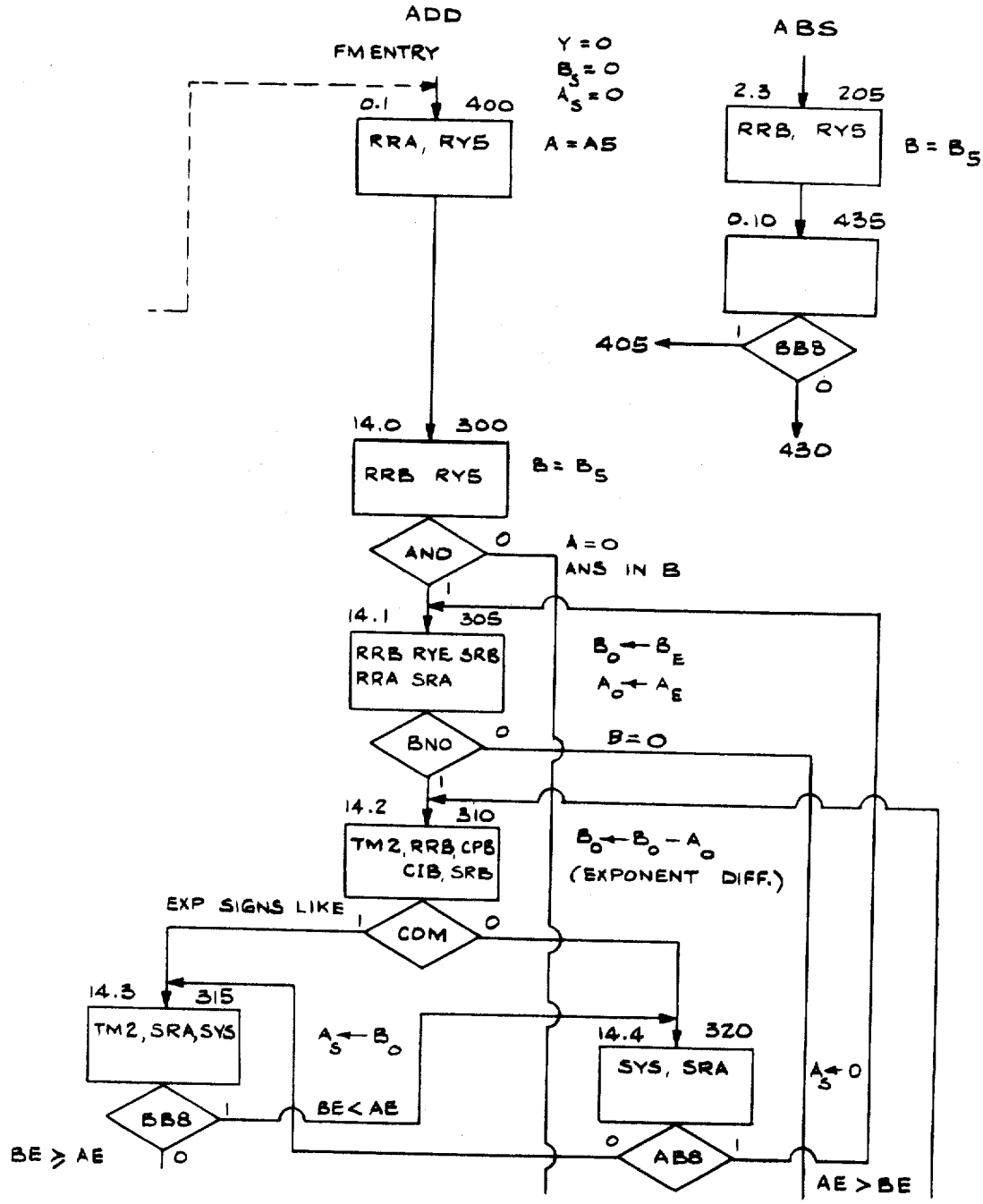

ADD
FM ENTRY
Y = 0
$B_S$ = 0
$A_S$ = 0
0.1
400
RRA, RY5
A = A5
ABS
2.3
205
RRB, RY5
B = $B_5$
0.10
435
405
BBB
1
0
430
14.0
300
RRB RY5
B = $B_5$
AND
0
A = 0
ANS IN B
1
14.1
305
RRB RYE SRB
RRA SRA
$B_0 \leftarrow B_E$
$A_0 \leftarrow A_E$
BNO
0
B = 0
1
14.2
310
TM2, RRB, CPB
CIB, SRB
$B_0 \leftarrow B_0 - A_0$
(EXPONENT DIFF.)
EXP SIGNS LIKE
1
COM
0
14.3
315
TM2, SRA, SYS
$A_S \leftarrow B_0$
14.4
320
SYS, SRA
$A_S \leftarrow 0$
BBB
1
BE < AE
BE ≥ AE
0
0
ABB
1
AE > BE

FIG. 48B

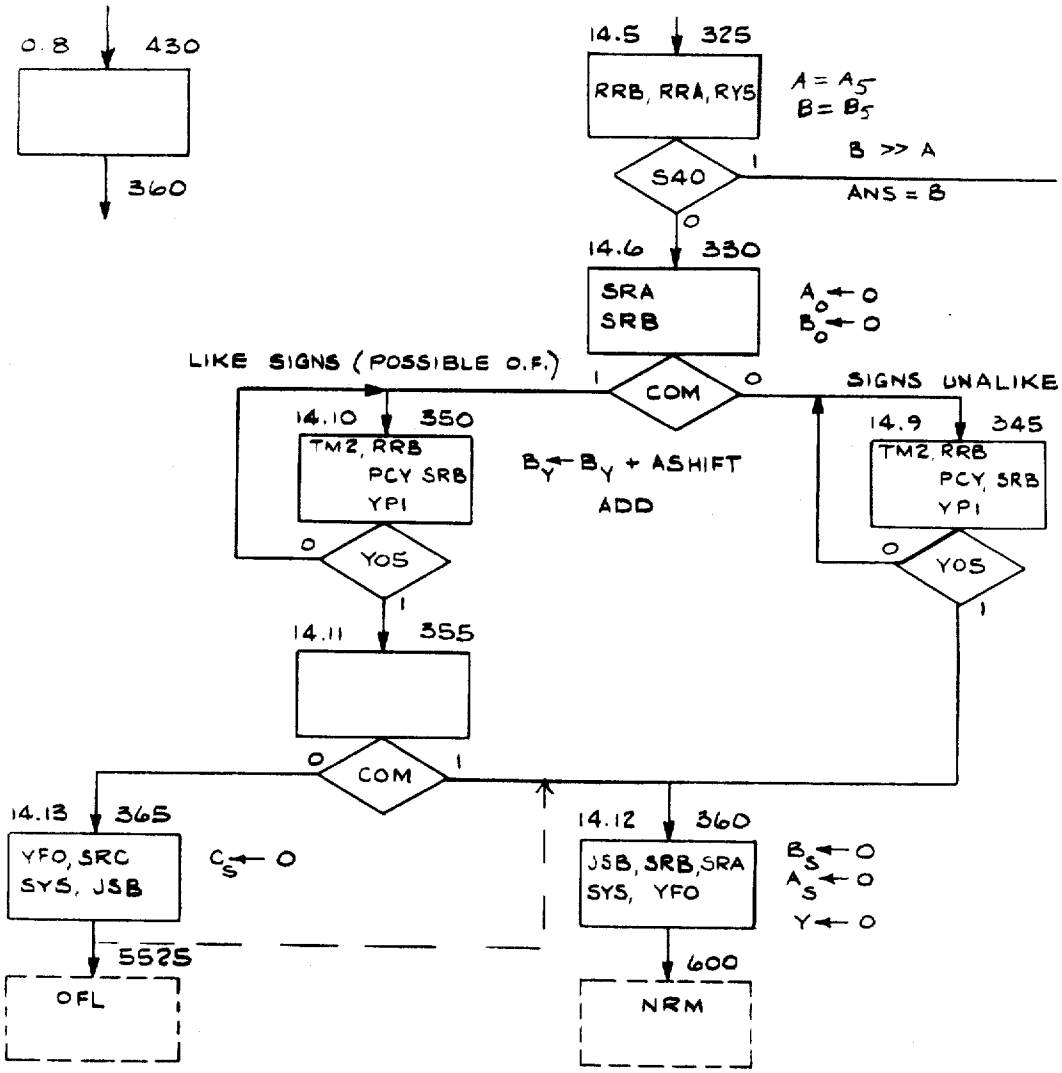

0.8
430
360
14.5
325
RRB, RRA, RYS
A = A5
B = B5
S40
B >> A
ANS = B
14.6
330
SRA
SRB
A0 ← 0
B0 ← 0
LIKE SIGNS (POSSIBLE O.F.)
COM
SIGNS UNALIKE
14.10
350
TM2, RRB
PCY SRB
YP1
BY ← BY + ASHIFT
ADD
14.9
345
TM2, RRB
PCY, SRB
YP1
YO5
YO5
14.11
355
COM
14.13
365
YFO, SRC
SYS, JSB
CS ← 0
14.12
360
JSB, SRB, SRA
SYS, YFO
BS ← 0
AS ← 0
Y ← 0
5525
OFL
600
NRM

FIG.48C

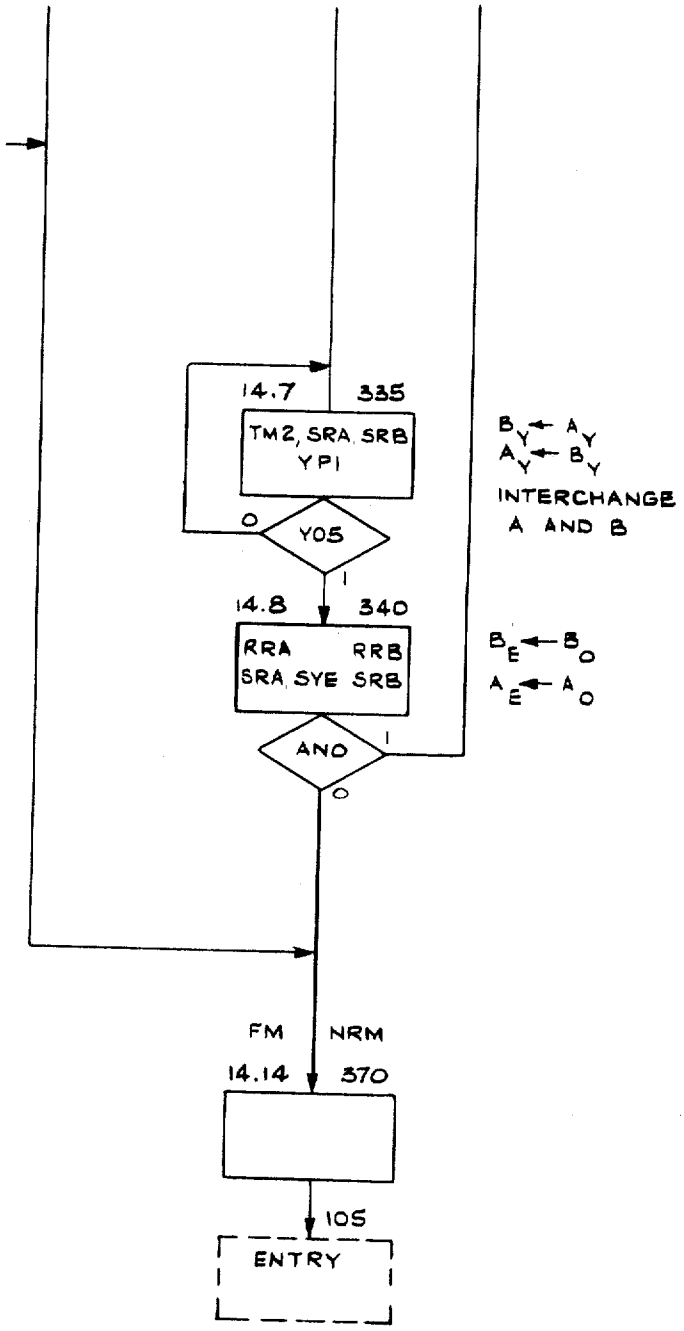

14.7
335
TM2, SRA, SRB
YP1
Y05
0
1
14.8
340
RRA RRB
SRA, SYE SRB
AN0
1
0
$B_Y \leftarrow A_Y$
$A_Y \leftarrow B_Y$
INTERCHANGE
A AND B
$B_E \leftarrow B_O$
$A_E \leftarrow A_O$
FM
NRM
14.14
370
105
ENTRY

FIG. 48D

ADD/SUB/ABS/NEG FLOWCHARTS

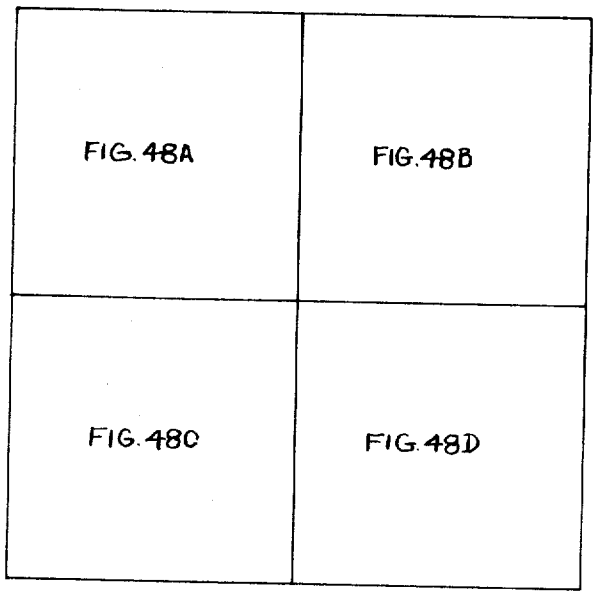



FIG. 49

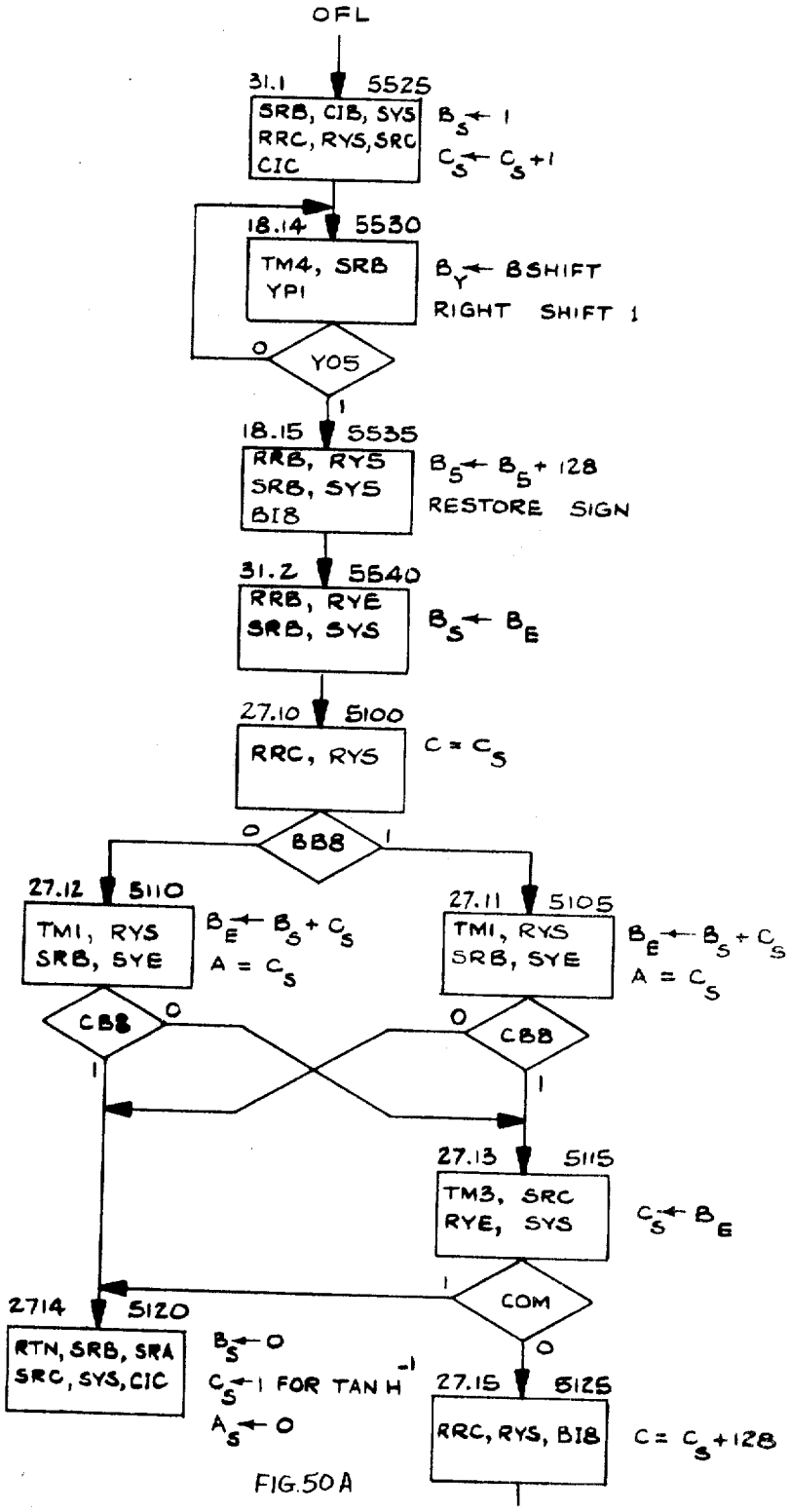

OFL
31.1 5525
SRB, CIB, SYS
RRC, RYS, SRC
CIC
$B_S \leftarrow 1$
$C_S \leftarrow C_S + 1$
18.14 5530
TM4, SRB
YPI
$B_Y \leftarrow$ BSHIFT
RIGHT SHIFT 1
0
Y05
1
18.15 5535
RRB, RYS
SRB, SYS
BI8
$B_S \leftarrow B_S + 128$
RESTORE SIGN
31.2 5540
RRB, RYE
SRB, SYS
$B_S \leftarrow B_E$
27.10 5100
RRC, RYS
$C = C_S$
0
BB8
1
27.12 5110
TM1, RYS
SRB, SYE
$B_E \leftarrow B_S + C_S$
$A = C_S$
27.11 5105
TM1, RYS
SRB, SYE
$B_E \leftarrow B_S + C_S$
$A = C_S$
CB8
0
1
0
CB8
1
27.13 5115
TM3, SRC
RYE, SYS
$C_S \leftarrow B_E$
1
COM
0
2714 5120
RTN, SRB, SRA
SRC, SYS, CIC
$B_S \leftarrow 0$
$C_S \leftarrow 1$ FOR TAN H$^{-1}$
$A_S \leftarrow 0$
27.15 5125
RRC, RYS, BI8
$C = C_S + 128$

FIG. 50A

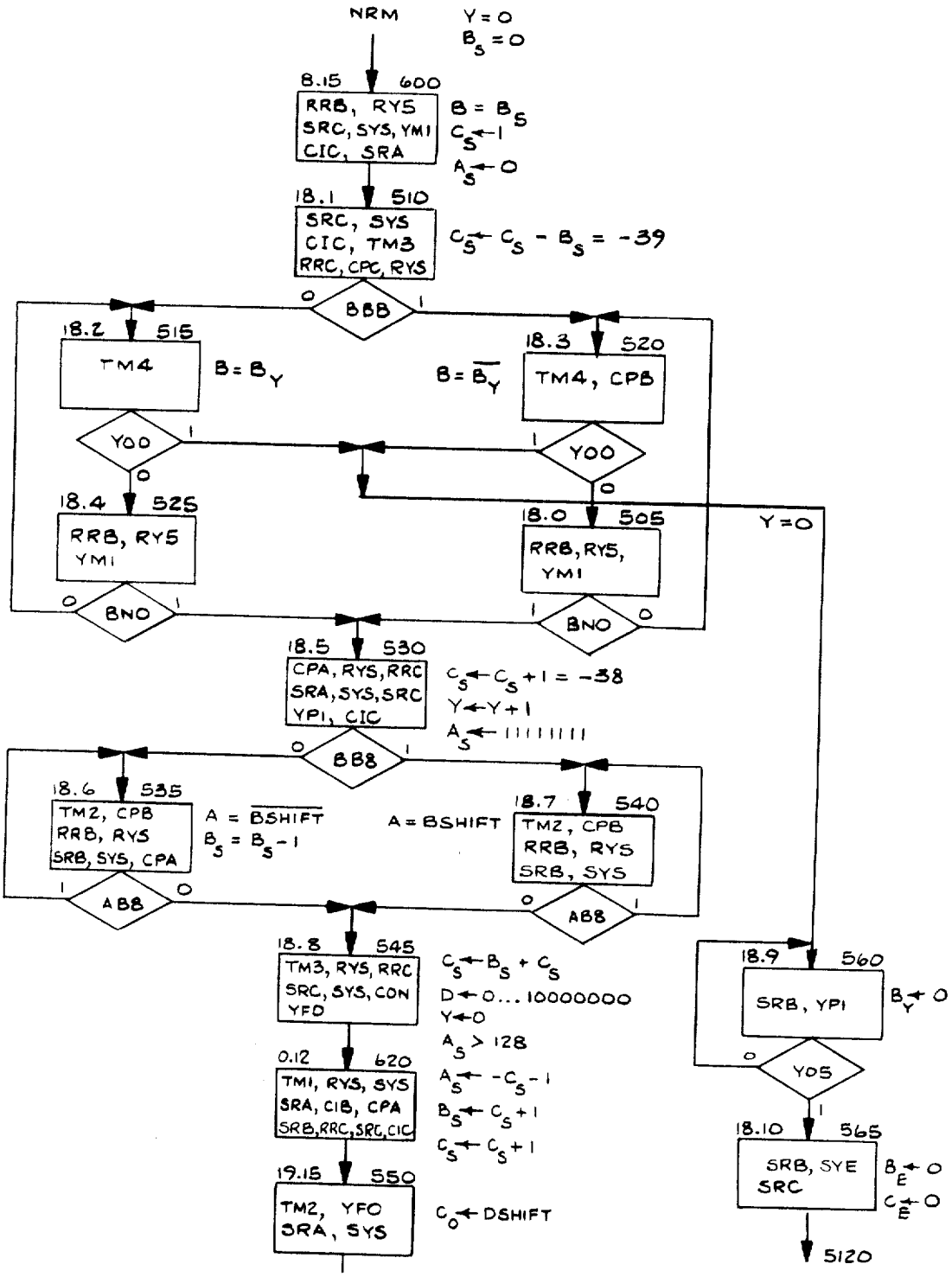

NRM
Y = 0
$B_S$ = 0
8.15 600
RRB, RY5
SRC, SYS, YM1
CIC, SRA
B = $B_S$
$C_S \leftarrow 1$
$A_S \leftarrow 0$
18.1 510
SRC, SYS
CIC, TM3
RRC, CPC, RYS
$C_S \leftarrow C_S - B_S = -39$
BBB
18.2 515
TM4
B = $B_Y$
18.3 520
TM4, CPB
B = $\overline{B_Y}$
Y00
Y00
18.4 525
RRB, RY5
YM1
18.0 505
RRB, RY5,
YM1
Y = 0
BNO
BNO
18.5 530
CPA, RYS, RRC
SRA, SYS, SRC
YP1, CIC
$C_S \leftarrow C_S + 1 = -38$
$Y \leftarrow Y + 1$
$A_S \leftarrow$ 11111111
BB8
18.6 535
TM2, CPB
RRB, RYS
SRB, SYS, CPA
A = $\overline{BSHIFT}$
$B_S = B_S - 1$
A = BSHIFT
18.7 540
TM2, CPB
RRB, RYS
SRB, SYS
ABB
ABB
18.8 545
TM3, RYS, RRC
SRC, SYS, CON
YFO
$C_S \leftarrow B_S + C_S$
D ← 0...10000000
Y ← 0
$A_S > 128$
0.12 620
TM1, RYS, SYS
SRA, CIB, CPA
SRB, RRC, SRC, CIC
$A_S \leftarrow -C_S - 1$
$B_S \leftarrow C_S + 1$
$C_S \leftarrow C_S + 1$
19.15 550
TM2, YFO
SRA, SYS
$C_0 \leftarrow$ DSHIFT
18.9 560
SRB, YP1
$B_Y \leftarrow 0$
Y05
18.10 565
SRB, SYE
SRC
$B_E \leftarrow 0$
$C_E \leftarrow 0$
5120

FIG. 50B

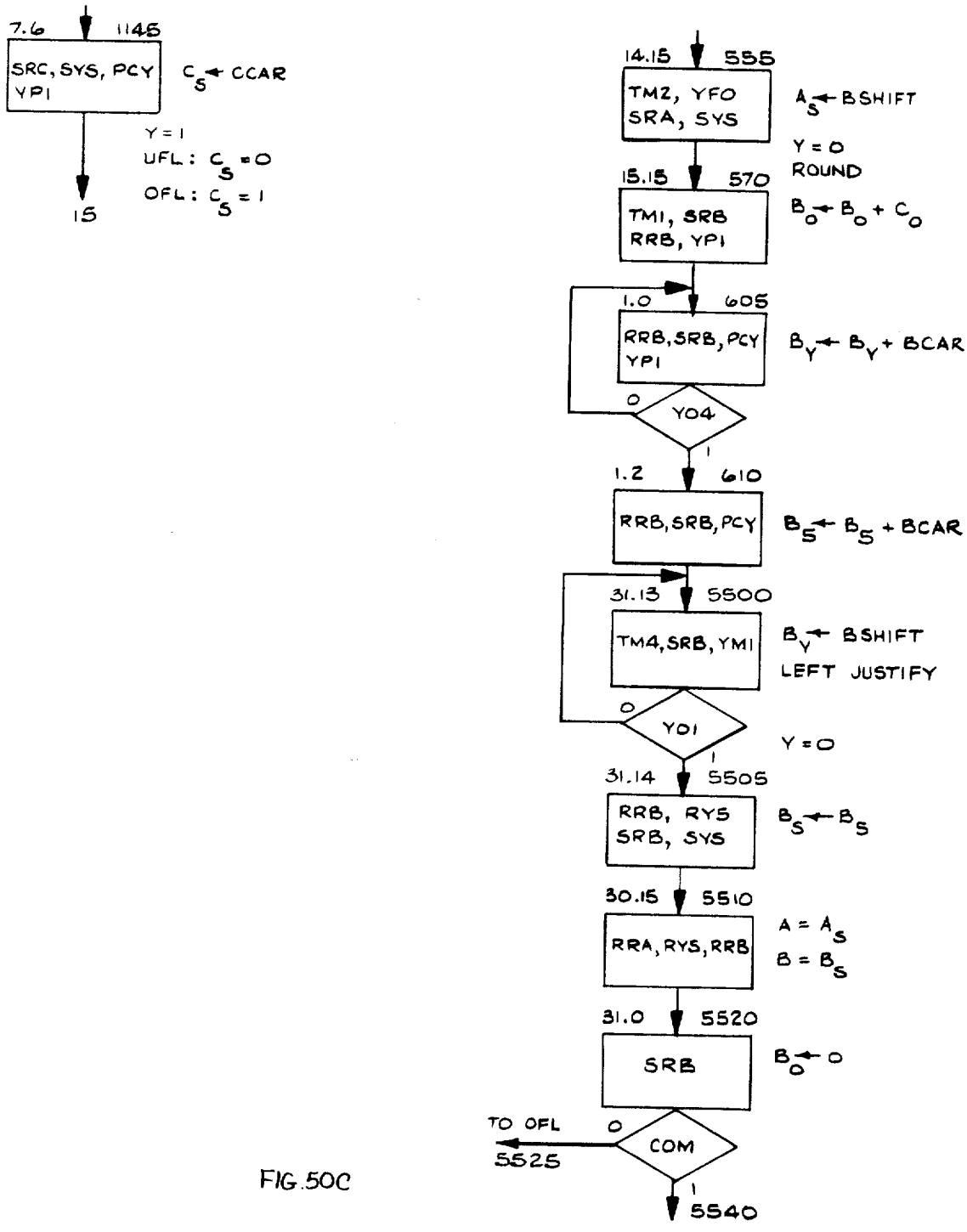

7.6
1145
SRC, SYS, PCY
YP1
$C_S \leftarrow$ CCAR
Y = 1
UFL: $C_S = 0$
OFL: $C_S = 1$
15
14.15
555
TM2, YFO
SRA, SYS
$A_S \leftarrow$ BSHIFT
Y = 0
ROUND
15.15
570
TM1, SRB
RRB, YP1
$B_O \leftarrow B_O + C_O$
1.0
605
RRB, SRB, PCY
YP1
$B_Y \leftarrow B_Y +$ BCAR
0
Y04
1
1.2
610
RRB, SRB, PCY
$B_S \leftarrow B_S +$ BCAR
31.13
5500
TM4, SRB, YM1
$B_Y \leftarrow$ BSHIFT
LEFT JUSTIFY
0
Y01
Y = 0
1
31.14
5505
RRB, RYS
SRB, SYS
$B_S \leftarrow B_S$
30.15
5510
RRA, RYS, RRB
A = $A_S$
B = $B_S$
31.0
5520
SRB
$B_O \leftarrow 0$
TO OFL
5525
0
COM
1
5540

FIG. 50C

OVERFLOW/NORMALIZE FLOWCHARTS

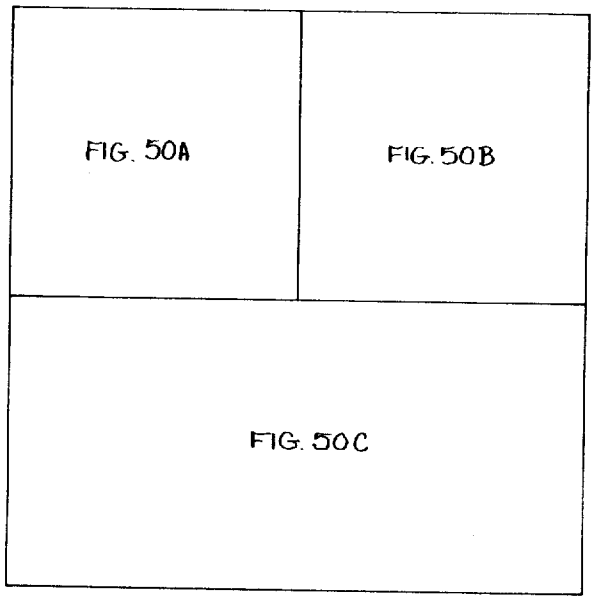



FIG. 51

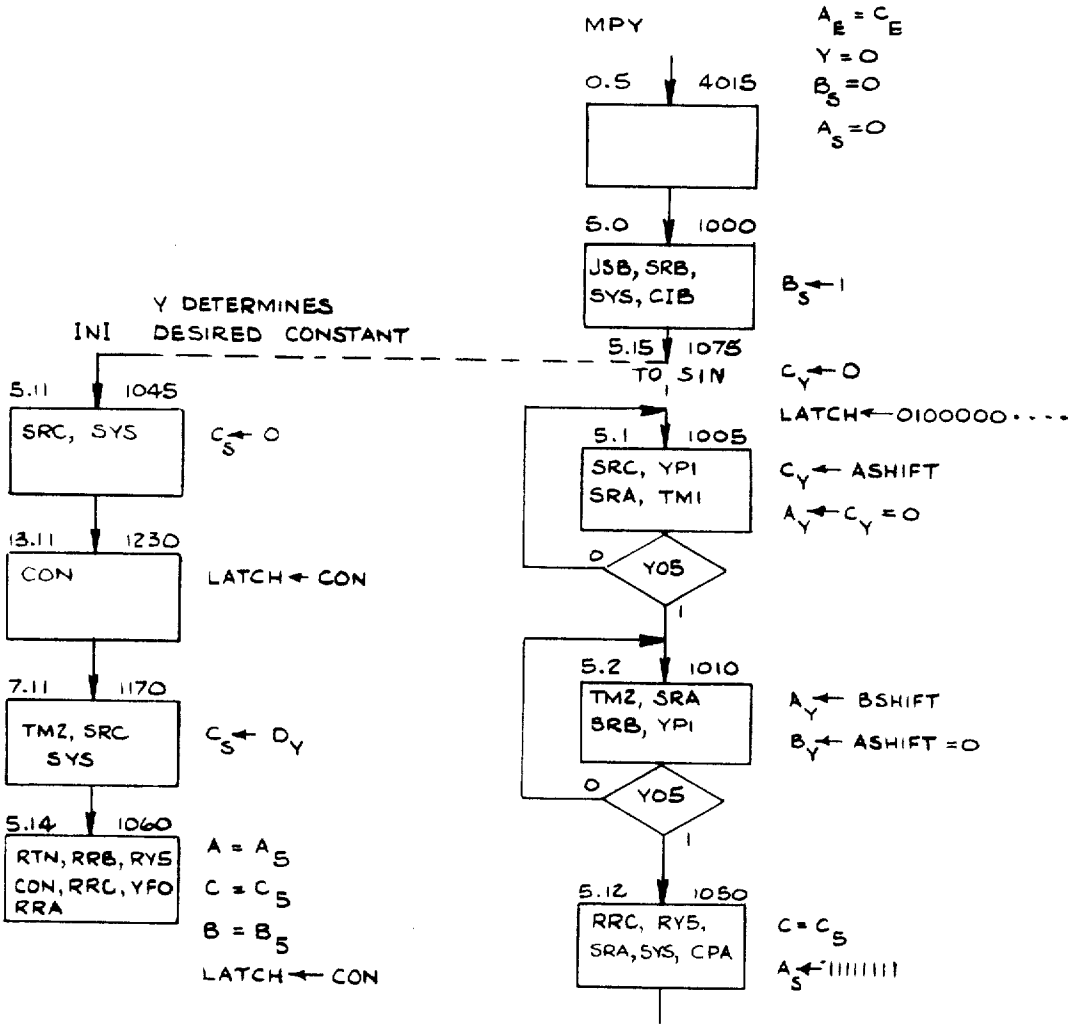

MPY
$A_E = C_E$
Y = 0
$B_S = 0$
$A_S = 0$
0.5
4015
5.0
1000
JSB, SRB, SYS, CIB
$B_S \leftarrow 1$
5.15
1075
TO SIN
$C_Y \leftarrow 0$
LATCH ← 0100000....
Y DETERMINES
INI DESIRED CONSTANT
5.11
1045
SRC, SYS
$C_S \leftarrow 0$
5.1
1005
SRC, YPI
SRA, TMI
$C_Y \leftarrow$ ASHIFT
$A_Y \leftarrow C_Y = 0$
0
Y05
1
13.11
1230
CON
LATCH ← CON
5.2
1010
TMZ, SRA
SRB, YPI
$A_Y \leftarrow$ BSHIFT
$B_Y \leftarrow$ ASHIFT = 0
0
Y05
1
7.11
1170
TMZ, SRC
SYS
$C_S \leftarrow D_Y$
5.14
1060
RTN, RRB, RYS
CON, RRC, YFO
RRA
A = $A_S$
C = $C_S$
B = $B_S$
LATCH ← CON
5.12
1050
RRC, RY5,
SRA, SYS, CPA
C = $C_S$
$A_S \leftarrow$ 11111111

FIG. 52A

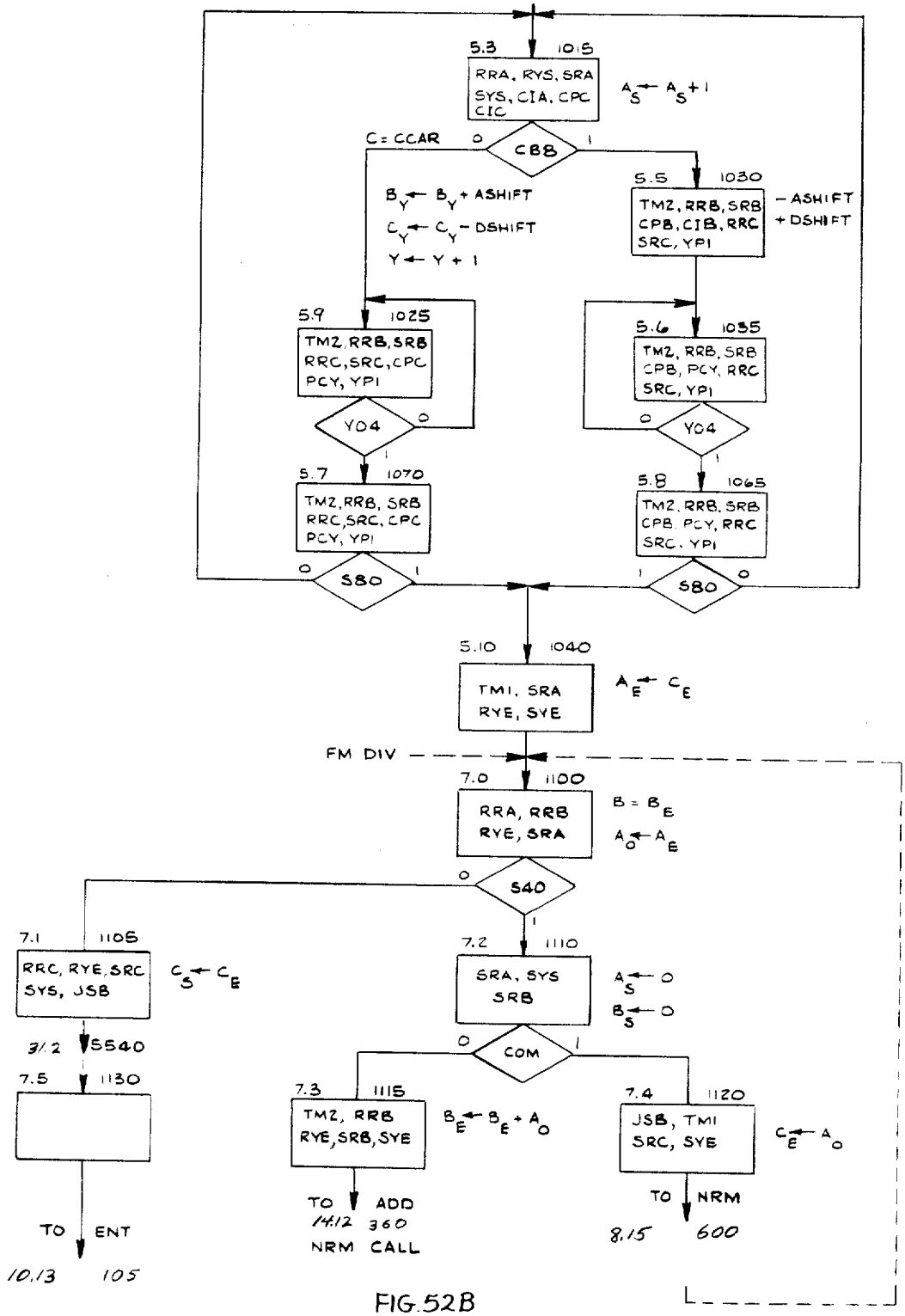

5.3 1015
RRA, RYS, SRA SYS, CIA, CPC CIC
$A_S \leftarrow A_S + 1$
C = CCAR
CBB
$B_Y \leftarrow B_Y +$ ASHIFT
$C_Y \leftarrow C_Y -$ DSHIFT
$Y \leftarrow Y + 1$
5.5 1030
TMZ, RRB, SRB CPB, CIB, RRC SRC, YPI
-ASHIFT +DSHIFT
5.9 1025
TMZ, RRB, SRB RRC, SRC, CPC PCY, YPI
YO4
5.6 1035
TMZ, RRB, SRB CPB, PCY, RRC SRC, YPI
YO4
5.7 1070
TMZ, RRB, SRB RRC, SRC, CPC PCY, YPI
5.8 1065
TMZ, RRB, SRB CPB, PCY, RRC SRC, YPI
S80
S80
5.10 1040
TMI, SRA RYE, SYE
$A_E \leftarrow C_E$
FM DIV
7.0 1100
RRA, RRB RYE, SRA
$B = B_E$
$A_O \leftarrow A_E$
S40
7.1 1105
RRC, RYE, SRC SYS, JSB
$C_S \leftarrow C_E$
3/2 S540
7.5 1130
TO ENT
10.13 105
7.2 1110
SRA, SYS SRB
$A_S \leftarrow 0$
$B_S \leftarrow 0$
COM
7.3 1115
TMZ, RRB RYE, SRB, SYE
$B_E \leftarrow B_E + A_O$
TO ADD
14.12 360
NRM CALL
7.4 1120
JSB, TMI SRC, SYE
$C_E \leftarrow A_O$
TO NRM
8.15 600

FIG. 52B

MULTIPLY/ INITIALIZE FLOWCHARTS

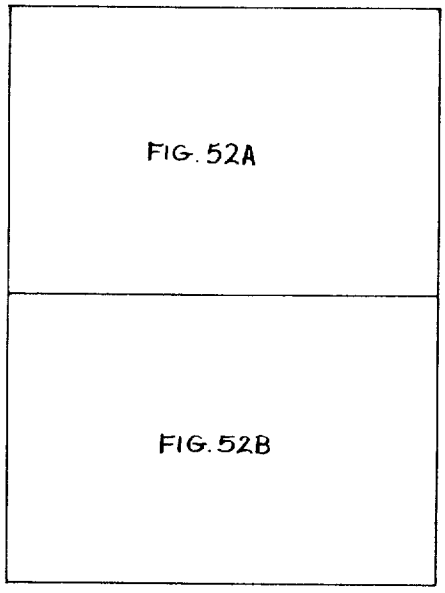



FIG. 53

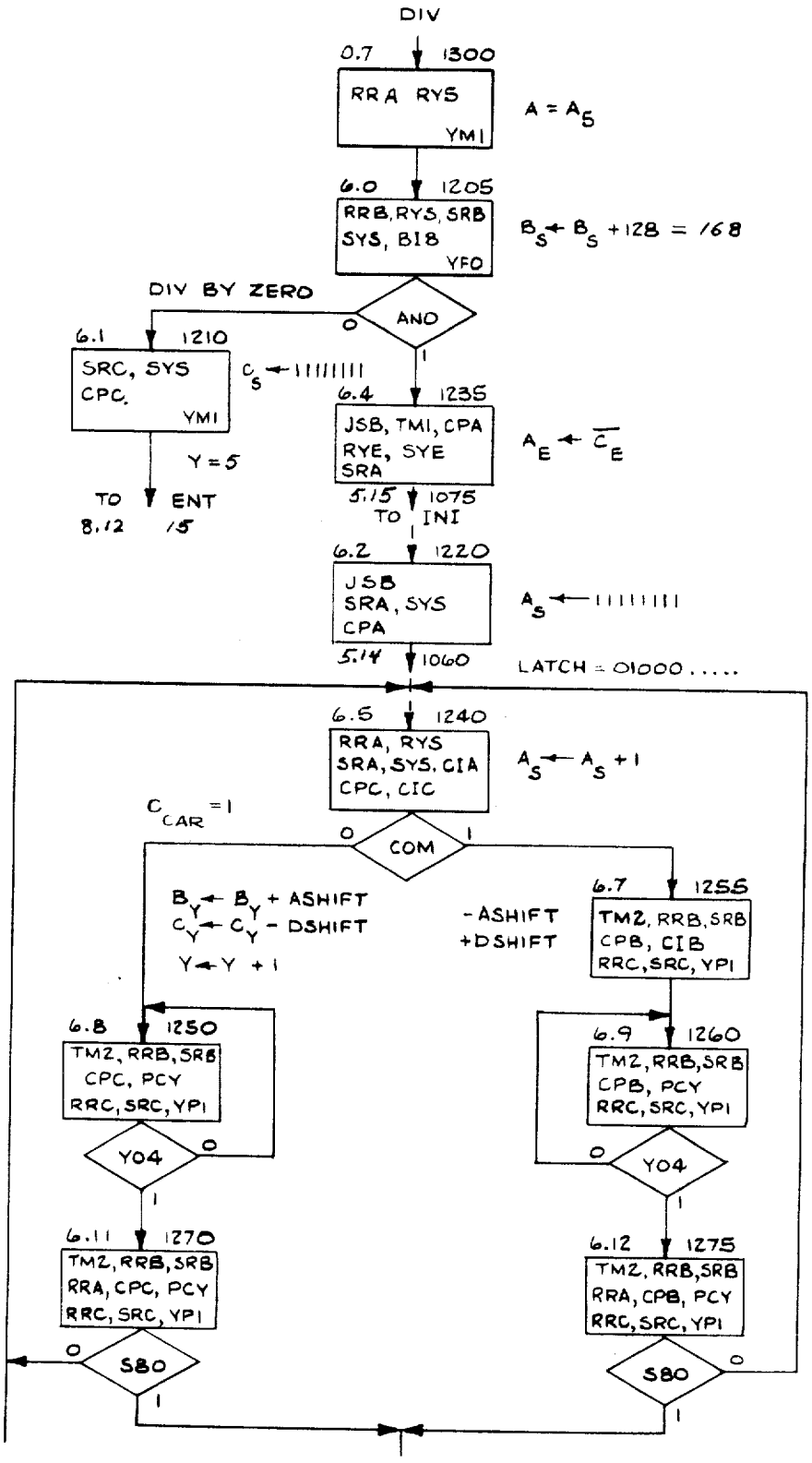

DIV
0.7 1300
RRA RYS
YM1
A = A5
6.0 1205
RRB, RYS, SRB
SYS, B18
YF0
BS ← BS + 128 = 168
DIV BY ZERO
ANO
0
1
6.1 1210
SRC, SYS
CPC.
YM1
CS ← 11111111
Y = 5
TO 8.12
ENT 15
6.4 1235
JSB, TMI, CPA
RYE, SYE
SRA
AE ← C̄E
5.15 TO 1075 INI
6.2 1220
JSB
SRA, SYS
CPA
AS ← 11111111
5.14 1060
LATCH = 01000.....
6.5 1240
RRA, RYS
SRA, SYS, CIA
CPC, CIC
AS ← AS + 1
CCAR = 1
COM
BY ← BY + ASHIFT
CY ← CY - DSHIFT
Y ← Y + 1
-ASHIFT
+DSHIFT
6.7 1255
TM2, RRB, SRB
CPB, CIB
RRC, SRC, YP1
6.8 1250
TM2, RRB, SRB
CPC, PCY
RRC, SRC, YP1
6.9 1260
TM2, RRB, SRB
CPB, PCY
RRC, SRC, YP1
Y04
6.11 1270
TM2, RRB, SRB
RRA, CPC, PCY
RRC, SRC, YP1
6.12 1275
TM2, RRB, SRB
RRA, CPB, PCY
RRC, SRC, YP1
S80

FIG. 54A

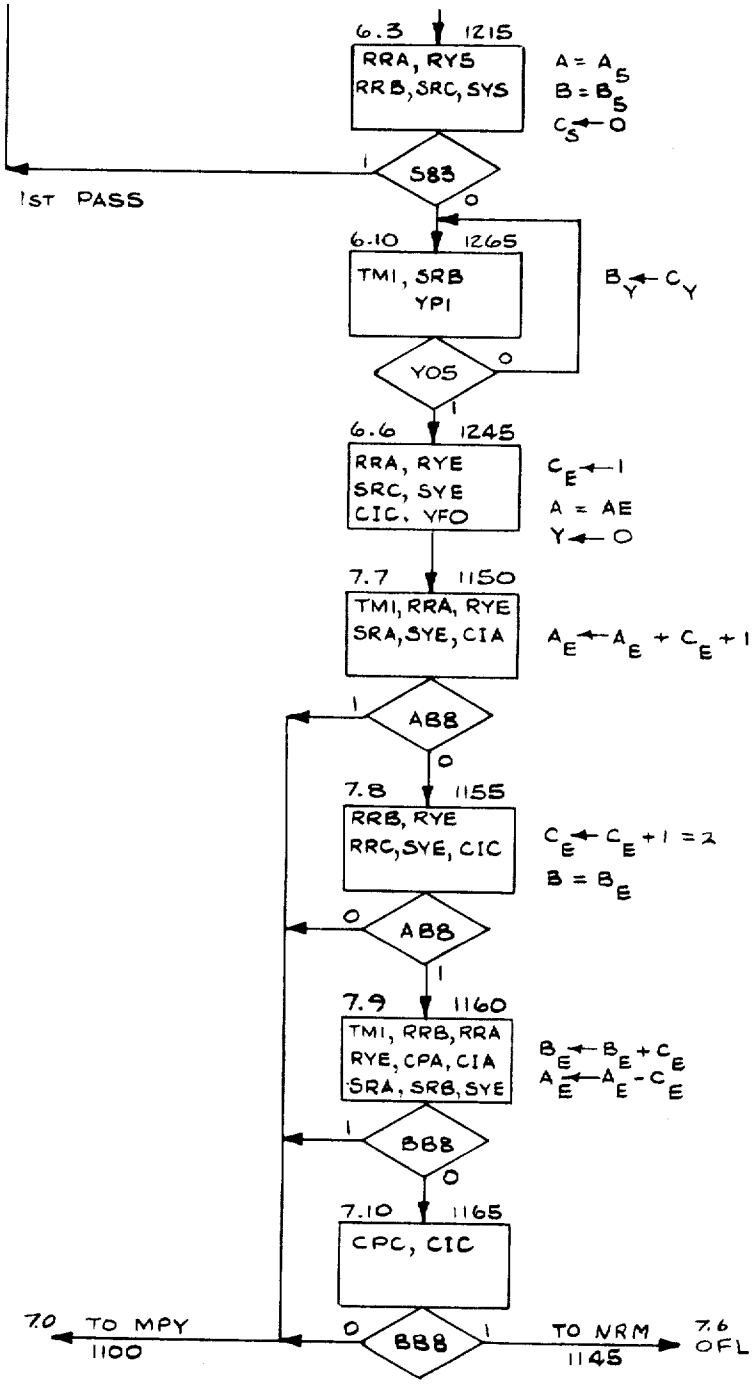

6.3 1215
RRA, RY5
RRB, SRC, SYS
A = A5
B = B5
CS ← 0
S83
1ST PASS
6.10 1265
TMI, SRB
YPI
BY ← CY
Y05
6.6 1245
RRA, RYE
SRC, SYE
CIC, YFO
CE ← 1
A = AE
Y ← 0
7.7 1150
TMI, RRA, RYE
SRA, SYE, CIA
AE ← AE + CE + 1
AB8
7.8 1155
RRB, RYE
RRC, SYE, CIC
CE ← CE + 1 = 2
B = BE
AB8
7.9 1160
TMI, RRB, RRA
RYE, CPA, CIA
SRA, SRB, SYE
BE ← BE + CE
AE ← AE - CE
BB8
7.10 1165
CPC, CIC
7.0 TO MPY 1100
BB8
TO NRM 1145
7.6 OFL

FIG. 54B

DIVISION ROUTINE FLOWCHARTS

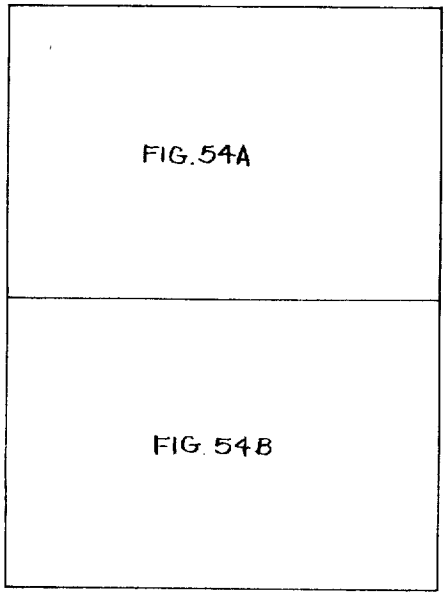



FIG. 55

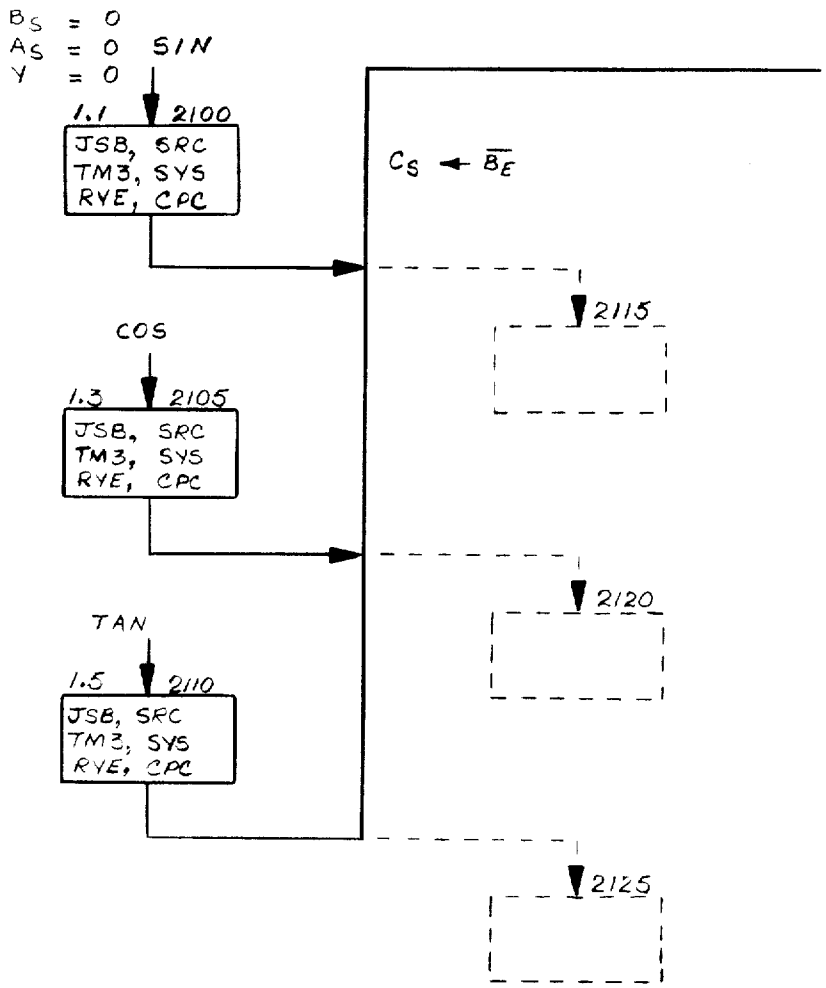

$B_S = 0$
$A_S = 0$
$Y = 0$
SIN
1.1
2100
JSB, SRC
TM3, SYS
RYE, CPC
$C_S \leftarrow \overline{B_E}$
2115
COS
1.3
2105
JSB, SRC
TM3, SYS
RYE, CPC
2120
TAN
1.5
2110
JSB, SRC
TM3, SYS
RYE, CPC
2125

FIG. 56A

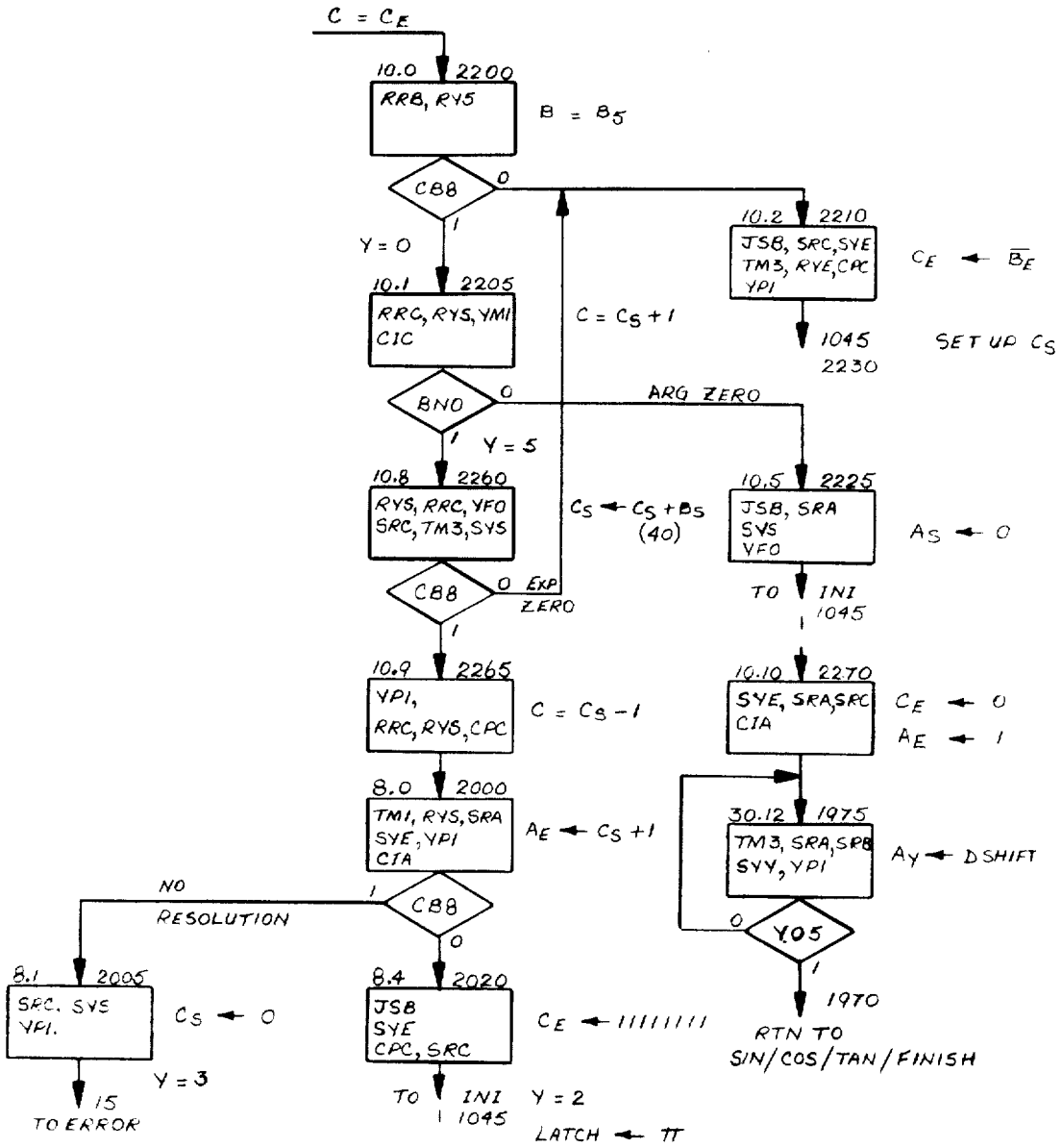

C = CE
10.0 2200
RRB, RY5
B = B5
CBB
0
1
Y = 0
10.1 2205
RRC, RYS, YMI
CIC
C = CS + 1
BNO
0
1
Y = 5
ARG ZERO
10.2 2210
JSB, SRC, SYE
TM3, RYE, CPC
YPI
CE ← B̄E
1045
2230
SET UP CS
10.8 2260
RYS, RRC, YFO
SRC, TM3, SYS
CS ← CS + B5
(40)
CBB
0 EXP ZERO
1
10.5 2225
JSB, SRA
SYS
YFO
AS ← 0
TO INI 1045
10.9 2265
YPI,
RRC, RYS, CPC
C = CS - 1
10.10 2270
SYE, SRA, SRC
CIA
CE ← 0
AE ← 1
8.0 2000
TMI, RYS, SRA
SYE, YPI
CIA
AE ← CS + 1
30.12 1975
TM3, SRA, SRB
SYY, YPI
AY ← D SHIFT
NO RESOLUTION
1
CBB
0
Y05
0
1
8.1 2005
SRC, SYS
YPI.
CS ← 0
Y = 3
15
TO ERROR
8.4 2020
JSB
SYE
CPC, SRC
CE ← 11111111
TO INI 1045
Y = 2
LATCH ← TT
1970
RTN TO
SIN/COS/TAN/FINISH

FIG. 56B

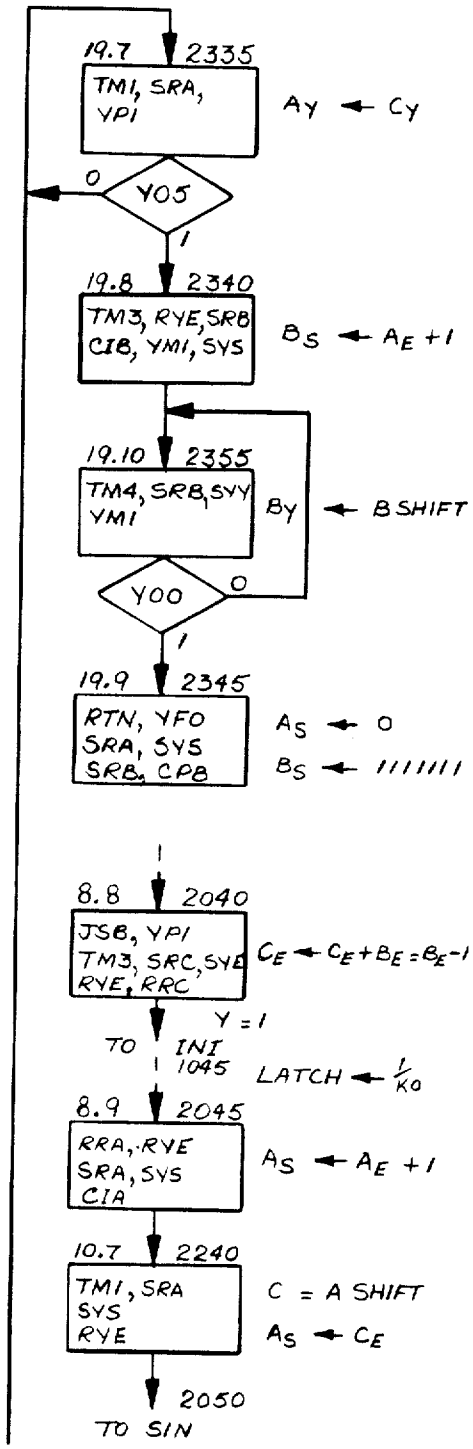

19.7
2335
TM1, SRA,
YP1
AY ← CY
0
Y05
1
19.8
2340
TM3, RYE, SRB
CIB, YM1, SYS
BS ← AE + 1
19.10
2355
TM4, SRB, SYY
YM1
BY ← B SHIFT
Y00
0
1
19.9
2345
RTN, YFO
SRA, SYS
SRB, CPB
AS ← 0
BS ← 1111111
1
8.8
2040
JSB, YP1
TM3, SRC, SYB
RYE, RRC
CE ← CE + BE = BE - 1
Y = 1
TO INI 1045
LATCH ← 1/K0
8.9
2045
RRA, RYE
SRA, SYS
CIA
AS ← AE + 1
10.7
2240
TM1, SRA
SYS
RYE
C = A SHIFT
AS ← CE
2050
TO SIN

FIG.56C

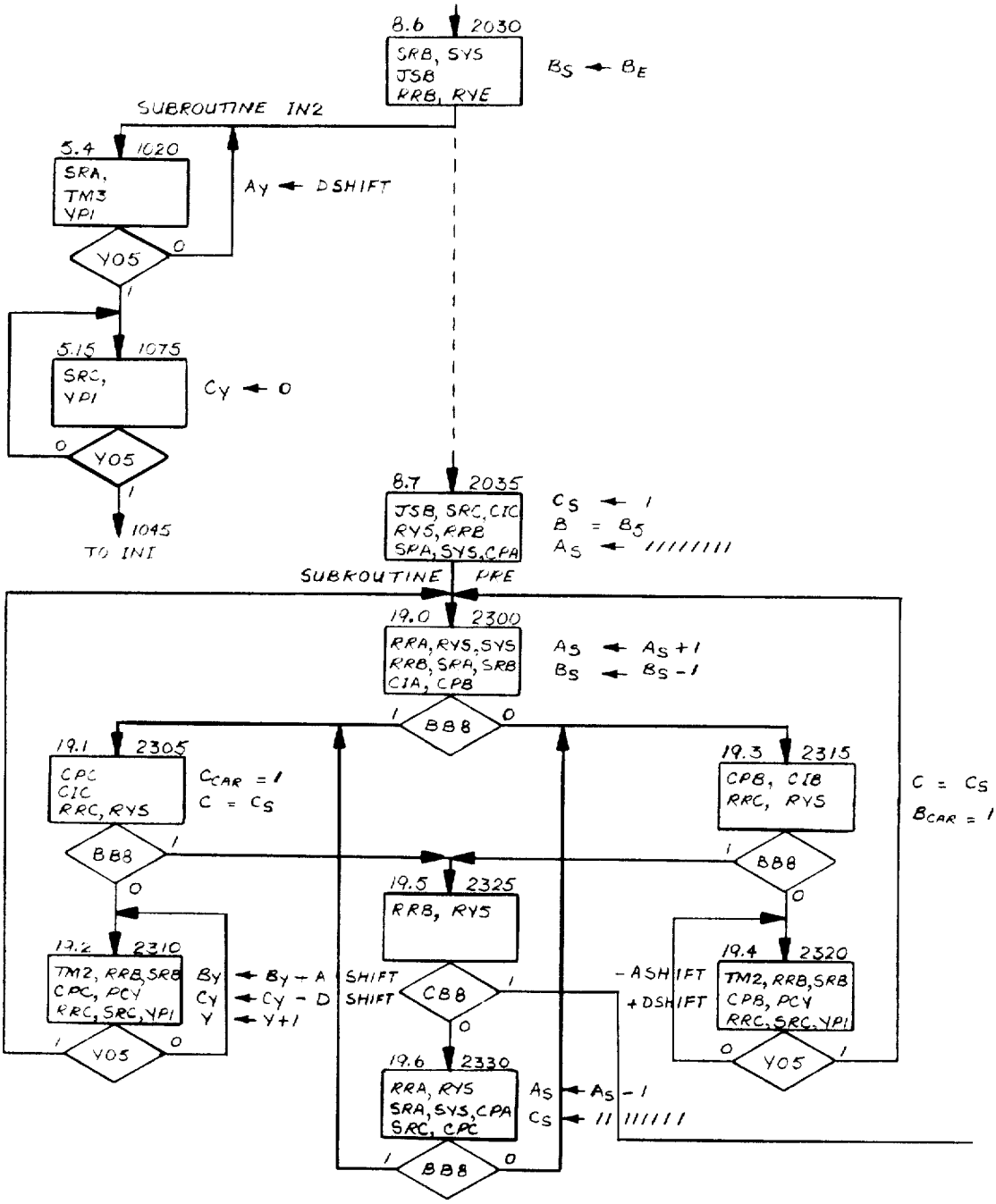

8.6 2030
SRB, SYS
JSB
RRB, RYE
BS ← BE
SUBROUTINE IN2
5.4 1020
SRA,
TM3
YP1
AY ← DSHIFT
Y05
5.15 1075
SRC,
YP1
CY ← 0
Y05
1045
TO INI
8.7 2035
JSB, SRC, CIC
RYS, RRB
SRA, SYS, CPA
CS ← 1
B = BS
AS ← ////////
SUBROUTINE PRE
19.0 2300
RRA, RYS, SYS
RRB, SRA, SRB
CIA, CPB
AS ← AS + 1
BS ← BS - 1
BBB
19.1 2305
CPC
CIC
RRC, RYS
CCAR = 1
C = CS
BBB
19.2 2310
TM2, RRB, SRB
CPC, PCY
RRC, SRC, YP1
BY ← BY - A
CY ← CY - D
Y ← Y+1
Y05
19.3 2315
CPB, CIB
RRC, RYS
C = CS
BCAR = 1
BBB
19.4 2320
TM2, RRB, SRB
CPB, PCY
RRC, SRC, YP1
-ASHIFT
+DSHIFT
Y05
19.5 2325
RRB, RYS
SHIFT
SHIFT
CBB
19.6 2330
RRA, RYS
SRA, SYS, CPA
SRC, CPC
AS ← AS - 1
CS ← 1 ////////
BBB

FIG. 56D

SIN/COS/TAN/HYPER PRESCALE FLOWCHARTS

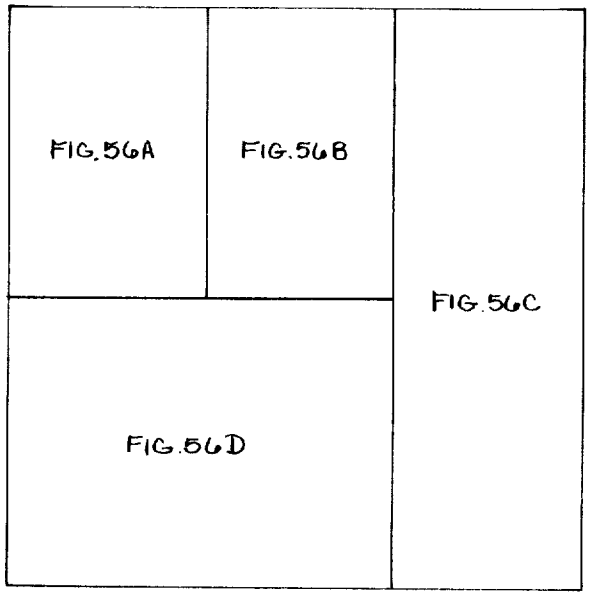



FIG.57

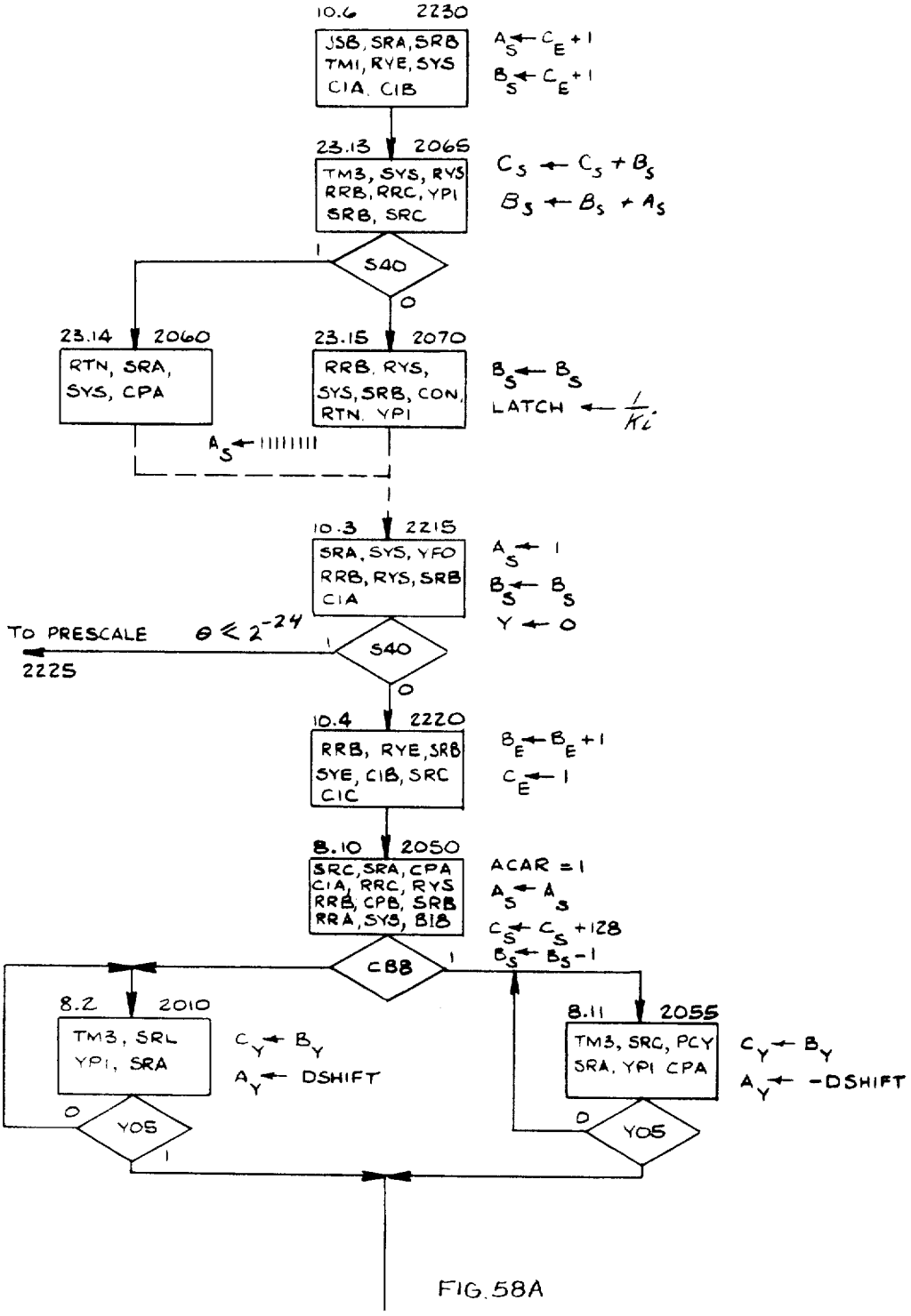

10.6 2230
JSB, SRA, SRB TMI, RYE, SYS CIA, CIB
$A_S \leftarrow C_E + 1$
$B_S \leftarrow C_E + 1$
23.13 2065
TM3, SYS, RYS RRB, RRC, YPI SRB, SRC
$C_S \leftarrow C_S + B_S$
$B_S \leftarrow B_S + A_S$
S40
1
0
23.14 2060
RTN, SRA, SYS, CPA
23.15 2070
RRB, RYS, SYS, SRB, CON, RTN, YPI
$B_S \leftarrow B_S$
LATCH $\leftarrow \frac{1}{Kc}$
$A_S \leftarrow$ !!!!!!!!
10.3 2215
SRA, SYS, YFO RRB, RYS, SRB CIA
$A_S \leftarrow 1$
$B_S \leftarrow B_S$
$Y \leftarrow 0$
TO PRESCALE
$\theta \ll 2^{-24}$
2225
S40
1
0
10.4 2220
RRB, RYE, SRB SYE, CIB, SRC CIC
$B_E \leftarrow B_E + 1$
$C_E \leftarrow 1$
8.10 2050
SRC, SRA, CPA CIA, RRC, RYS RRB, CPB, SRB RRA, SYS, B18
ACAR = 1
$A_S \leftarrow A_S$
$C_S \leftarrow C_S + 128$
$B_S \leftarrow B_S - 1$
C88
1
8.2 2010
TM3, SRL YPI, SRA
$C_Y \leftarrow B_Y$
$A_Y \leftarrow$ DSHIFT
YOS
0
1
8.11 2055
TM3, SRC, PCY SRA, YPI CPA
$C_Y \leftarrow B_Y$
$A_Y \leftarrow -$DSHIFT
YOS
0

FIG. 58A

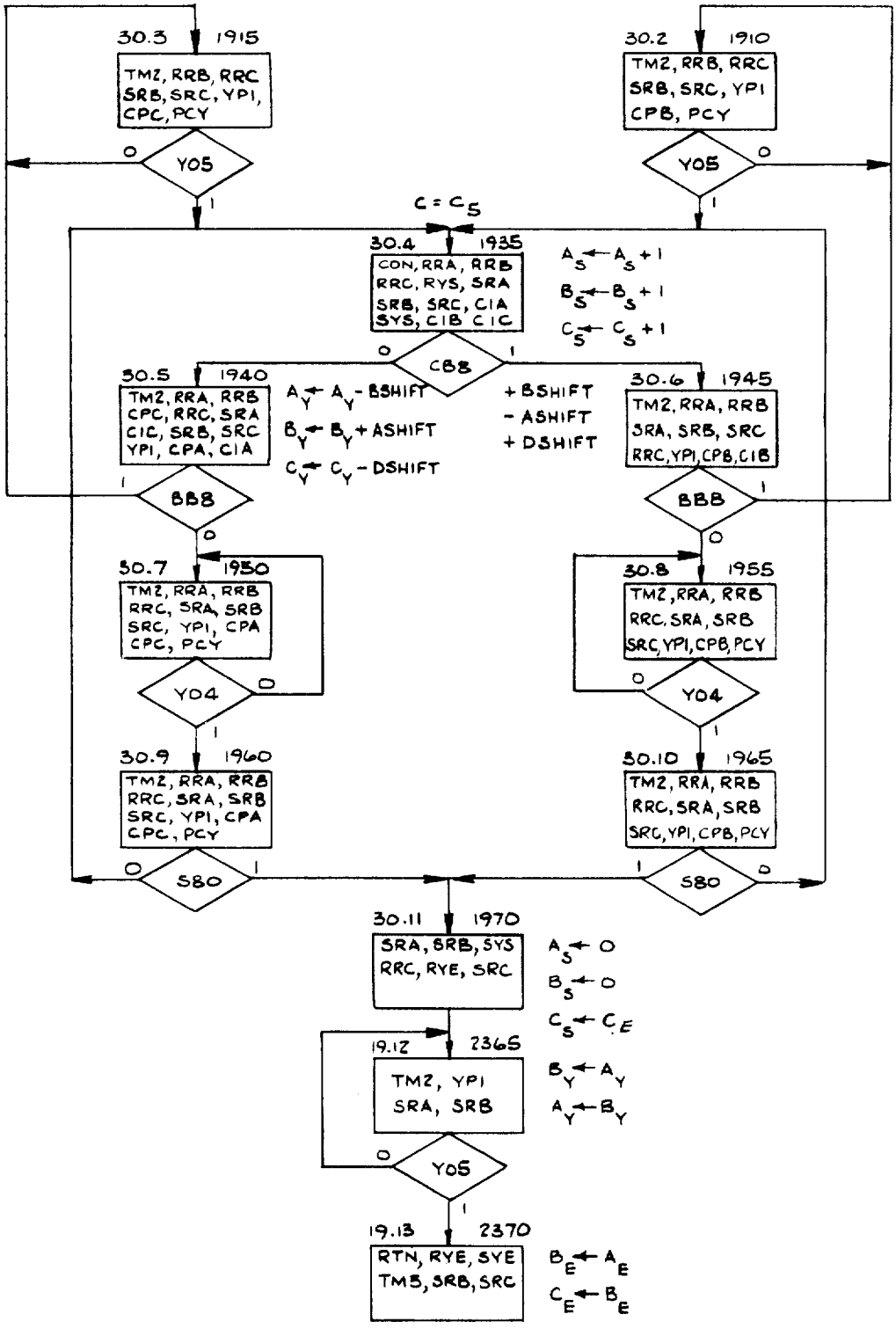
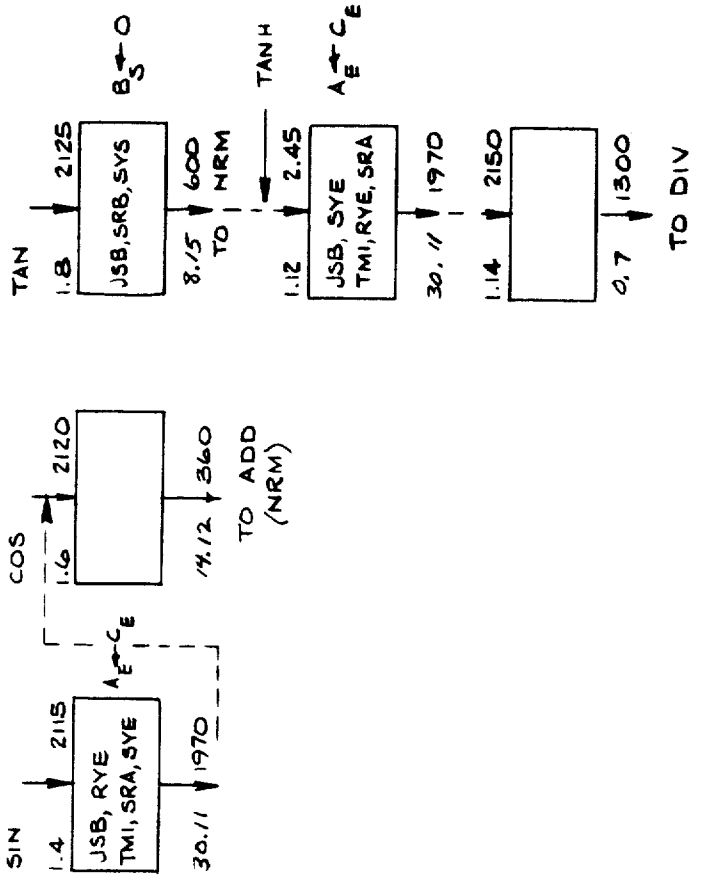

30.3 1915
TM2, RRB, RRC
SRB, SRC, YP1,
CPC, PCY
Y05
30.2 1910
TM2, RRB, RRC
SRB, SRC, YP1
CPB, PCY
Y05
C = C5
30.4 1935
CON, RRA, RRB
RRC, RYS, SRA
SRB, SRC, C1A
SYS, C1B C1C
AS ← AS + 1
BS ← BS + 1
CS ← CS + 1
CBB
30.5 1940
TM2, RRA, RRB
CPC, RRC, SRA
C1C, SRB, SRC
YP1, CPA, C1A
AY ← AY - BSHIFT
BY ← BY + ASHIFT
CY ← CY - DSHIFT
+BSHIFT
-ASHIFT
+DSHIFT
30.6 1945
TM2, RRA, RRB
SRA, SRB, SRC
RRC, YP1, CPB, C1B
BBB
BBB
30.7 1950
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPA
CPC, PCY
30.8 1955
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPB, PCY
Y04
Y04
30.9 1960
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPA
CPC, PCY
30.10 1965
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPB, PCY
S80
S80
30.11 1970
SRA, SRB, SYS
RRC, RYE, SRC
AS ← 0
BS ← 0
CS ← CE
19.12 2365
TM2, YP1
SRA, SRB
BY ← AY
AY ← BY
Y05
19.13 2370
RTN, RYE, SYE
TM5, SRB, SRC
BE ← AE
CE ← BE

FIG. 58B


TAN
1.8
2125
JSB, SRB, SYS
$B_S \leftarrow 0$
600
8.15
TO NRM
TANH
2.45
1.12
JSB, SYE
TMI, RYE, SRA
$A_E \leftarrow C_E$
1970
30.11
2150
1.14
1300
0.7
TO DIV
COS
1.6
2120
360
14.12
TO ADD
(NRM)
SIN
1.4
2115
JSB, RYE
TMI, SRA, SYE
$A_E \leftarrow C_E$
1970
30.11

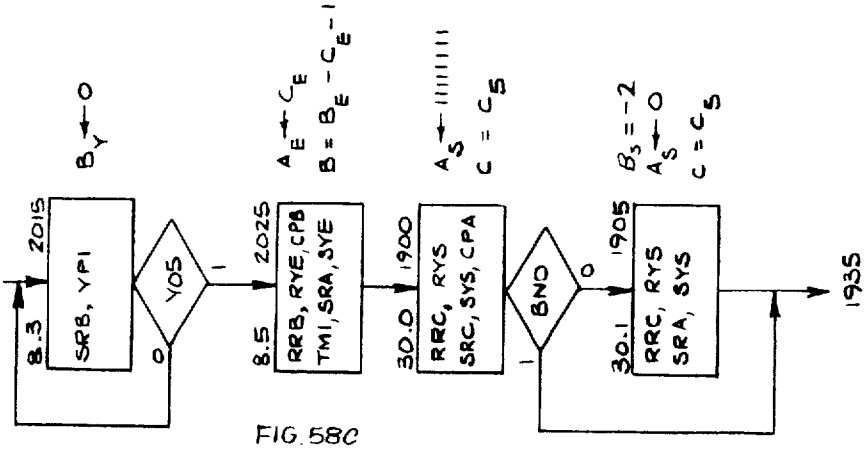

$B_Y \leftarrow 0$
2015
8.3
SRB, YPI
YOS
0
1
2025
8.5
RRB, RYE, CPB
TMI, SRA, SYE
$A_E \leftarrow C_E$
$B = B_E - C_{E-1}$
1900
30.0
RRC, RYS
SRC, SYS, CPA
$A_S \leftarrow$ !!!!!!!!
$C = C_S$
BND
0
1
1905
30.1
RRC, RYS
SRA, SYS
$B_S = -2$
$A_S \leftarrow 0$
$C = C_S$
1935

FIG. 58C

SIN RESOLVER FLOWCHARTS

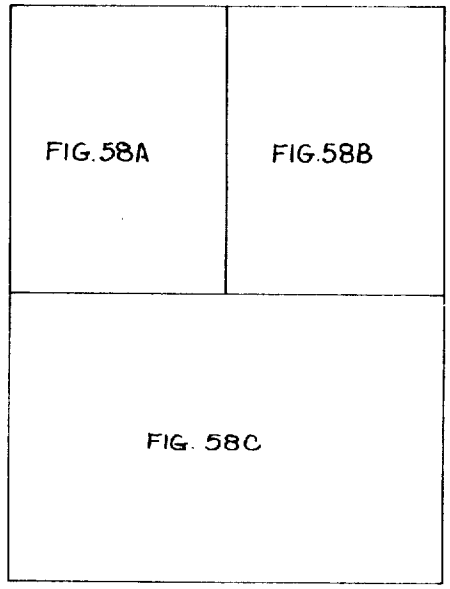



FIG. 59

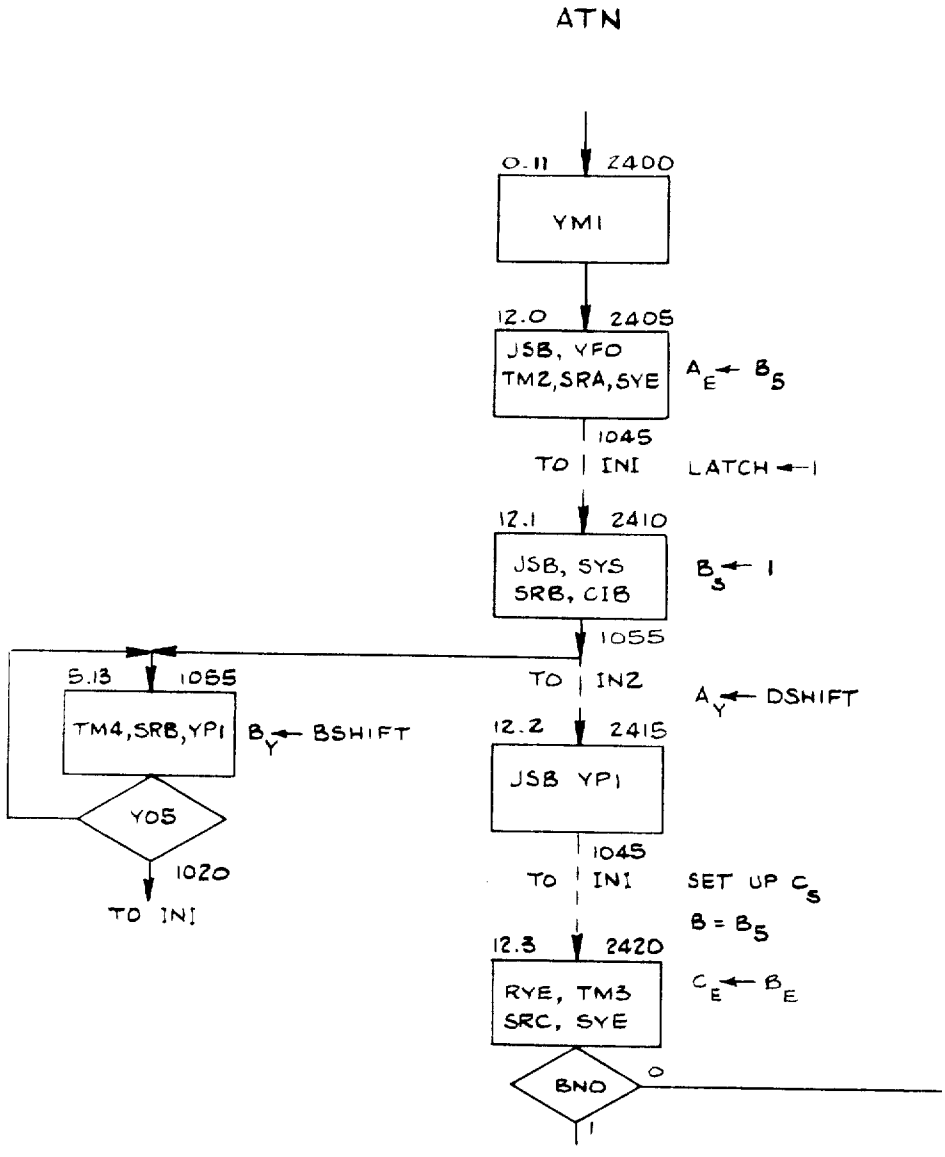

ATN
0.11
2400
YM1
12.0
2405
JSB, YF0
TM2,SRA,SYE
$A_E \leftarrow B_5$
1045
TO | INI
LATCH ← 1
12.1
2410
JSB, SYS
SRB, CIB
$B_5 \leftarrow 1$
1055
TO | IN2
$A_Y \leftarrow$ DSHIFT
12.2
2415
JSB YP1
1045
TO | IN1
SET UP $C_5$
$B = B_5$
12.3
2420
RYE, TM3
SRC, SYE
$C_E \leftarrow B_E$
BNO
0
1
5.13
1055
TM4,SRB,YP1
$B_Y \leftarrow$ BSHIFT
Y05
1020
TO IN1

FIG. 60A

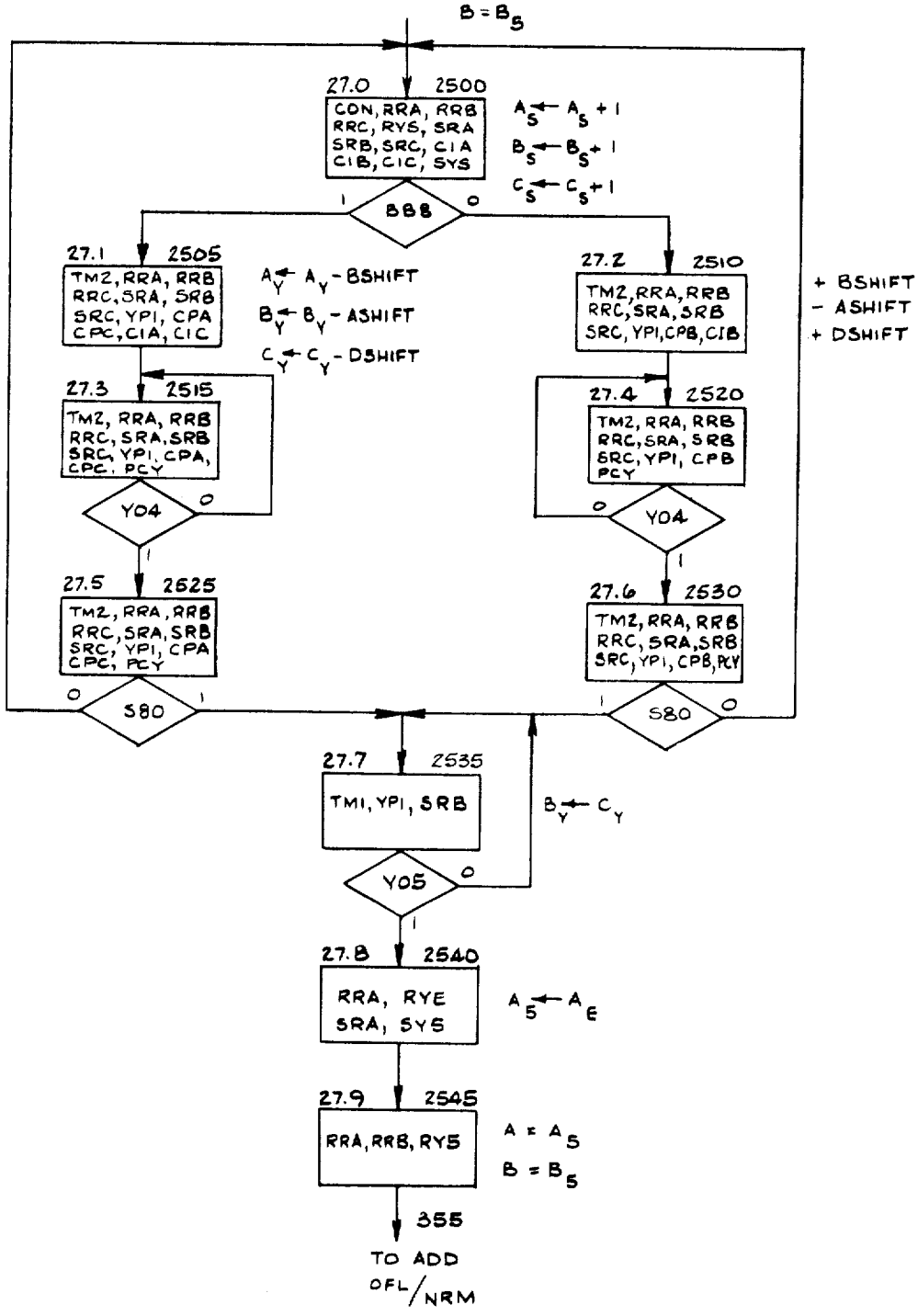

B = B5
27.0 2500
CON, RRA, RRB
RRC, RYS, SRA
SRB, SRC, C1A
C1B, C1C, SYS
A5 ← A5 + 1
B5 ← B5 + 1
C5 ← C5 + 1
BBB
1
0
27.1 2505
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPA
CPC, C1A, C1C
AY ← AY - BSHIFT
BY ← BY - ASHIFT
CY ← CY - DSHIFT
27.2 2510
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPB, C1B
+ BSHIFT
- ASHIFT
+ DSHIFT
27.3 2515
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPA,
CPC, PCY
Y04
0
1
27.4 2520
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPB
PCY
Y04
0
1
27.5 2625
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPA
CPC, PCY
S80
0
1
27.6 2530
TM2, RRA, RRB
RRC, SRA, SRB
SRC, YP1, CPB, PCY
S80
1
0
27.7 2535
TM1, YP1, SRB
BY ← CY
Y05
0
1
27.8 2540
RRA, RYE
SRA, SYS
A5 ← AE
27.9 2545
RRA, RRB, RY5
A = A5
B = B5
355
TO ADD
OFL/NRM

FIG. 60B

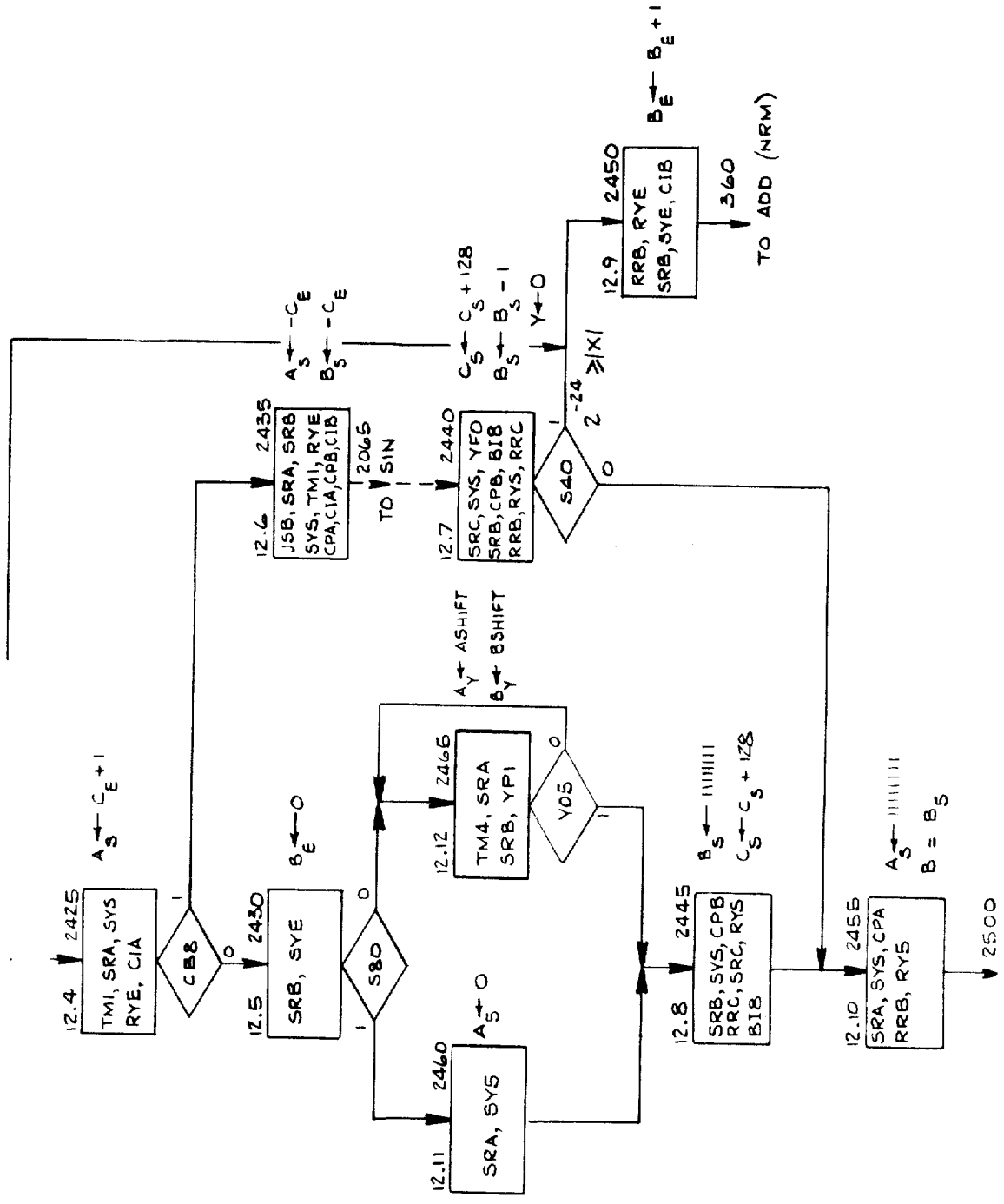

12.4 2425
TM1, SRA, SYS
RYE, C1A
$A_S \leftarrow C_E + 1$
CBB
12.5 2430
SRB, SYE
$B_E \leftarrow 0$
SB0
12.6 2435
JSB, SRA, SRB
SYS, TM1, RYE
CPA, C1A, CPB, C1B
2065
TO SIN
12.7 2440
SRC, SYS, YF0
SRB, CPB, B1B
RRB, RYS, RRC
S40
$2^{-24} \geq |X|$
$A_S \leftarrow -C_E$
$B_S \leftarrow -C_E$
$C_S \leftarrow C_S + 128$
$B_S \leftarrow B_S - 1$
$Y \leftarrow 0$
12.9 2450
RRB, RYE
SRB, SYE, C1B
$B_E \leftarrow B_E + 1$
360
TO ADD (NRM)
$A_Y \leftarrow$ ASHIFT
$B_Y \leftarrow$ BSHIFT
12.12 2465
TM4, SRA
SRB, YP1
Y05
12.11 2460
SRA, SYS
$A_S \leftarrow 0$
12.8 2445
SRB, SYS, CPB
RRC, SRC, RYS
B1B
$B_S \leftarrow$ !!!!!!!!
$C_S \leftarrow C_S + 128$
12.10 2455
SRA, SYS, CPA
RRB, RYS
$A_S \leftarrow$ !!!!!!!!
$B = B_S$
2500

FIG. 60 C

ARCTAN ROUTINE FLOWCHARTS

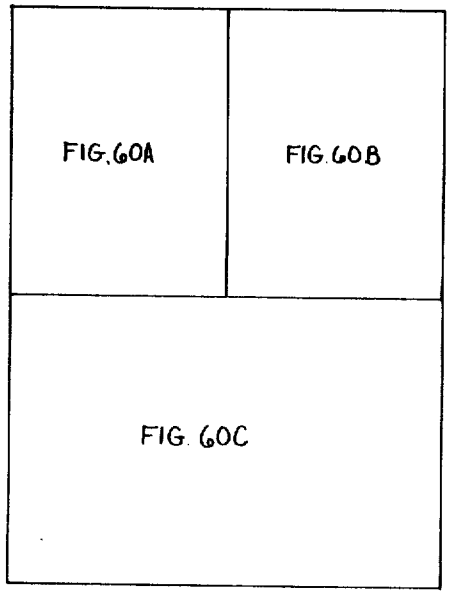



FIG. 61

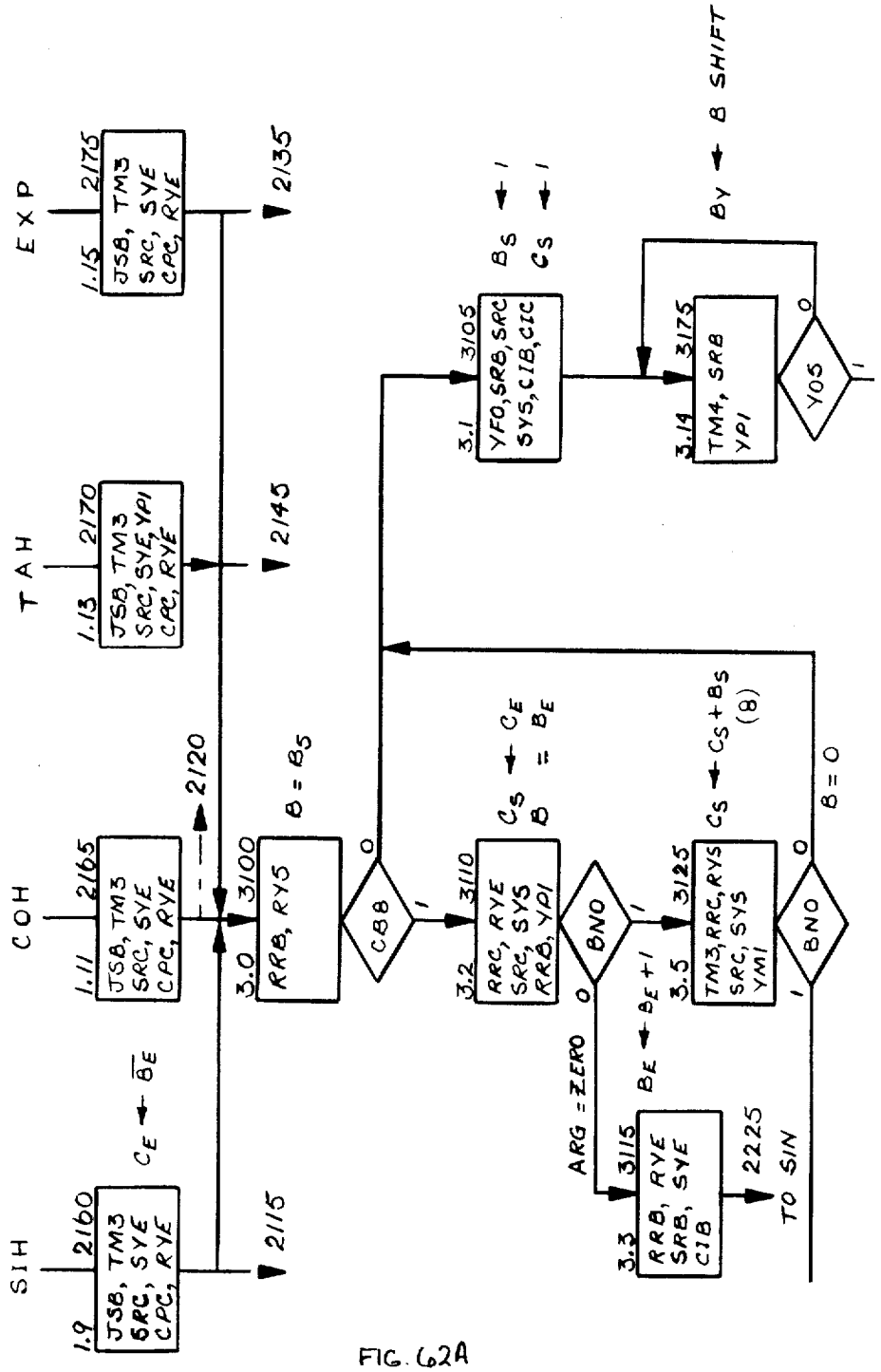

EXP
TAH
COH
SIH
1.15 2175 JSB, TM3 SRC, SYE CPC, RYE
1.13 2170 JSB, TM3 SRC, SYE, YP1 CPC, RYE
1.11 2165 JSB, TM3 SRC, SYE CPC, RYE
1.9 2160 JSB, TM3 SRC, SYE CPC, RYE
2135
2145
2120
2115
$C_E \leftarrow \overline{B_E}$
3.0 3100 RRB, RY5
$B = B_S$
C88
3.1 3105 YF0, SRB, SRC SYS, C1B, C1C
$B_S \leftarrow 1$
$C_S \leftarrow 1$
3.14 3175 TM4, SRB YP1
Y05
$B_Y \leftarrow B$ SHIFT
3.2 3110 RRC, RYE SRC, SYS RRB, YP1
$C_S \leftarrow C_E$
$B = B_E$
BNO
ARG = ZERO
3.3 3115 RRB, RYE SRB, SYE C1B
$B_E \leftarrow B_E + 1$
2225
TO SIN
3.5 3125 TM3, RRC, RY5 SRC, SYS YM1
$C_S \leftarrow C_S + B_S$ (8)
BNO
B = 0

FIG. 62A

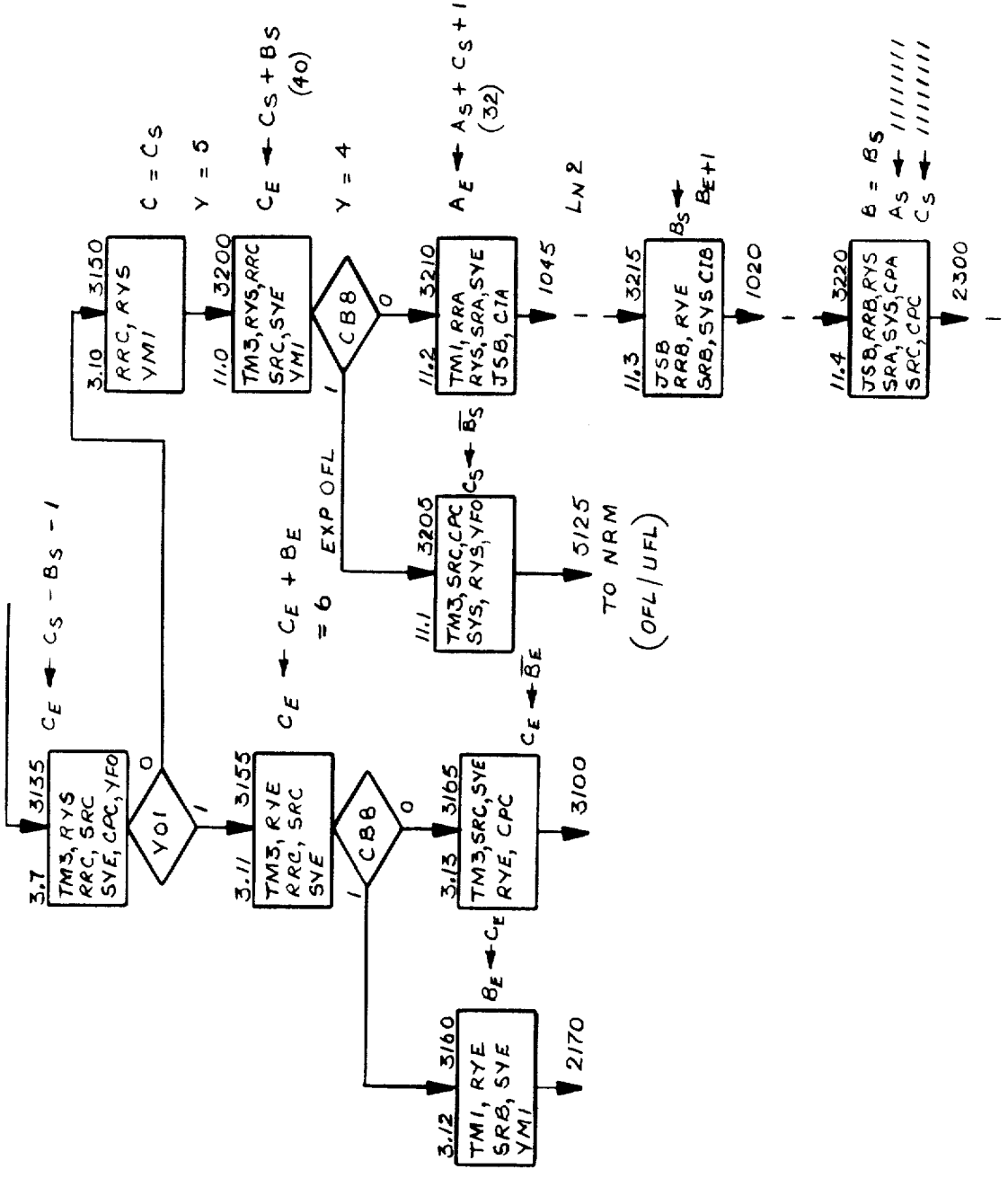

3.7
3135
TM3, RYS
RRC, SRC
SYE, CPC, YFO
$C_E \leftarrow C_S - B_S - 1$
YO1
0
1
3.11
3155
TM3, RYE
RRC, SRC
SYE
$C_E \leftarrow C_E + B_E$
= 6
EXP OFL
CBB
0
1
3.10
3150
RRC, RYS
YMI
$C = C_S$
$Y = 5$
11.0
3200
TM3, RYS, RRC
SRC, SYE
YMI
$C_E \leftarrow C_S + B_S$
(40)
$Y = 4$
CBB
0
1
11.2
3210
TM1, RRA
RYS, SRA, SYE
JSB, C1A
$A_E \leftarrow A_S + C_S + 1$
(32)
1045
LN 2
11.3
3215
JSB
RRB, RYE
SRB, SYS C1B
$B_S \leftarrow B_{E+1}$
1020
11.4
3220
JSB, RRB, RYS
SRA, SYS, CPA
SRC, CPC
$B = B_S$
$A_S \leftarrow$ ////////
$C_S \leftarrow$ ////////
2300
11.1
3205
TM3, SRC, CPC
SYS, RYS, YFO
$C_S \leftarrow \overline{B_S}$
5125
TO NRM
(OFL/UFL)
$C_E \leftarrow \overline{B_E}$
3100
3.13
3165
TM3, SRC, SYE
RYE, CPC
$B_E \leftarrow C_E$
3.12
3160
TM1, RYE
SRB, SYE
YMI
2170

FIG. 62B

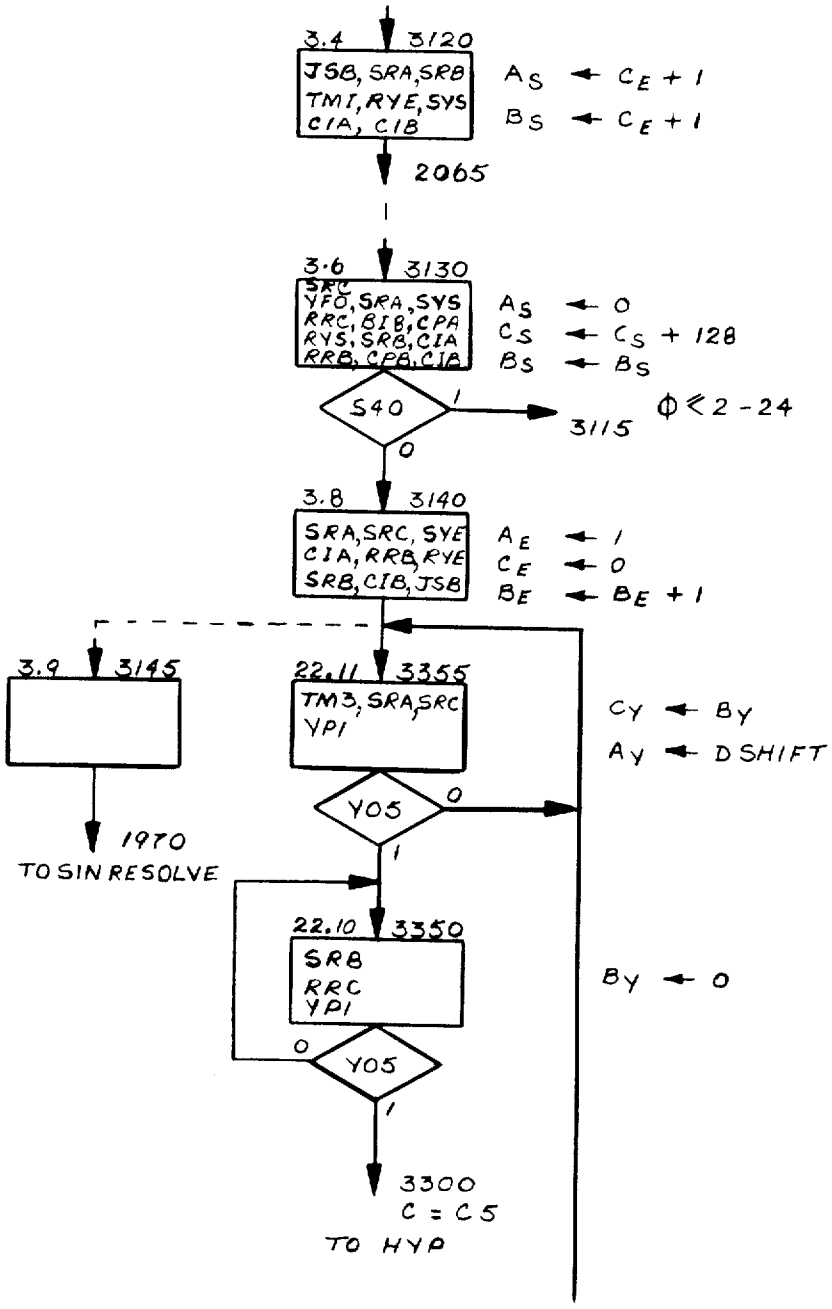

3.4 3120
JSB, SRA, SRB
TMI, RYE, SYS
CIA, CIB
AS ← CE + 1
BS ← CE + 1
2065
3.6 3130
SRC
YFO, SRA, SYS
RRC, BIB, CPA
RYS, SRB, CIA
RRB, CPB, CIB
AS ← 0
CS ← CS + 128
BS ← BS
S40
1
3115
Φ ≪ 2 - 24
0
3.8 3140
SRA, SRC, SYE
CIA, RRB, RYE
SRB, CIB, JSB
AE ← 1
CE ← 0
BE ← BE + 1
3.9 3145
1970
TO SIN RESOLVE
22.11 3355
TM3, SRA, SRC
YPI
CY ← BY
AY ← D SHIFT
Y05
0
1
22.10 3350
SRB
RRC
YPI
BY ← 0
0
Y05
1
3300
C = C5
TO HYP

FIG. 62C

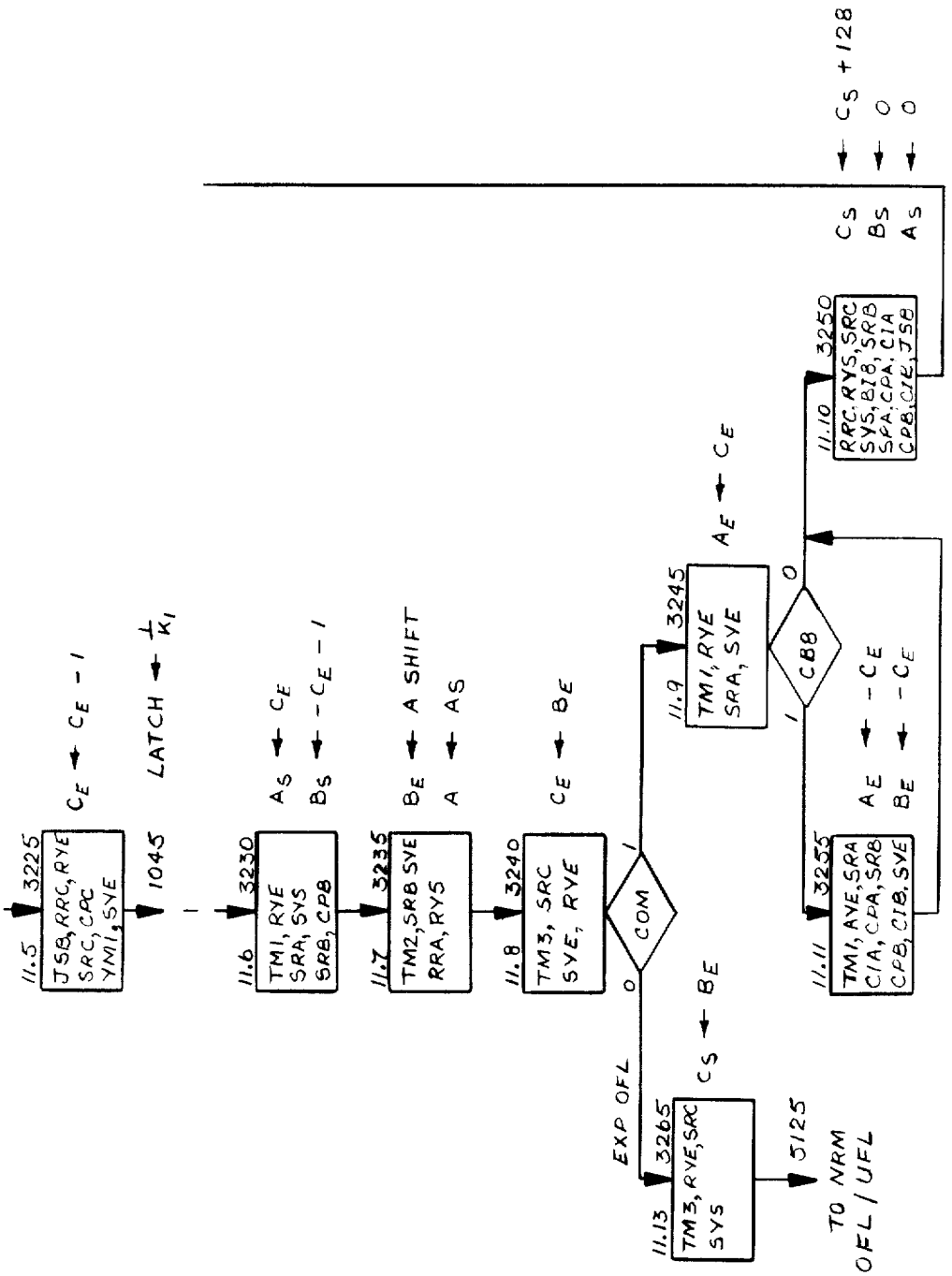

11.5
3225
JSB, RRC, RYE
SRC, CPC
YM1, SYE
$C_E \leftarrow C_E - 1$
1045
LATCH $\leftarrow \frac{1}{K_1}$
11.6
3230
TM1, RYE
SRA, SYS
SRB, CPB
$A_S \leftarrow C_E$
$B_S \leftarrow -C_E - 1$
11.7
3235
TM2, SRB SYE
RRA, RYS
$B_E \leftarrow$ A SHIFT
$A \leftarrow A_S$
11.8
3240
TM3, SRC
SYE, RYE
$C_E \leftarrow B_E$
COM
11.9
3245
TM1, RYE
SRA, SYE
$A_E \leftarrow C_E$
CBB
11.10
3250
RRC, RYS, SRC
SYS, BIB, SRB
SPA, CPA, CIA
CPB, CIE, JSB
$C_S$
$B_S$
$A_S$
$C_S + 128$
0
0
11.11
3255
TM1, AYE, SRA
CIA, CPA, SRB
CPB, CIB, SYE
$A_E \leftarrow -C_E$
$B_E \leftarrow -C_E$
EXP OFL
11.13
3265
TM3, RYE, SRC
SYS
$C_S \leftarrow B_E$
5125
TO NRM
OFL / UFL

FIG. 62D

HYPERBOLIC PERSCALE FLOWCHART

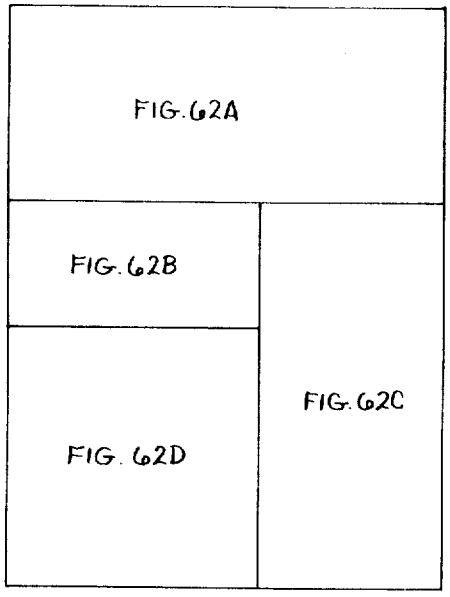



FIG. 63

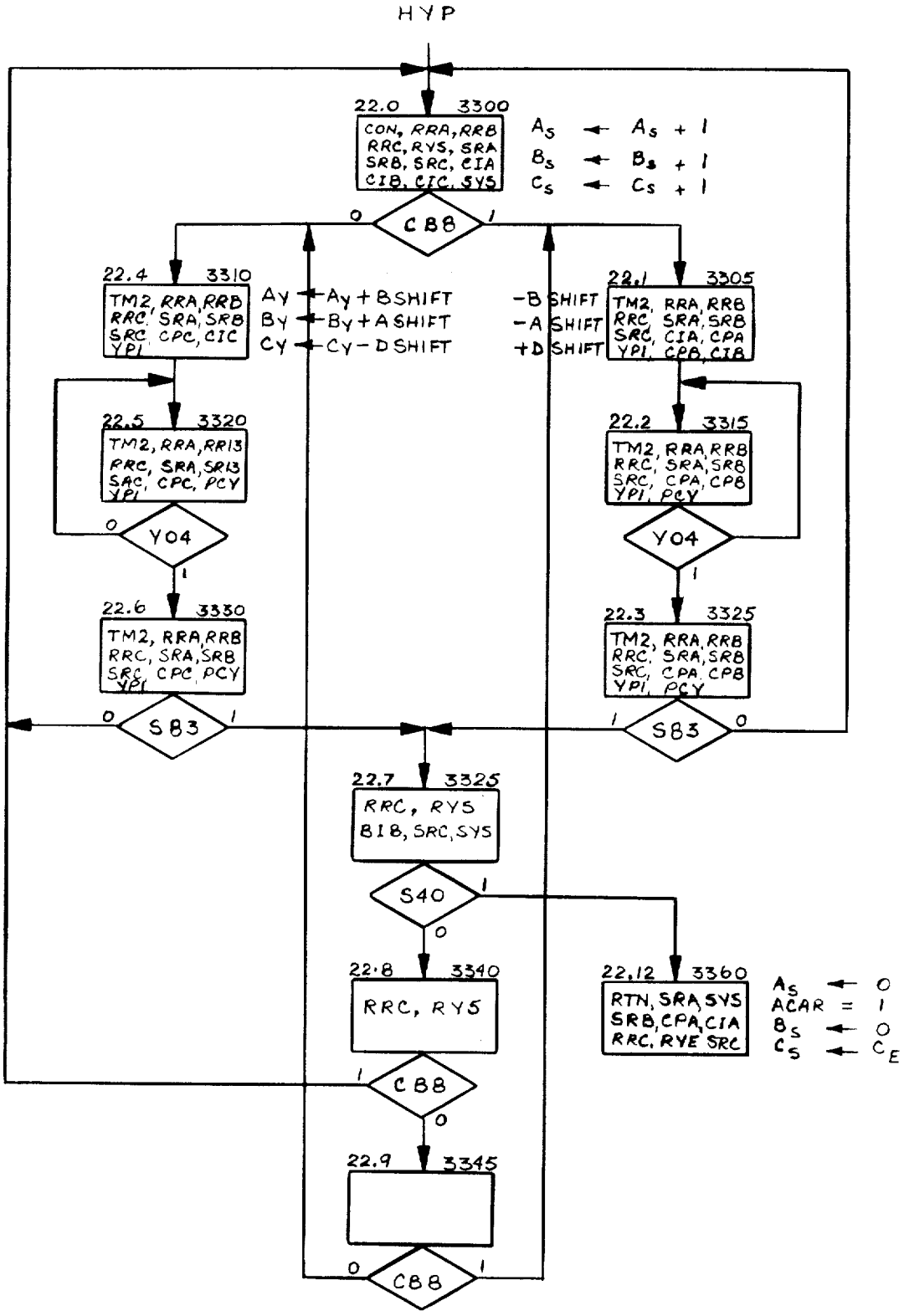

HYP
22.0 3300
CON, RRA, RRB
RRC, RYS, SRA
SRB, SRC, CIA
CIB, CIC, SYS
$A_S \leftarrow A_S + 1$
$B_S \leftarrow B_S + 1$
$C_S \leftarrow C_S + 1$
CBB
22.4 3310
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CPC, CIC
YPI
$A_Y \leftarrow A_Y + B$ SHIFT
$B_Y \leftarrow B_Y + A$ SHIFT
$C_Y \leftarrow C_Y - D$ SHIFT
22.1 3305
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CIA, CPA
YPI, CPB, CIB
$-B$ SHIFT
$-A$ SHIFT
$+D$ SHIFT
22.5 3320
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CPC, PCY
YPI
Y04
22.2 3315
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CPA, CPB
YPI, PCY
Y04
22.6 3330
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CPC, PCY
YPI
SB3
22.3 3325
TM2, RRA, RRB
RRC, SRA, SRB
SRC, CPA, CPB
YPI, PCY
SB3
22.7 3325
RRC, RYS
BIB, SRC, SYS
S40
22.8 3340
RRC, RYS
CBB
22.12 3360
RTN, SRA, SYS
SRB, CPA, CIA
RRC, RYE SRC
$A_S \leftarrow 0$
ACAR = 1
$B_S \leftarrow 0$
$C_S \leftarrow C_E$
22.9 3345
CBB

FIG. 64A

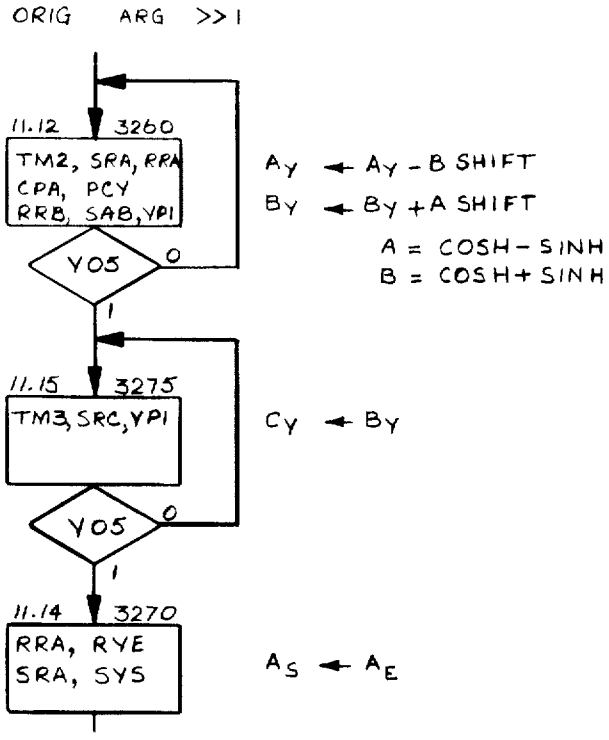

ORIG ARG >>1
11.12 3260
TM2, SRA, RRA
CPA, PCY
RRB, SAB, YPI
YO5
0
1
11.15 3275
TM3, SRC, YPI
YO5
0
1
11.14 3270
RRA, RYE
SRA, SYS
AY ← AY - B SHIFT
BY ← BY + A SHIFT
A = COSH - SINH
B = COSH + SINH
CY ← BY
AS ← AE

FIG. 64B

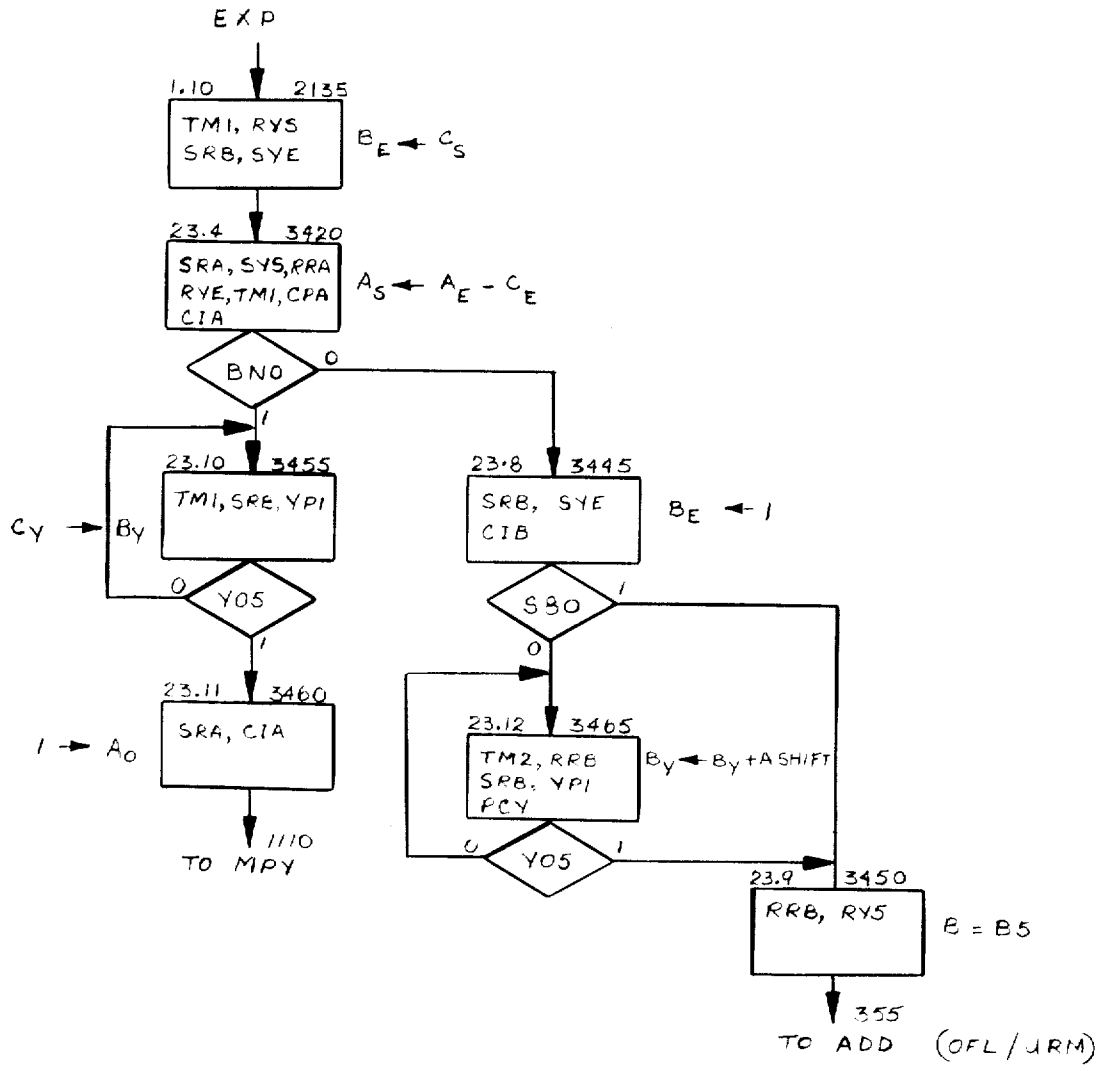

EXP
1.10
2135
TM1, RYS
SRB, SYE
$B_E \leftarrow C_S$
23.4
3420
SRA, SYS, RRA
RYE, TMI, CPA
CIA
$A_S \leftarrow A_E - C_E$
BNO
0
1
23.10
3455
TM1, SRE, YP1
$C_Y$
$B_Y$
0
Y05
1
23.11
3460
SRA, CIA
1 → $A_0$
1110
TO MPY
23.8
3445
SRB, SYE
CIB
$B_E \leftarrow 1$
SBO
1
0
23.12
3465
TM2, RRB
SRB, YP1
PCY
$B_Y \leftarrow B_Y$ + A SHIFT
0
Y05
1
23.9
3450
RRB, RY5
B = B5
355
TO ADD
(OFL / URM)

FIG. 64C

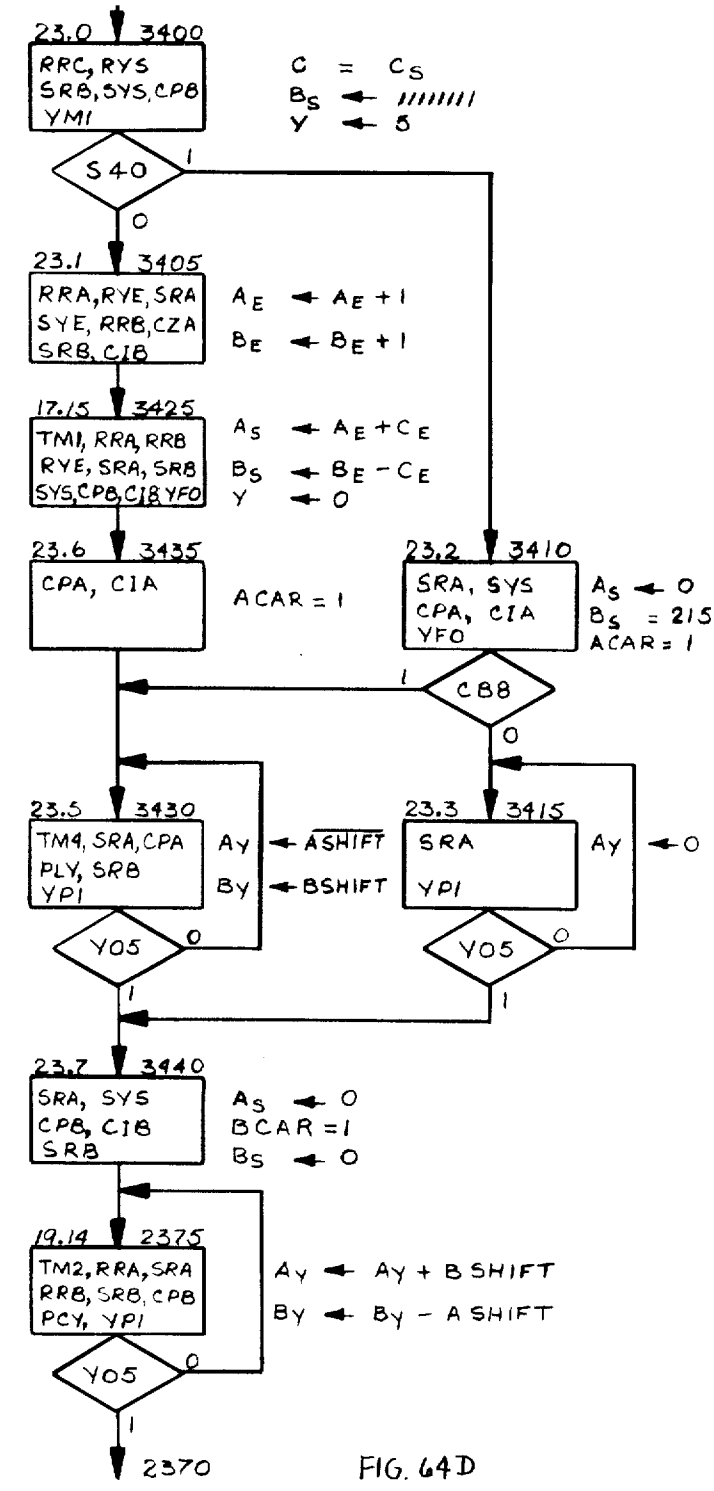

23.0 3400
RRC, RYS
SRB, SYS, CPB
YM1
C = CS
BS ← ////////
Y ← 5
S40
1
0
23.1 3405
RRA, RYE, SRA
SYE, RRB, CZA
SRB, CIB
AE ← AE + 1
BE ← BE + 1
17.15 3425
TM1, RRA, RRB
RYE, SRA, SRB
SYS, CPB, CIB YF0
AS ← AE + CE
BS ← BE - CE
Y ← 0
23.6 3435
CPA, CIA
ACAR = 1
23.2 3410
SRA, SYS
CPA, CIA
YF0
AS ← 0
BS = 215
ACAR = 1
CB8
1
0
23.5 3430
TM4, SRA, CPA
PLY, SRB
YP1
AY ← ASHIFT
BY ← BSHIFT
23.3 3415
SRA
YP1
AY ← 0
Y05
0
1
23.7 3440
SRA, SYS
CPB, CIB
SRB
AS ← 0
BCAR = 1
BS ← 0
19.14 2375
TM2, RRA, SRA
RRB, SRB, CPB
PCY, YP1
AY ← AY + B SHIFT
BY ← BY - A SHIFT
2370
TO SIN RESOLVE

FIG. 64D

SINH/COSH RESOLVE FLOWCHARTS

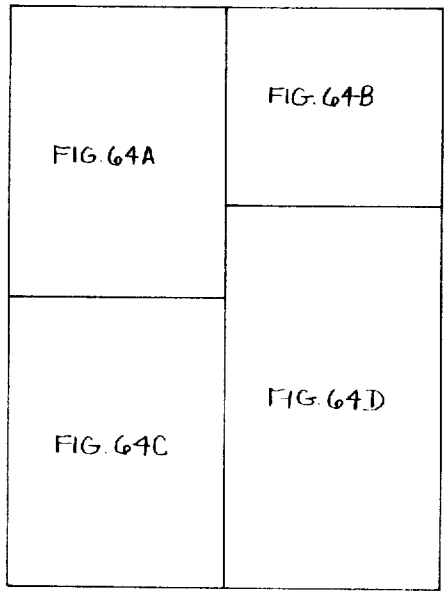



FIG. 65

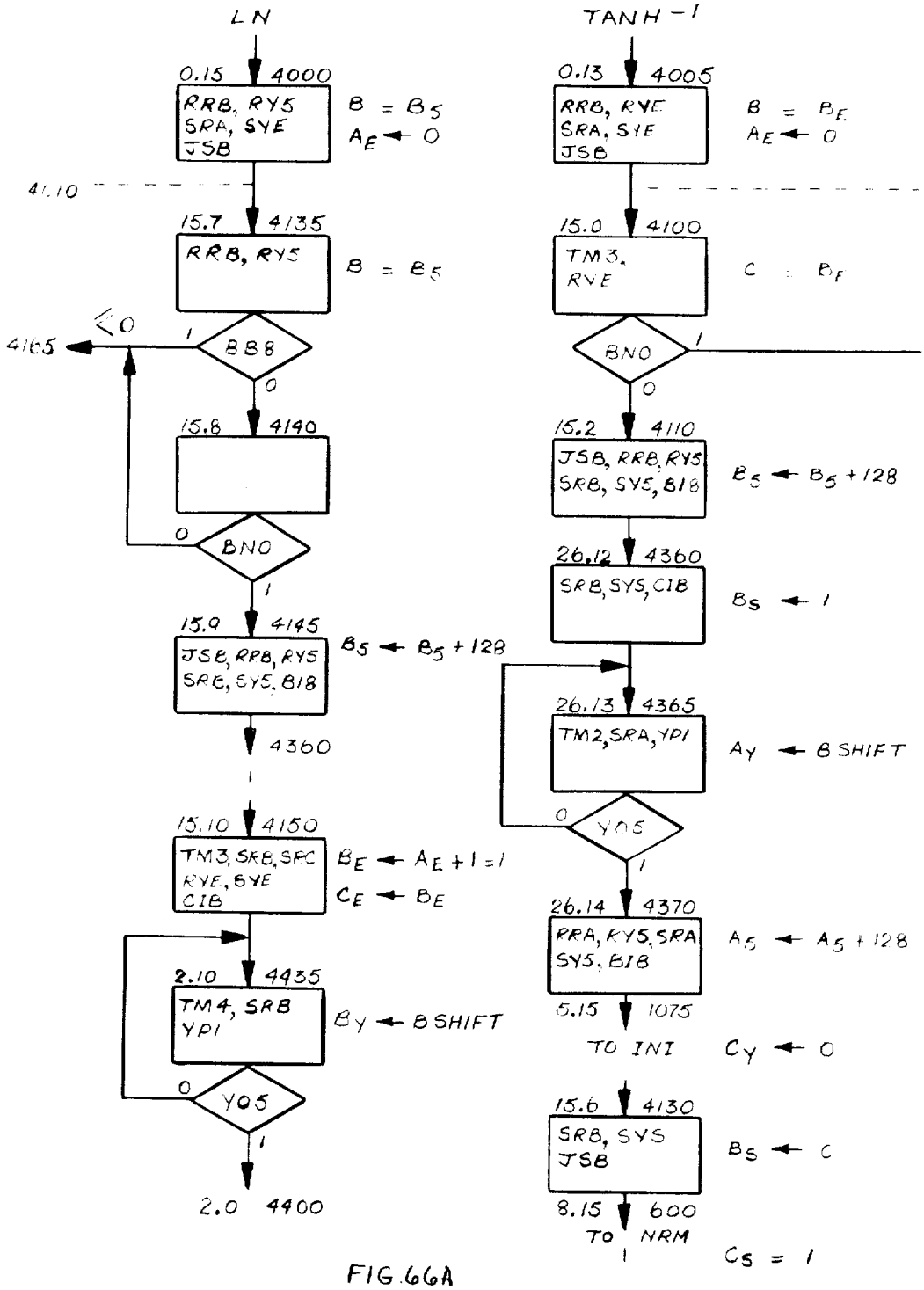

LN
0.15 4000
RRB, RY5
SRA, SYE
JSB
B = B5
AE ← 0
4110
15.7 4135
RRB, RY5
B = B5
<0
4165
1
BBB
0
15.8 4140
0
BNO
1
15.9 4145
JSB, RPB, RY5
SRE, SY5, B18
B5 ← B5 + 128
4360
15.10 4150
TM3, SRB, SPC
RYE, SYE
CIB
BE ← AE + 1 = 1
CE ← BE
2.10 4435
TM4, SRB
YP1
BY ← B SHIFT
0
Y05
1
2.0 4400
TANH-1
0.13 4005
RRB, RYE
SRA, SYE
JSB
B = BE
AE ← 0
15.0 4100
TM3,
RYE
C = BF
BNO
1
0
15.2 4110
JSB, RRB, RY5
SRB, SY5, B18
B5 ← B5 + 128
26.12 4360
SRB, SY5, C1B
B5 ← 1
26.13 4365
TM2, SRA, YP1
AY ← B SHIFT
0
Y05
1
26.14 4370
RRA, RY5, SRA
SY5, B18
A5 ← A5 + 128
5.15 1075
TO INI
CY ← 0
15.6 4130
SRB, SY5
JSB
B5 ← C
8.15 600
TO NRM
CS = 1

FIG. 66A

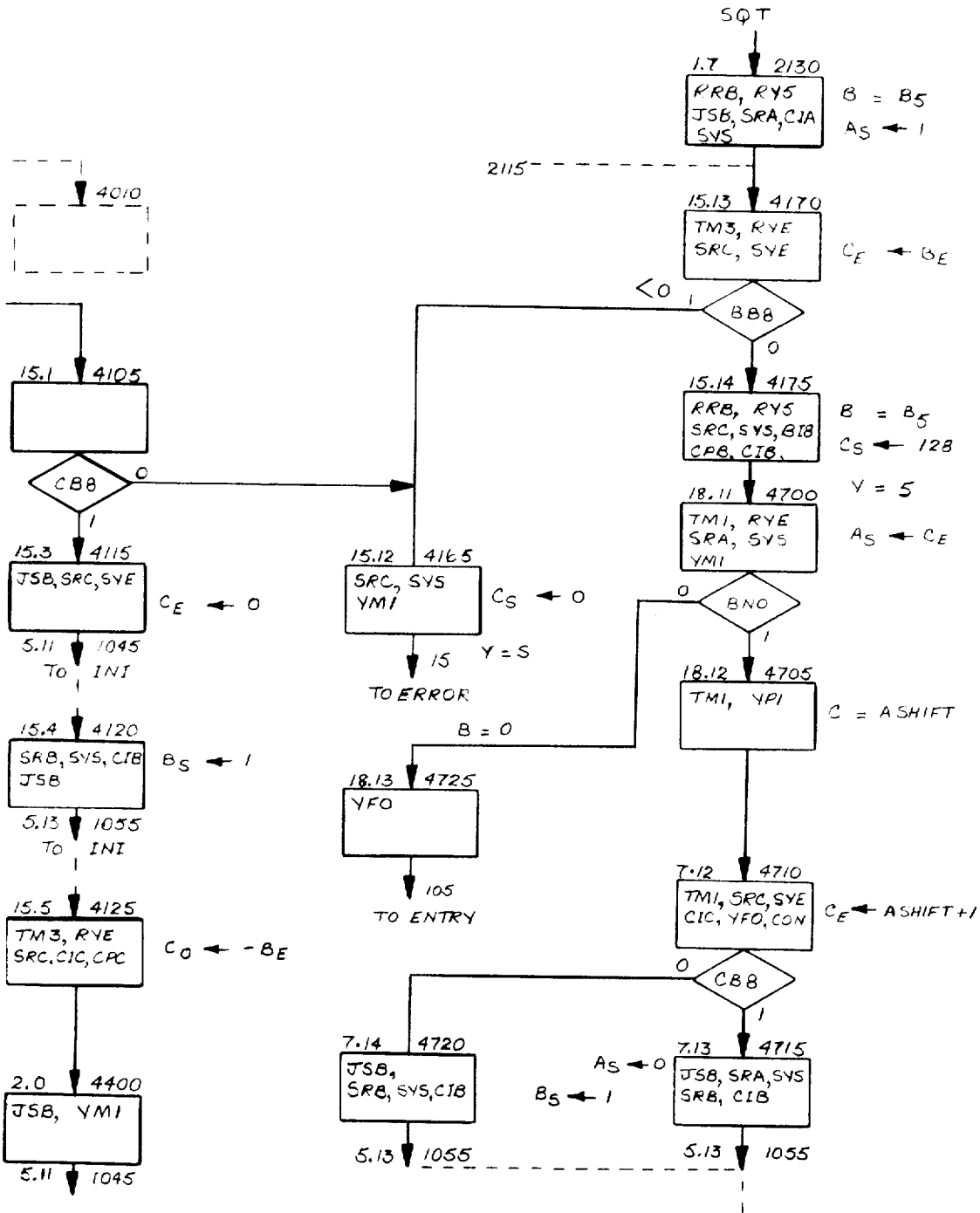

SQT
1.7 2130
RRB, RYS
JSB, SRA, CIA
SYS
B = BS
AS ← 1
2115
15.13 4170
TM3, RYE
SRC, SYE
CE ← BE
<0
BBB
0
1
4010
15.1 4105
CBB
0
1
15.3 4115
JSB, SRC, SYE
CE ← 0
5.11 1045
TO INI
15.4 4120
SRB, SYS, CIB
JSB
BS ← 1
5.13 1055
TO INI
15.5 4125
TM3, RYE
SRC, CIC, CPC
C0 ← -BE
2.0 4400
JSB, YM1
5.11 1045
15.12 4165
SRC, SYS
YM1
CS ← 0
15
Y = S
TO ERROR
15.14 4175
RRB, RYS
SRC, SYS, BIB
CPB, CIB,
B = BS
CS ← 128
Y = 5
18.11 4700
TM1, RYE
SRA, SYS
YM1
AS ← CE
0
BNO
1
18.12 4705
TM1, YP1
C = ASHIFT
B = 0
18.13 4725
YFO
105
TO ENTRY
7.12 4710
TM1, SRC, SYE
CIC, YFO, CON
CE ← ASHIFT+1
0
CBB
1
7.14 4720
JSB,
SRB, SYS, CIB
AS ← 0
BS ← 1
5.13 1055
7.13 4715
JSB, SRA, SYS
SRB, CIB
5.13 1055

FIG. 66B

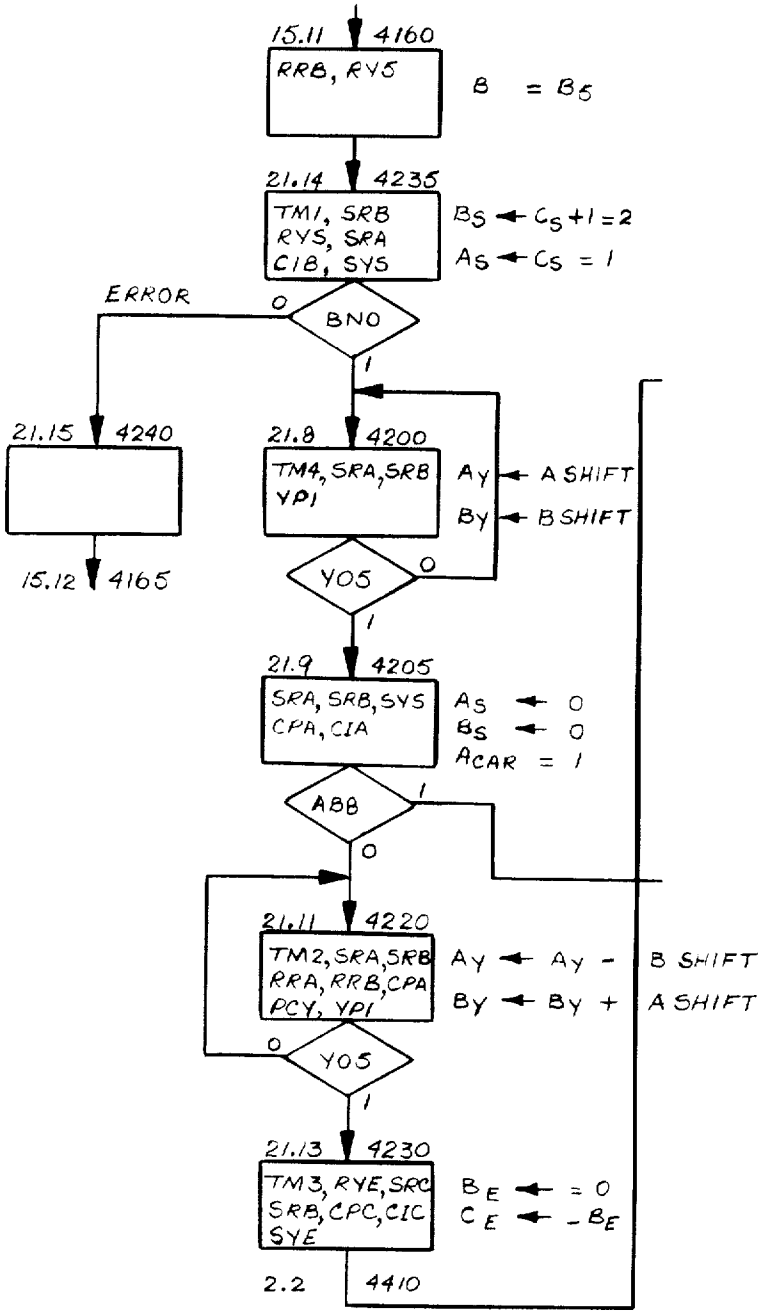

15.11 4160
RRB, RYS
B = BS
21.14 4235
TM1, SRB
RYS, SRA
CIB, SYS
BS ← CS + 1 = 2
AS ← CS = 1
ERROR
0
BNO
1
21.15 4240
15.12 4165
21.8 4200
TM4, SRA, SRB
YP1
AY ← A SHIFT
BY ← B SHIFT
YO5
0
1
21.9 4205
SRA, SRB, SYS
CPA, CIA
AS ← 0
BS ← 0
ACAR = 1
ABB
1
0
21.11 4220
TM2, SRA, SRB
RRA, RRB, CPA
PCY, YP1
AY ← AY − B SHIFT
BY ← BY + A SHIFT
0
YO5
1
21.13 4230
TM3, RYE, SRC
SRB, CPC, CIC
SYE
BE ← = 0
CE ← − BE
2.2 4410

FIG. 66C

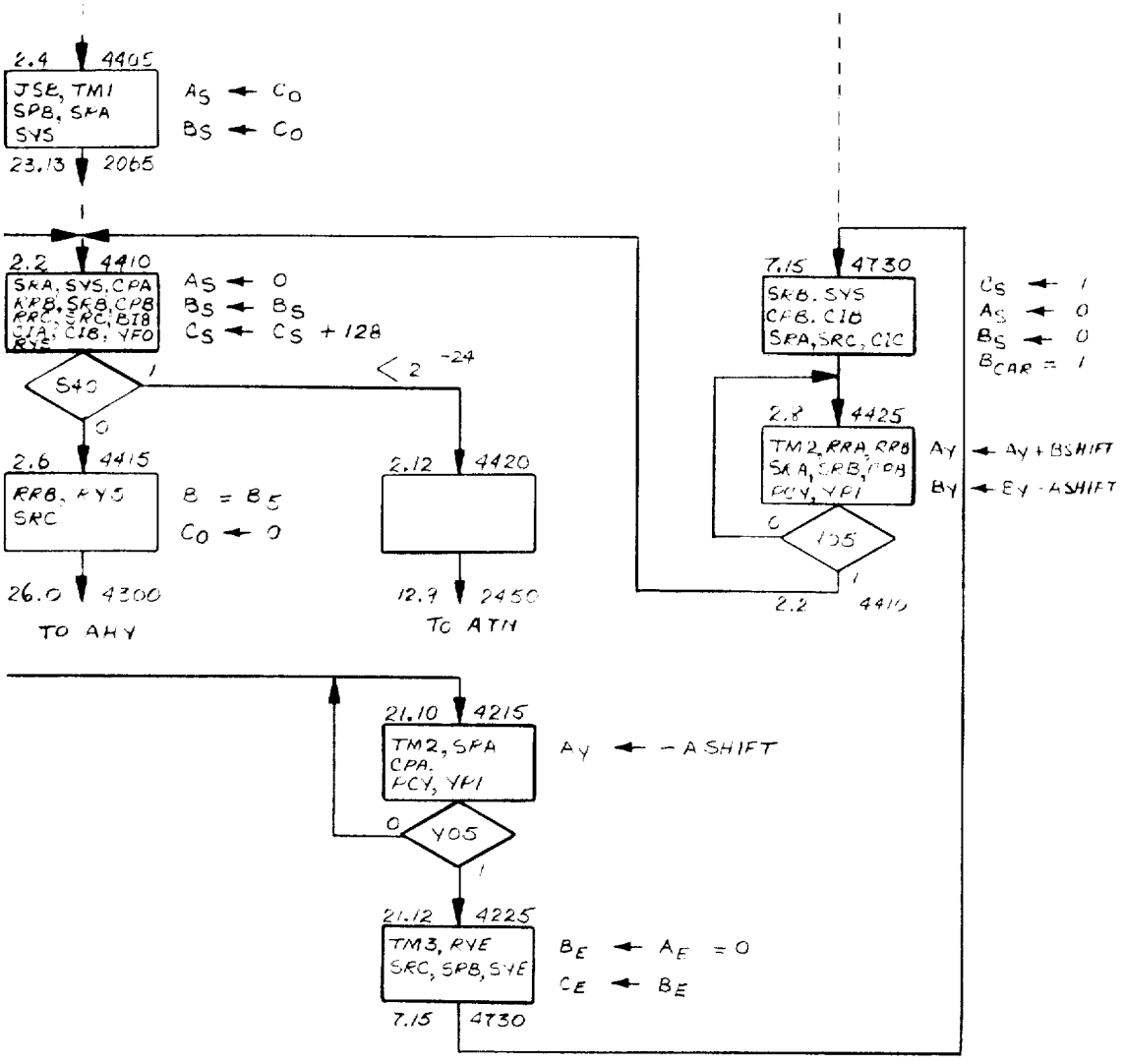

2.4 4405
JSE, TM1
SPB, SRA
SYS
23.13 2065
AS ← CO
BS ← CO
2.2 4410
SRA, SYS, CPA
XPB, SRB, CPB
PRC, SRC, BI8
CIA, CI8, YFO
RYE
AS ← 0
BS ← BS
CS ← CS + 128
S40
1
0
< 2 -24
2.6 4415
RRB, RYS
SRC
B = BS
CO ← 0
26.0 4300
TO AHY
2.12 4420
12.9 2450
TO ATN
7.15 4730
SRB, SYS
CPB, CI8
SRA, SRC, CIC
CS ← 1
AS ← 0
BS ← 0
BCAR = 1
2.8 4425
TM2, RRA, RPB
SRA, SPB, YPA
PCY, YP1
AY ← AY + BSHIFT
BY ← BY - ASHIFT
105
0
1
2.2 4410
21.10 4215
TM2, SPA
CPA,
PCY, YP1
AY ← - ASHIFT
YO5
0
1
21.12 4225
TM3, RYE
SRC, SPB, SYE
BE ← AE = 0
CE ← BE
7.15 4730

FIG. 66D

LN/ARCTANH/Sgt. PRESCALE FLOWCHARTS

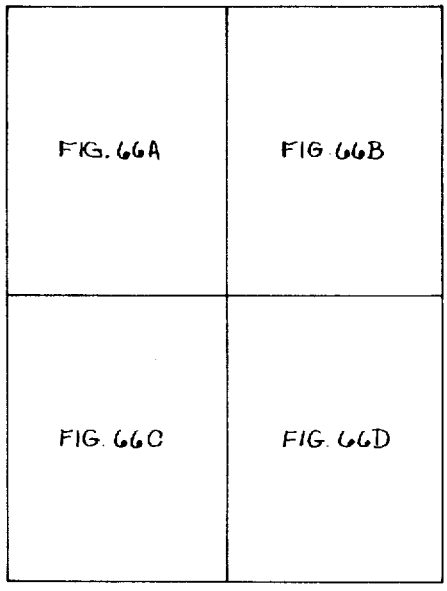



FIG. 67

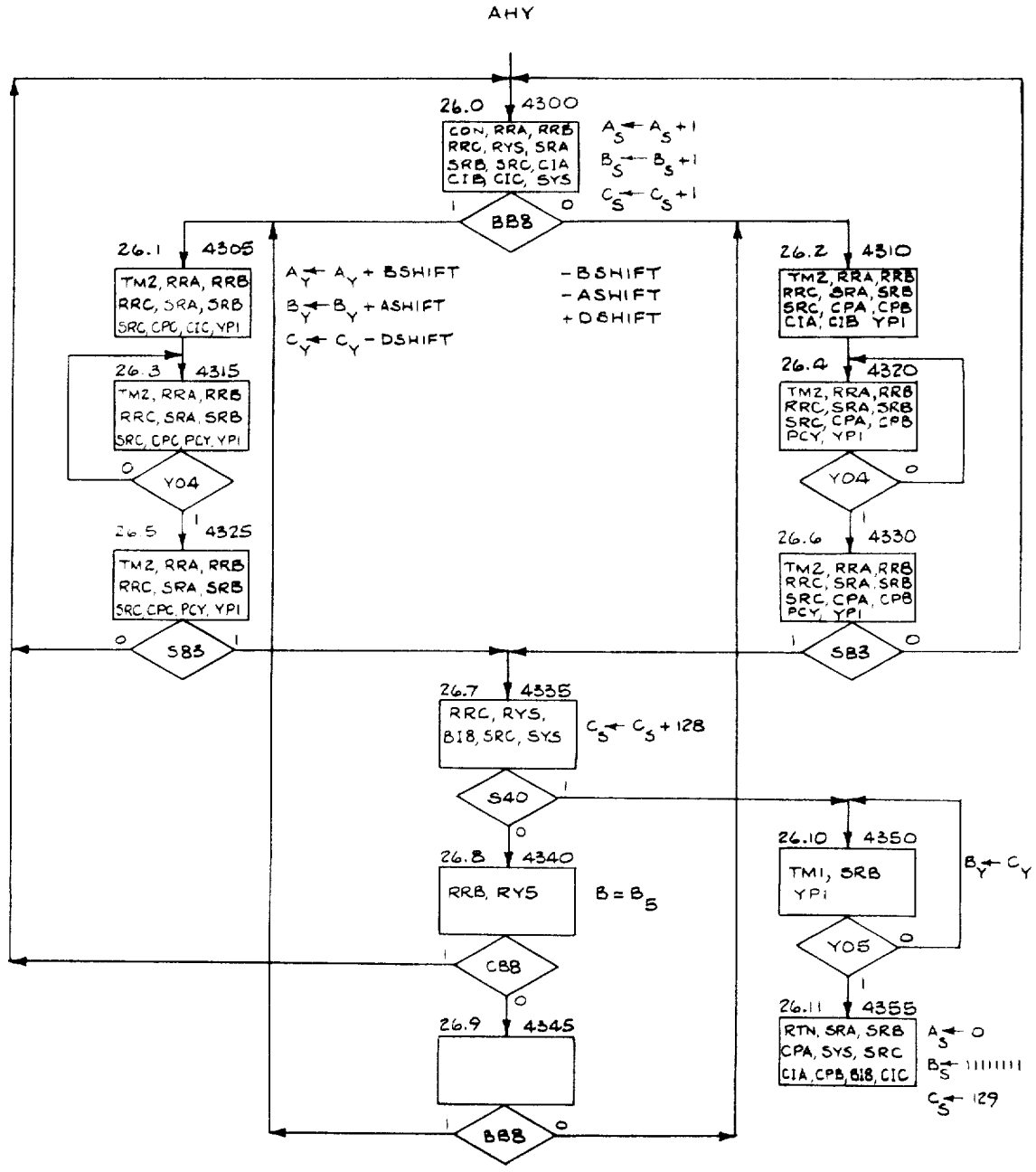

AHY
26.0 4300
CON, RRA, RRB RRC, RYS, SRA SRB, SRC, CIA CIB, CIC, SYS
$A_S \leftarrow A_S + 1$
$B_S \leftarrow B_S + 1$
$C_S \leftarrow C_S + 1$
BB8
26.1 4305
TM2, RRA, RRB RRC, SRA, SRB SRC, CPC, CIC, YPI
$A_Y \leftarrow A_Y$ + BSHIFT
$B_Y \leftarrow B_Y$ + ASHIFT
$C_Y \leftarrow C_Y$ - DSHIFT
-BSHIFT
-ASHIFT
+DSHIFT
26.2 4310
TM2, RRA, RRB RRC, SRA, SRB SRC, CPA, CPB CIA, CIB YPI
26.3 4315
TM2, RRA, RRB RRC, SRA, SRB SRC, CPC, PCY, YPI
Y04
26.4 4320
TM2, RRA, RRB RRC, SRA, SRB SRC, CPA, CPB PCY, YPI
Y04
26.5 4325
TM2, RRA, RRB RRC, SRA, SRB SRC, CPC, PCY, YPI
26.6 4330
TM2, RRA, RRB RRC, SRA, SRB SRC, CPA, CPB PCY, YPI
S85
S83
26.7 4335
RRC, RYS, BI8, SRC, SYS
$C_S \leftarrow C_S + 128$
S40
26.8 4340
RRB, RY5
$B = B_S$
CB8
26.9 4345
BB8
26.10 4350
TM1, SRB YPi
$B_Y \leftarrow C_Y$
Y05
26.11 4355
RTN, SRA, SRB CPA, SYS, SRC CIA, CPB, BI8, CIC
$A_S \leftarrow 0$
$B_S \leftarrow$ 11111111
$C_S \leftarrow$ 129

FIG. 68A

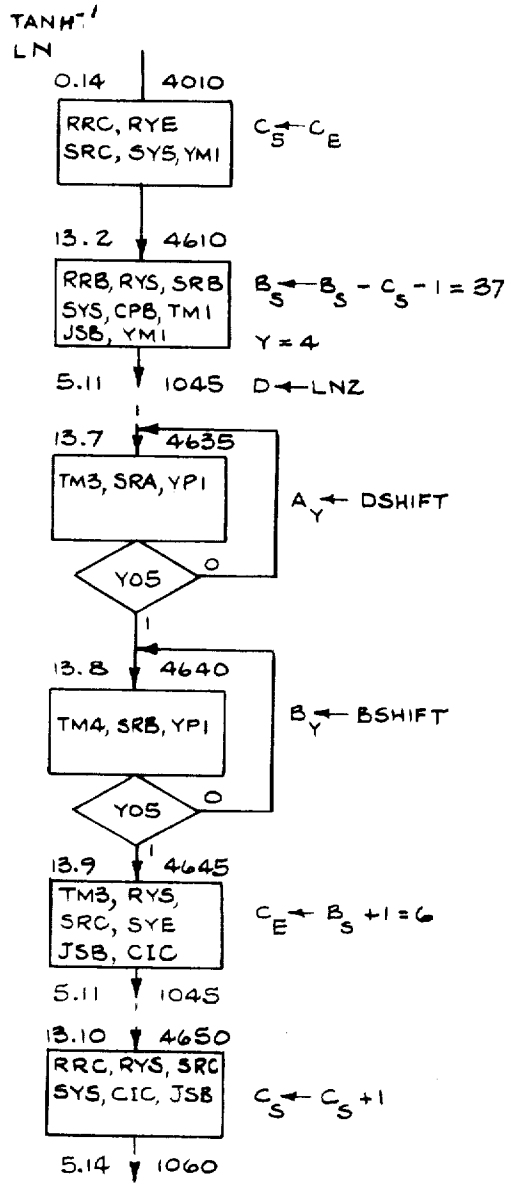

TANH⁻¹
LN
0.14
4010
RRC, RYE
SRC, SYS, YM1
$C_S \leftarrow C_E$
13.2
4610
RRB, RYS, SRB
SYS, CPB, TM1
JSB, YM1
$B_S \leftarrow B_S - C_S - 1 = 37$
Y = 4
5.11
1045
D ← LNZ
13.7
4635
TM3, SRA, YP1
$A_Y \leftarrow$ DSHIFT
Y05
0
1
13.8
4640
TM4, SRB, YP1
$B_Y \leftarrow$ BSHIFT
Y05
0
1
13.9
4645
TM3, RYS,
SRC, SYE
JSB, CIC
$C_E \leftarrow B_S + 1 = 6$
5.11
1045
13.10
4650
RRC, RYS, SRC
SYS, CIC, JSB
$C_S \leftarrow C_S + 1$
5.14
1060

FIG. 68B

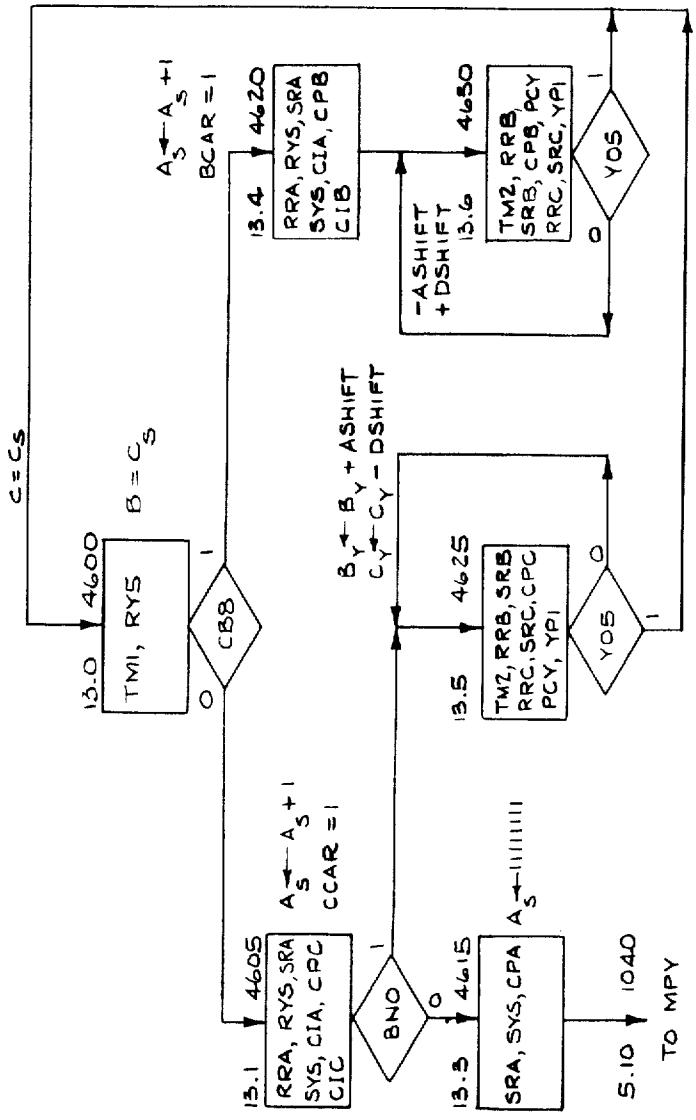

C = Cs
B = Cs
13.0 4600
TM1, RYS
CBB
1
0
A_S ← A_S + 1
BCAR = 1
13.4 4620
RRA, RYS, SRA
SYS, CIA, CPB
CIB
-ASHIFT
+DSHIFT
13.6 4630
TM2, RRB
SRB, CPB, PCY
RRC, SRC, YPI
YO5
B_Y ← B_Y + ASHIFT
C_Y ← C_Y - DSHIFT
13.5 4625
TM2, RRB, SRB
RRC, SRC, CPC
PCY, YPI
YO5
A_S ← A_S + 1
CCAR = 1
13.1 4605
RRA, RYS, SRA
SYS, CIA, CPC
CIC
BNO
13.3 4615
SRA, SYS, CPA
A_S ← 11111111
5.10 1040
TO MPY

FIG 68C

ARC HYPER RESOLVE FLOWCHARTS

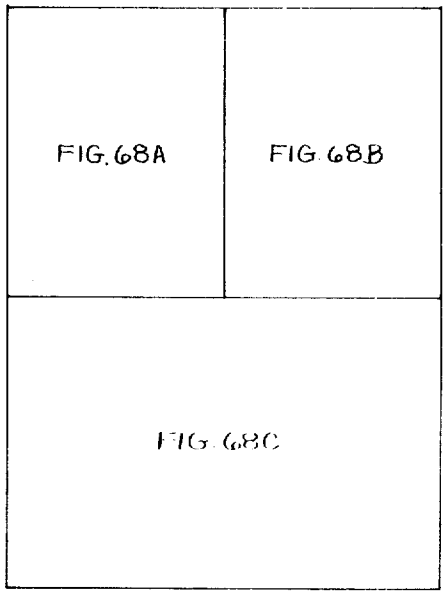



FIG. 69

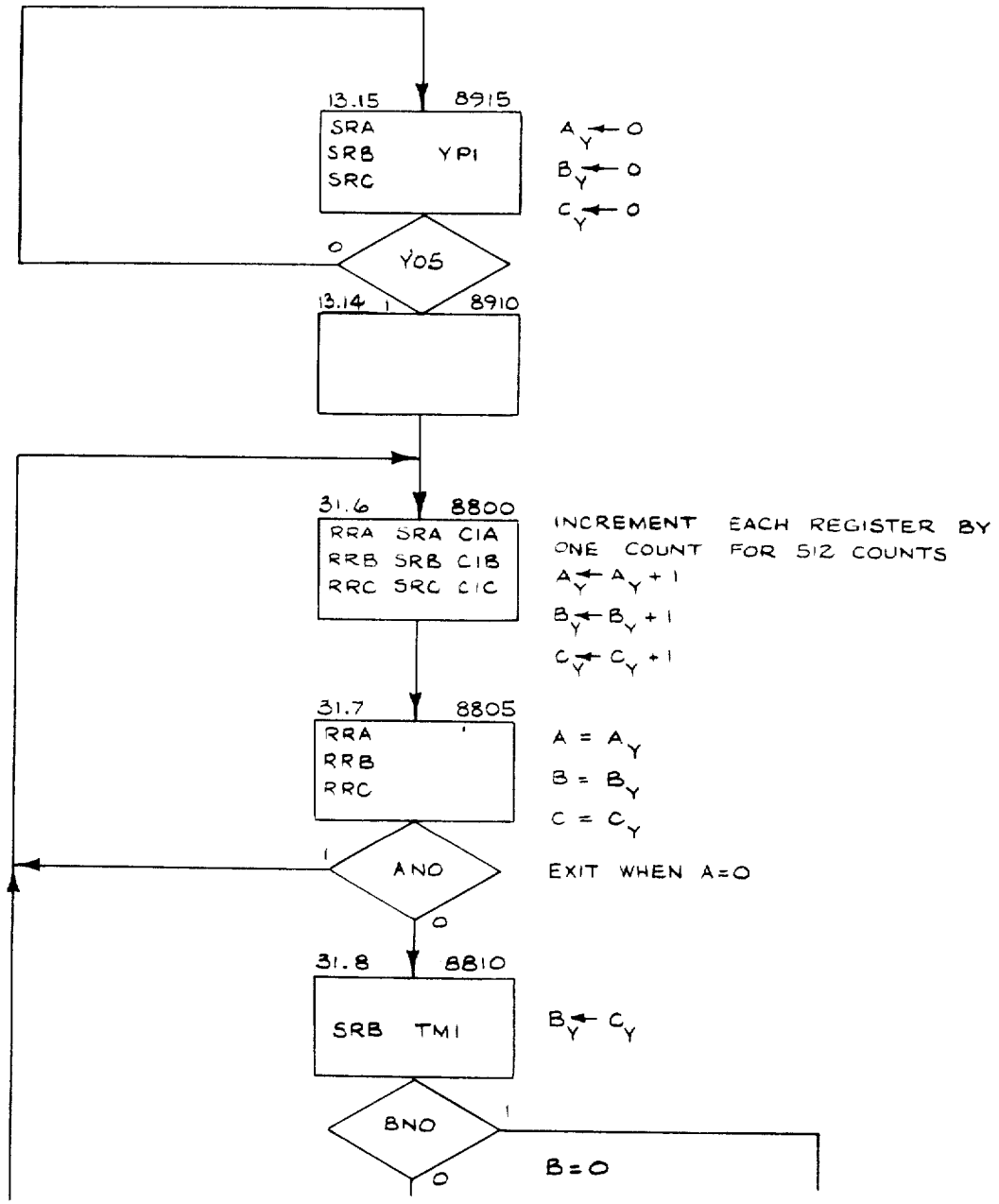

13.15
8915
SRA
SRB
SRC
YP1
A_Y ← 0
B_Y ← 0
C_Y ← 0
0
Y05
13.14
1
8910
31.6
8800
RRA SRA CIA
RRB SRB CIB
RRC SRC CIC
INCREMENT EACH REGISTER BY ONE COUNT FOR 512 COUNTS
A_Y ← A_Y + 1
B_Y ← B_Y + 1
C_Y ← C_Y + 1
31.7
8805
RRA
RRB
RRC
A = A_Y
B = B_Y
C = C_Y
1
ANO
EXIT WHEN A=0
0
31.8
8810
SRB TM1
B_Y ← C_Y
BNO
1
0
B=0

FIG. 70A

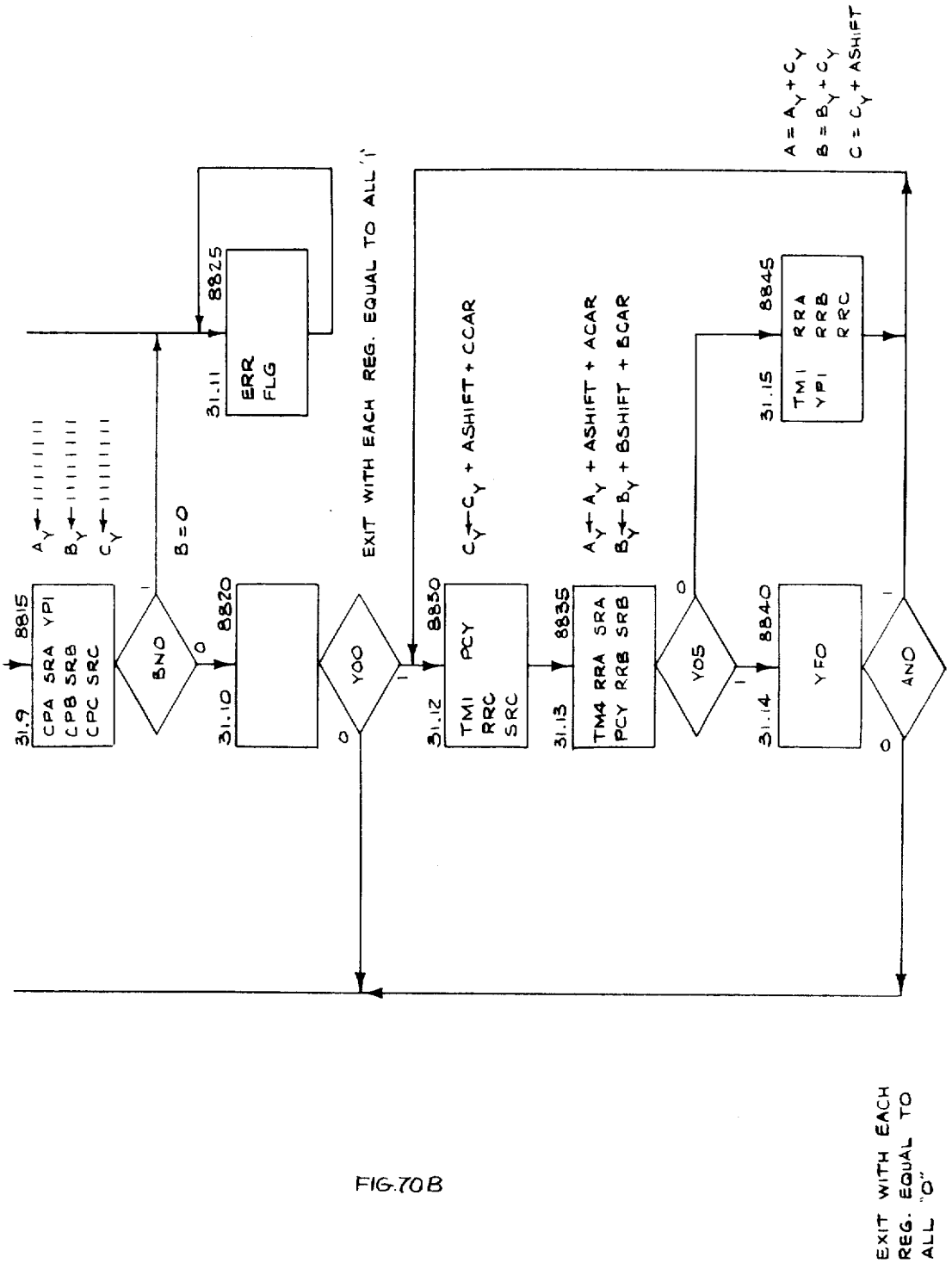

31.9 8815
CPA SRA YPI
CPB SRB
CPC SRC
BNO
B=0
8820
31.10
8825
31.11
ERR
FLG
EXIT WITH EACH REG. EQUAL TO ALL '1'
YOO
8830
31.12
TMI PCY
RRC
SRC
8835
31.13
TM4 RRA SRA
PCY RRB SRB
C_Y ← C_Y + ASHIFT + CCAR
A_Y ← A_Y + ASHIFT + ACAR
B_Y ← B_Y + BSHIFT + BCAR
YO5
8840
31.14
YFO
8845
31.15
TMI RRA
YPI RRB
RRC
A = A_Y + C_Y
B = B_Y + C_Y
C = C_Y + ASHIFT
ANO
EXIT WITH EACH REG. EQUAL TO ALL "0"

FIG. 70B

DIAGNOSTIC ROUTINE FLOWCHARTS

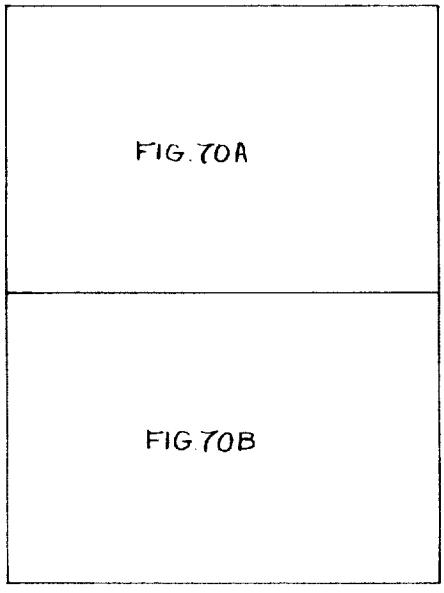



FIG. 71

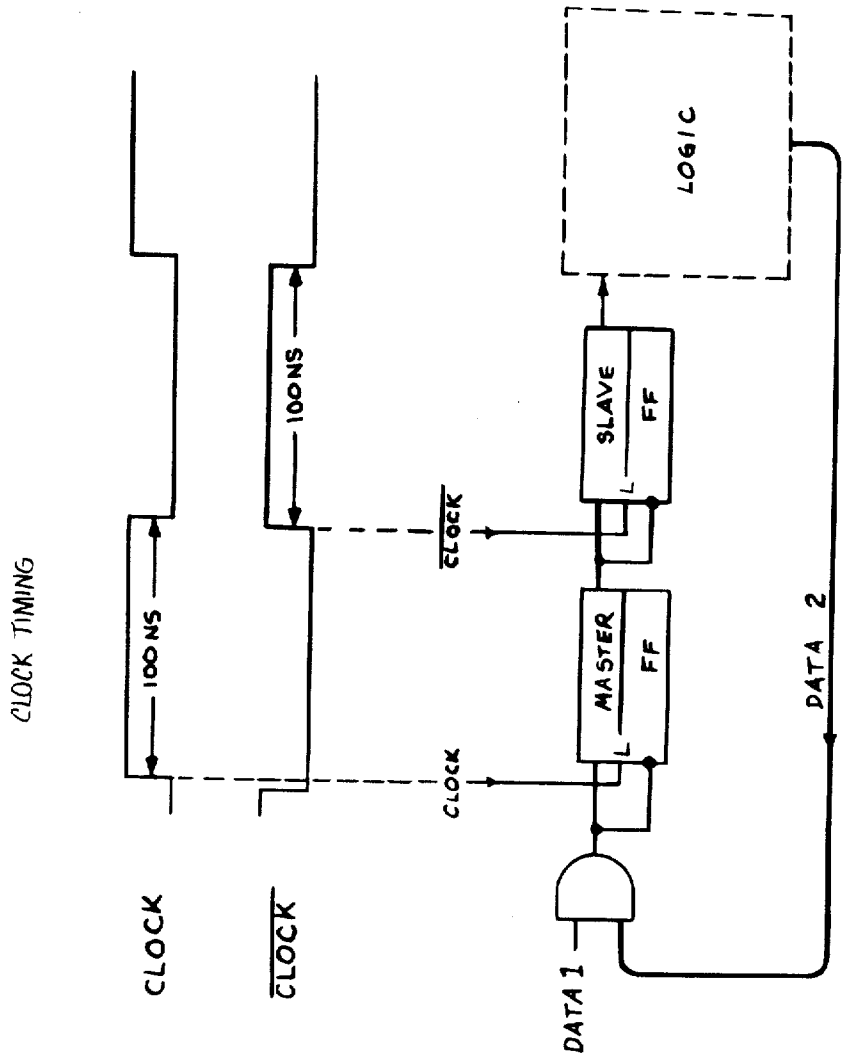

CLOCK TIMING
100NS
100NS
CLOCK
CLOCK
CLOCK
CLOCK
SLAVE
FF
MASTER
FF
LOGIC
DATA 1
DATA 2

FIG. 72

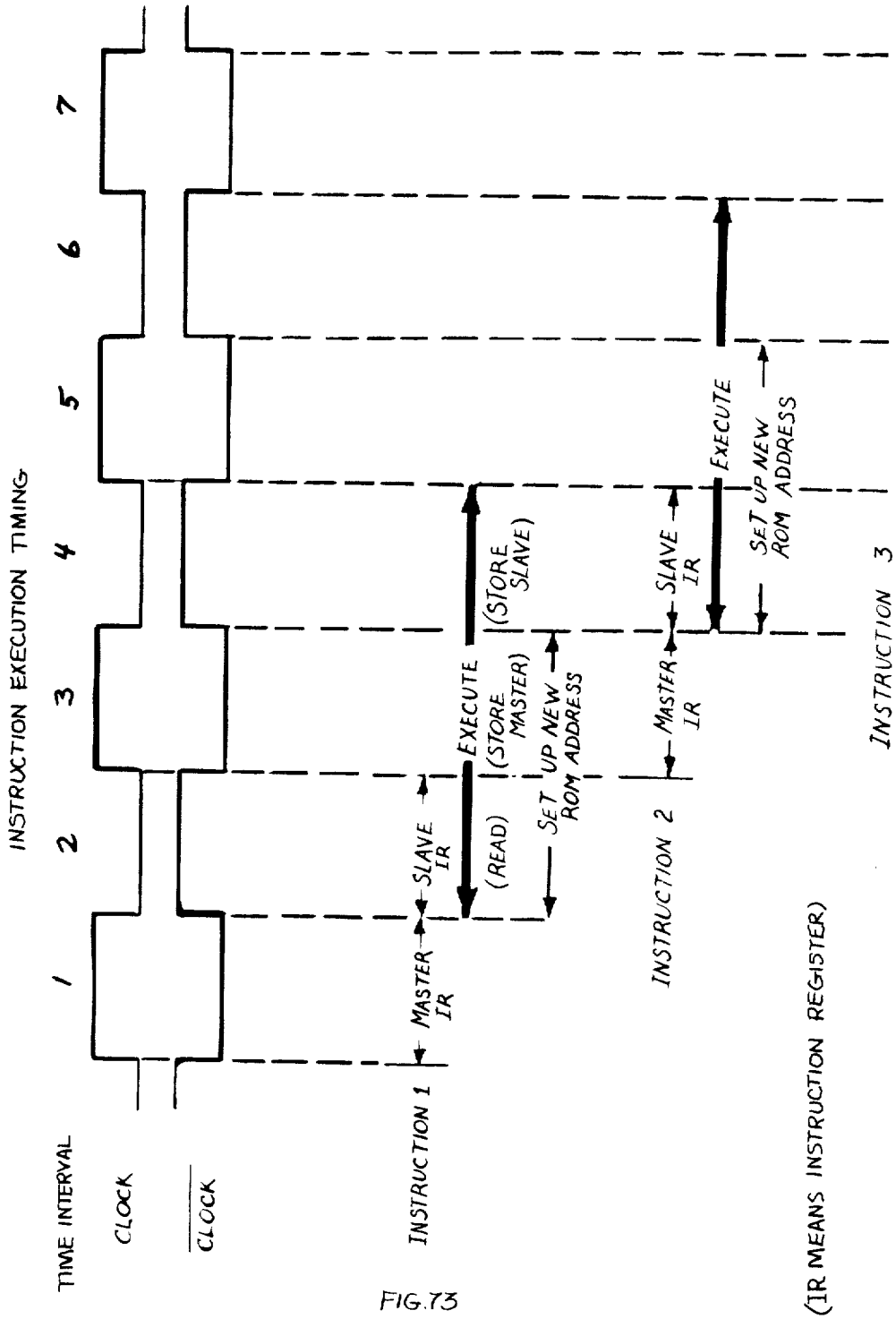

INSTRUCTION EXECUTION TIMING
TIME INTERVAL
1
2
3
4
5
6
7
CLOCK
$\overline{CLOCK}$
INSTRUCTION 1
MASTER IR
SLAVE IR
(READ)
EXECUTE
(STORE MASTER)
SET UP NEW ROM ADDRESS
(STORE SLAVE)
INSTRUCTION 2
MASTER IR
SLAVE IR
EXECUTE
SET UP NEW ROM ADDRESS
INSTRUCTION 3
(IR MEANS INSTRUCTION REGISTER)

FIG. 73

ROM INSTRUCTION CODING

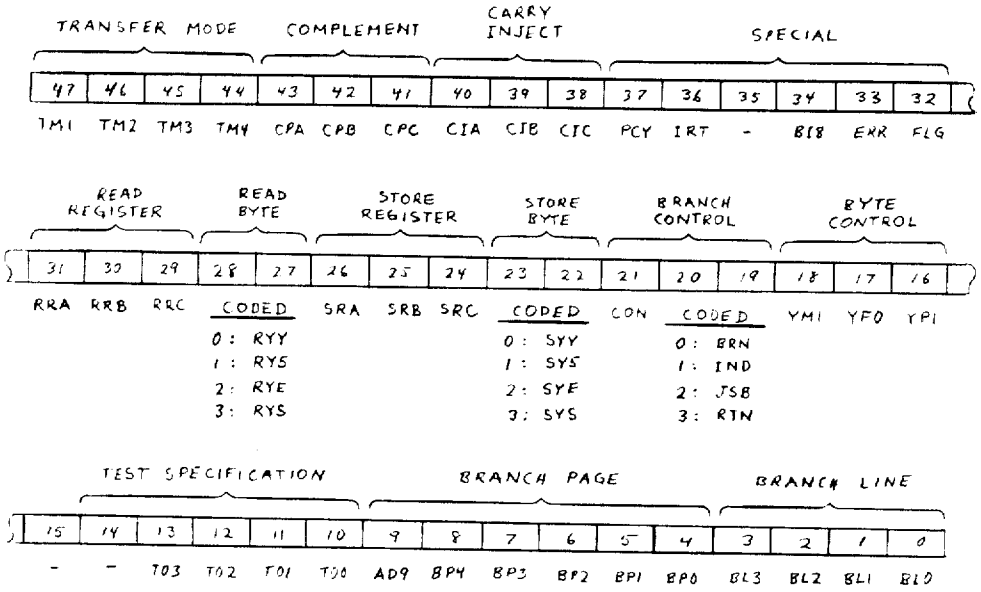



FIG. 74

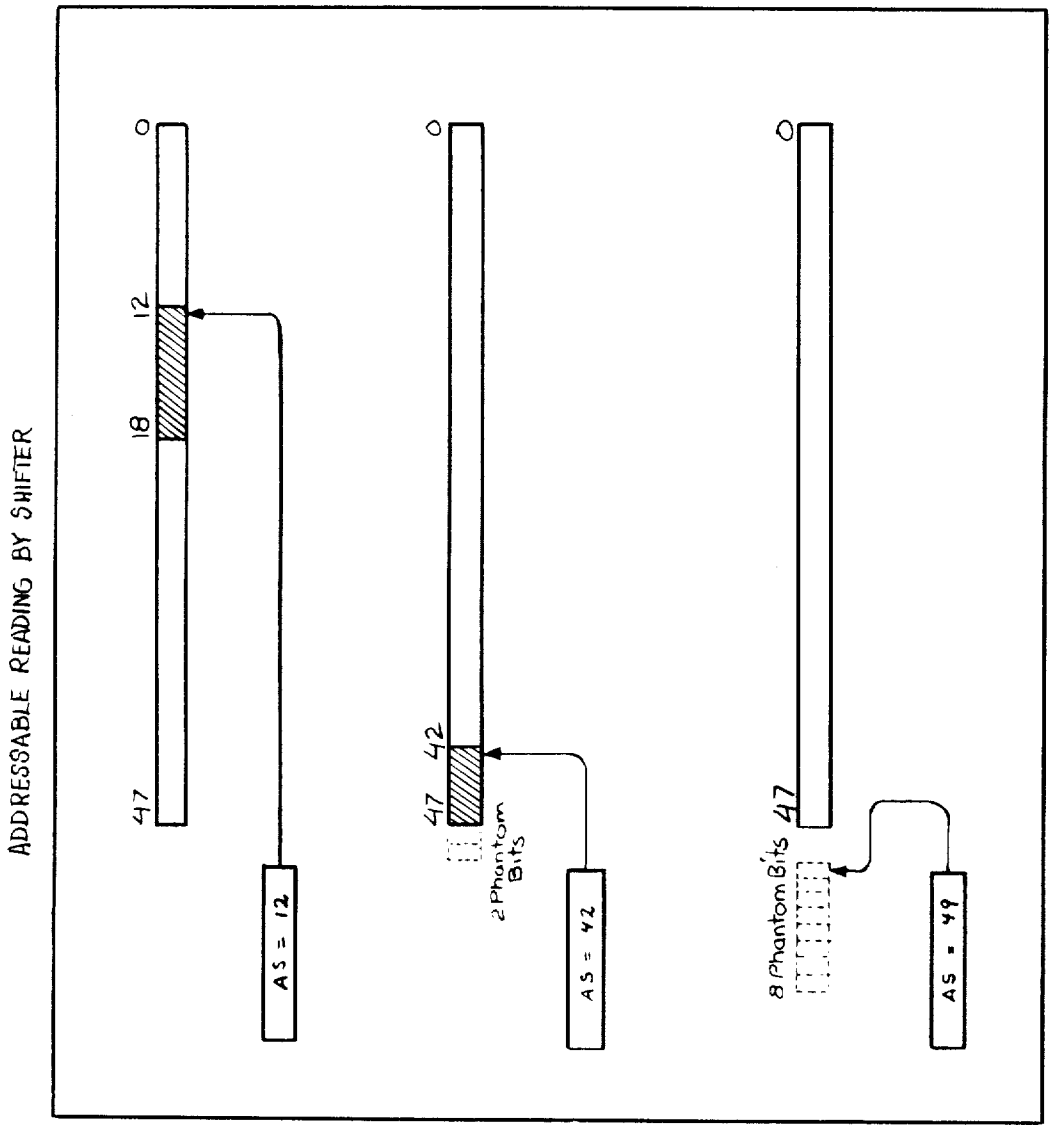
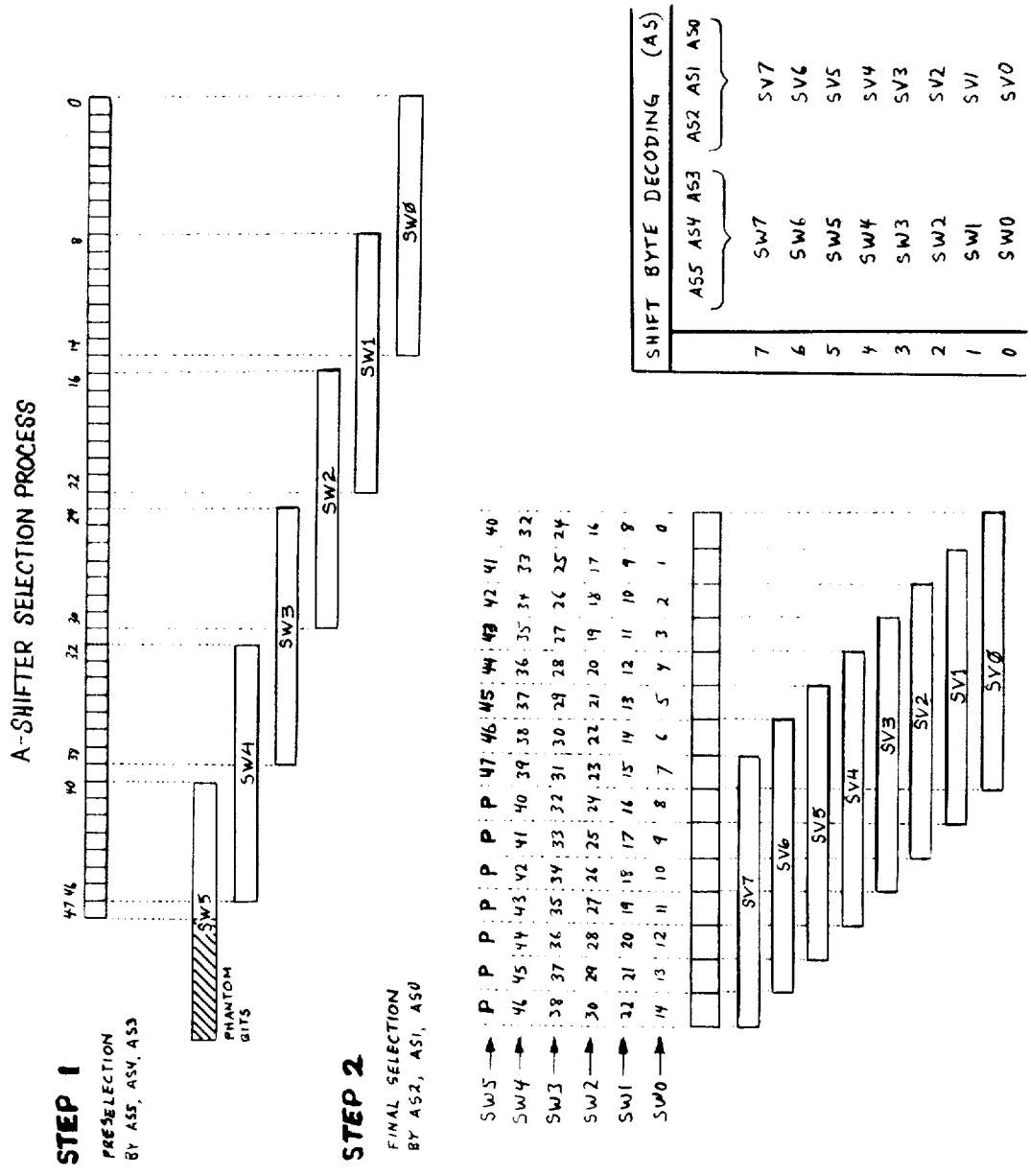

ADDRESSABLE READING BY SHIFTER
0
12
18
47
AS = 12
0
42
47
2 Phantom Bits
AS = 42
0
47
8 Phantom Bits
AS = 49

FIG. 75




| SHIFT BYTE DECODING (AS) | | | | | |
|---|---|---|---|---|---|
| | AS5 AS4 AS3 | | AS2 AS1 AS0 | | |
| 7 | SW7 | | | SV7 | |
| 6 | SW6 | | | SV6 | |
| 5 | SW5 | | | SV5 | |
| 4 | SW4 | | | SV4 | |
| 3 | SW3 | | | SV3 | |
| 2 | SW2 | | | SV2 | |
| 1 | SW1 | | | SV1 | |
| 0 | SW0 | | | SV0 | |

FIG 76

POWER SUPPLY BLOCK DIAGRAM

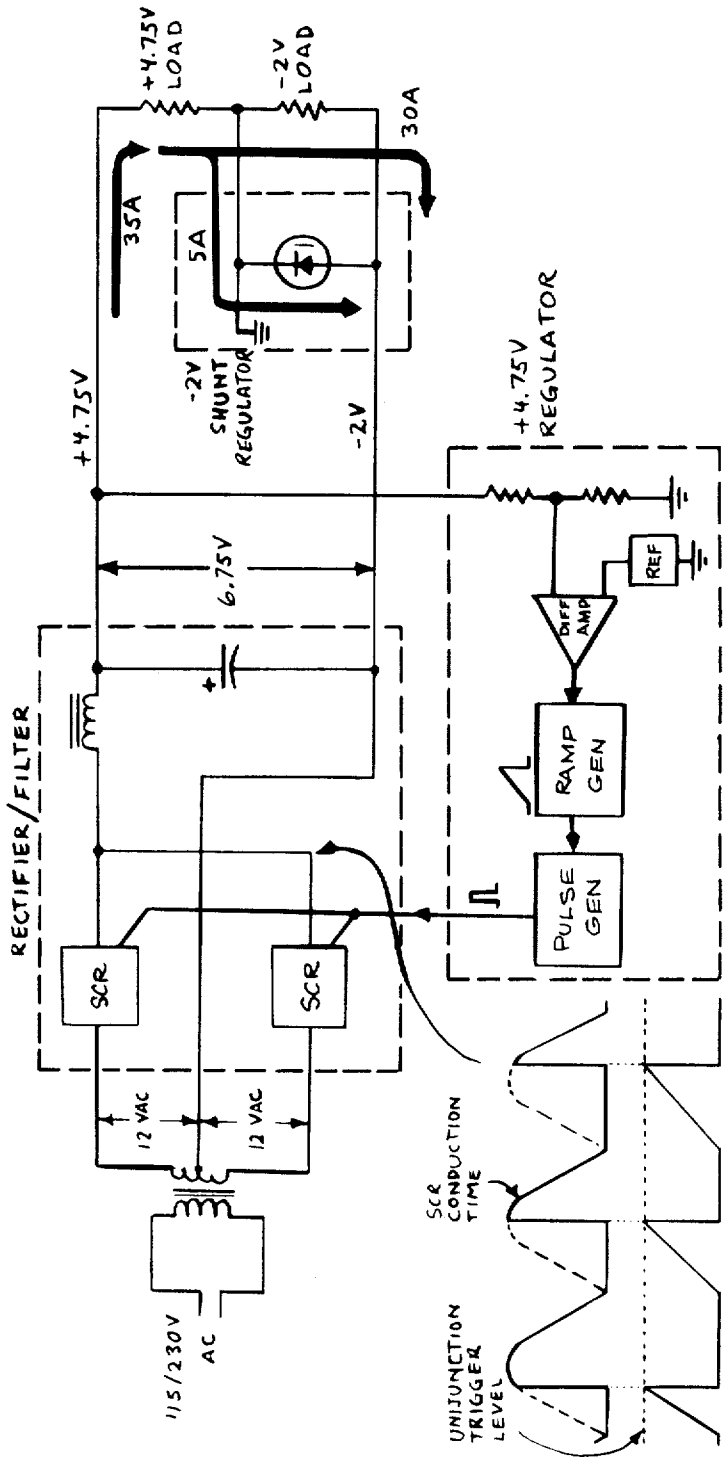



FIG 77

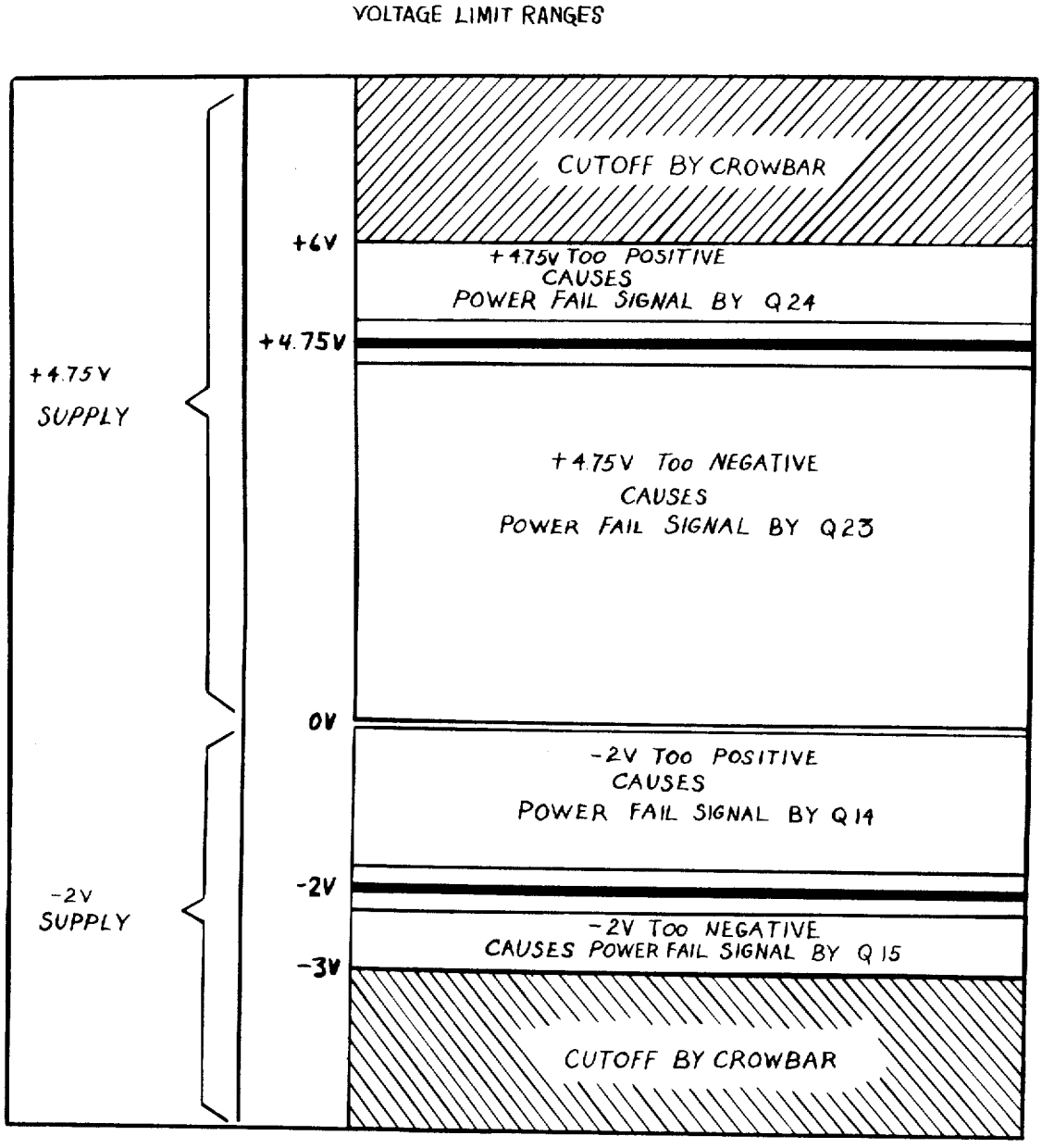

VOLTAGE LIMIT RANGES
CUTOFF BY CROWBAR
+6V
+4.75V TOO POSITIVE
CAUSES
POWER FAIL SIGNAL BY Q24
+4.75V
+4.75V
SUPPLY
+4.75V TOO NEGATIVE
CAUSES
POWER FAIL SIGNAL BY Q23
0V
-2V TOO POSITIVE
CAUSES
POWER FAIL SIGNAL BY Q14
-2V
-2V
SUPPLY
-2V TOO NEGATIVE
CAUSES POWER FAIL SIGNAL BY Q15
-3V
CUTOFF BY CROWBAR

FIG 78

LINE-FAIL SENSING

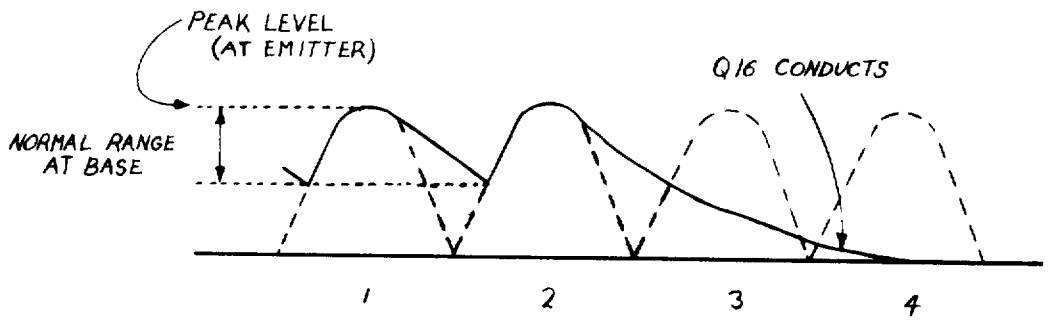



FIG. 79

# ELEMENTARY FLOATING POINT CORDIC FUNCTION PROCESSOR AND SHIFTER

## BACKGROUND OF THE INVENTION

This invention relates to a coordinate rotational digital computer (CORDIC) floating-point processor for computing elementary functions and to a data shifter for use therein.

Conventional minicomputers without floating-point hardware, such as the Hewlett-Packard 2116B, typically have a floating-point software precision of about 24 bits of mantissa and 8 bits of exponent (i.e., double precision) and a processing time of about 500 microseconds for addition, subtraction, multiplication, and division and about 10,000 microseconds for trigonometric and hyperbolic functions. Some floating-point processors are available for increasing the precision, decreasing the processing time, and extending the capability of such minicomputers. However, these floating-point processors can only perform a few elementary functions, typically limited to addition, subtraction, multiplication, division, and, in some cases, square root. A CORDIC processor is available that can perform many additional elementary functions including trigonometric functions. However, this CORDIC processor is unable to handle floating-point arguments and is inaccurate for small arguments. In addition, the present CORDIC processor employs bit-serial shifters and adder-subtractors which makes the processing time slower than desired. Parallel shifters and adder-subtractors could be employed to increase its speed. However, to do so would substantially increase the cost of the CORDIC processor since parallel shifters and adder-subtractors are much more expensive than bit-serial shifters and adder-subtractors.

## SUMMARY OF THE INVENTION

An object of this invention is to provide an improved processor for extending the hardware computing capability of a minicomputer.

Another object of this invention is to provide a faster and more accurate processor for performing an increased number of elementary functions economically.

Another object of this invention is to provide a CORDIC floating-point processor.

Another object of this invention is to provide a CORDIC floating-point processor for handling floating-point arguments and maintaining full accuracy for small arguments.

Still another object of this invention is to provide a shifter that is faster than conventional bit-serial shifters and less expensive than conventional parallel shifters.

Other and incidental objects of this invention will become apparent from a reading of this specification and an inspection of the accompanying drawings.

These objects are accomplished according to the preferred embodiment of this invention by employing three arithmetic units and three shifters operated in parallel, a read-only memory, a data storage register for storing constants read from the read-only memory, and control logic to provide a CORDIC floating-point processor that may be controlled by a microprogram stored in the read-only memory or, alternatively, by a tester. The microprogram includes a set of routines for performing the elementary functions of addition, subtraction, multiplication, division, absolute value, entier, complement, cosine, sine, tangent, arctangent, hyperbolic cosine, hyperbolic sine, hyperbolic tangent, archyperbolic tangent, exponential, natural logarithm, square root, round to nearest integer, and round to twenty-four bits. The routines for performing multiplication, division, the aforementioned trigonometric and hyperbolic functions, natural logarithm, exponential, and square root are based on a unified algorithm for performing all of these last-mentioned functions in order to simplify the control and hardware of the CORDIC floating-point processor. Each routine is optimized to perform one or more elementary functions. The control logic allows two levels of microprogrammed subroutines and permits conditional and imperative microinstructions to be executed simultaneously. Each arithmetic unit includes an adder-subtractor and an associated data storage register. The contents of these data storage registers are prescaled before performing selected elementary functions to provide an argument having a scale factor and a mantissa that falls within the domain of convergence of the unified algorithm. If the mantissa is small, it may also be prenormalized and maintained in a normalized form while the selected elementary function is performed. Certain steps of the unified algorithm for performing hyperbolic functions are repeated to maintain the convergence requirement while decreasing the processing time. Each shifter is capable of reading a fixed plural number of consecutive bits, beginning with any bit position, from an associated one of the data storage registers.



## DESCRIPTION OF THE DRAWINGS

FIG. 1 represents the angle and radius of a vector in a coordinate system parameterized by $m$.

FIG. 2 illustrates the input-output functions for CORDIC modes.

FIG. 3 is a simplified schematic diagram of a CORDIC floating point processor according to the preferred embodiment of this invention.

FIG. 4 is a simplified flow chart of the microprogram control for the CORDIC floating point processor of FIG. 3.

FIG. 5 represents different data formats including the extended floating point data format employed by the CORDIC floating point processor of FIG. 3.

FIG. 6 illustrates the ranges of extended floating point numbers.

FIG. 7 is a simplified schematic diagram illustrating the unary function routines.

FIG. 8 is a simplified schematic diagram illustrating the binary function routines.

FIGS. 9A–D are detailed block diagrams of a CORDIC floating point processor according to the preferred embodiment of this invention.

FIG. 10 is a composite figure map of FIGS. 9A–D.

FIGS. 11A–E are logic diagrams of the read-only memory portions of FIGS. 9A–D.

FIG. 12 is a composite figure map of FIGS. 11A–E.

FIG. 13 is a wiring diagram of the read-only memory of FIGS. 11A–E.

FIGS. 14A–R are logic diagrams of the read-only memory addressing portions of FIGS. 9A–D.

FIG. 15 is a composite figure map of FIGS. 14A–R.

FIGS. 16A–N are logic diagrams of the A-adder of FIGS. 9A–D.

FIG. 17 is a composite figure map of FIGS. 16A–N.

FIGS. 18A–H are logic diagrams of the A-shifter of FIGS. 9A–D.

FIG. 19 is a composite figure map of FIGS. 18A–H.

FIGS. 20A–C are logic diagrams of the transfer gates for the D-shifter.

FIG. 21 is a composite figure map of FIGS. 20A–C.

FIGS. 22A–N are logic diagrams of the B-adder of FIGS. 9A–D.

FIG. 23 is a composite figure map of FIGS. 22A–N.

FIGS. 24A–H are logic diagrams of the B-shifter of FIGS. 9A–D.

FIG. 25 is a composite figure map of FIGS. 24A–H.

FIGS. 26A–C are logic diagrams of the multiplexor for the B-shifter of FIGS. 9A–D.

FIG. 27 is a composite figure map of FIGS. 26A–C.

FIGS. 28A–N are logic diagrams of the C-adder of FIGS. 9A–D.

FIG. 29 is a composite figure map of FIGS. 28A–N.

FIGS. 30A–G are logic diagrams of the D-register of FIGS. 9A–D.

FIG. 31 is a composite figure map of FIGS. 30A–G.

FIGS. 32A–H are logic diagrams of the D-shifter of FIGS. 9A–D.

FIG. 33 is a composite figure map of FIGS. 32A–H.

FIGS. 34A–C are logic diagrams of the multiplexor for the D-shifter of FIGS. 9A–D.

FIG. 35 is a composite figure map of FIGS. 34A–C.

FIGS. 36A–G are logic diagrams of an interface card for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 37 is a composite figure map of FIGS. 36A–G.

FIGS. 38A–I are logic diagrams of a tester for use with the CORDIC floating point processor of FIGS. 9A–D.

FIG. 39 is a composite figure map of FIGS. 38A–I.

FIGS. 40A–J are schematic diagrams of the power supply for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 41 is a composite figure map of FIGS. 40A–J.

FIGS. 42A–F are block plain wiring diagrams for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 43 is a composite figure map of FIGS. 42A–F.

FIGS. 44A–D are flow charts for the entry routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 45 is a composite figure map of FIGS. 44A–D.

FIGS. 46A–C are flow charts for the entier/fix/load/-load l/round routines for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 47 is a composite figure map of FIGS. 46A–C.

FIGS. 48A–D are flow charts for the addition/subtraction/absolute/negative routines for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 49 is a composite figure map of FIGS. 48A–D.

FIGS. 50A–C are flow charts for the overflow/normalize routines for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 51 is a composite figure map of FIGS. 50A–C.

FIGS. 52A–B are flow charts for the multiply/initialize routines for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 53 is a composite figure map of FIGS. 52A–B.

FIGS. 54A–B are flow charts for the division routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 55 is a composite figure map of FIGS. 54A–B.

FIGS. 56A–D are flow charts for the sine/cosine/tangent/hyperbolic prescale routines for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 57 is a composite figure map of FIGS. 56A–D.

FIGS. 58A–C are flow charts for the sine resolver routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 59 is a composite figure map of FIGS. 58A–C.

FIGS. 60A–C are flow charts for the artangent routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 61 is a composite figure map of FIGS. 60A–C.

FIGS. 62A–D are flow charts for the hyperbolic prescale routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 63 is a composite figure map of FIGS. 62A–C.

FIGS. 64A–D are flow charts for the hyperbolic sine/hyperbolic cosine resolver routine.

FIG. 65 is a composite figure map of FIGS. 64A–D.

FIGS. 66A–D are flow charts for the natural log/hyperbolic arctangent/square root prescale routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 67 is a composite figure map of FIGS. 66A–D.

FIGS. 68A–C are flow charts for the arc hyper resolve routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 69 is a composite figure map of FIGS. 68A–C.

FIGS. 70A–B are flow charts for the diagnostic routine for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 71 is a composite figure map of FIGS. 70A–B.

FIG. 72 illustrates a clock timing diagram for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 73 illustrates the instruction execution timing diagram for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 74 represents the instruction coding of the read-only memory of the CORDIC floating point processor of FIGS. 9A–D.

FIG. 75 illustrates the addressable reading of a data storage register by an associated one of the shifter of the CORDIC floating point processor of FIGS. 9A–D.

FIG. 76 illustrates the A-shifter selection process of the CORDIC floating point processor of FIGS. 9A–D.

FIG. 77 is a block diagram of a power supply for the CORDIC floating point processor of FIGS. 9A–D.

FIG. 78 illustrates the voltage limit ranges of the power supply of FIG. 77.

FIG. 79 shows a partial filter power supply waveform used to detect power failure.

## DESCRIPTION OF THE PREFERRED EMBODIMENT

### INTRODUCTION

A floating-point processor (FPP) for extending the hardware computing capability of a minicomputer (COMP), such as the Hewlett-Packard 2116B, to include high-speed, extended-precision mathematical functions is described herein. The processing time of the FPP is in the range of 20 to 75 microseconds for floating-point addition, subtraction, multiplication, and division and in the range of 20 to 200 microseconds for the remaining mathematical functions. The precision of the FPP is 40 bits of mantissa and 8 bits of exponent (i.e., triple precision), which is equivalent to an accuracy of 12 decimal digits, and the accuracy is limited only by the truncation of input arguments. Triple-store and triple-load instructions are provided to transfer the triple-length quantities between an extended floating-point accumulator and three consecutive memory locations of the COMP. To provide compatibility with dou-

ble-precision data, instructions are also provided for performing the necessary format conversions. Both integer and floating-point double-precision data may be used.

Functionally, the FPP can be regarded as a calculator under the control of the COMP. In the same way that a human operator manually uses a free-standing calculator, the COMP enters an argument into the FPP and then issues a command to calculate a selected function of the argument. The answer appears in much less time than the COMP would have taken to calculate it. Thus, the benefits derived from the FPP by the COMP are much the same as the benefits derived from the calculator by the human operator, namely, speed and efficiency.

## A UNIFIED ALGORITHM FOR ELEMENTARY FUNCTIONS

The FPP includes a set of routines based on a single unified algorithm for the calculation of elementary functions including multiplication, division, sine, cosine, tangent, arctangent, hyperbolic sine, hyperbolic cosine, hyperbolic tangent, archyperbolic tangent, natural logarithm, exponential, and square root. The basis for the unified algorithm is coordinate rotation in a linear, circular, or hyperbolic coordinate system depending on which function is to be calculated. The only operations required are shifting, adding, subtracting, and the recall of prestored constants.

Referring to FIG. 1, the radius R and the angle A of the vector $P=(x,y)$ are defined as follows in a coordinate system parameterized by $m$:

$$R = [x^2+my^2]^{1/2} , \tag{1}$$

$$A = m^{-1/2} \tan^{-1} [m^{1/2} y/x] . \tag{2}$$

It can be shown that $R$ is the distance from the origin to the intersection of the curve of constant radius with the $x$ axis, while $A$ is twice the area enclosed by the vector, the $x$ axis, and the curve of constant radius, divided by the radius squared. The curves of constant radius for the circular ($m$=1), linear ($m$=0), and hyperbolic ($m$=−1) coordinate systems are shown in FIG. 1.

Let a new vector $P_{i+1} = (x_{i+1}, y_{i+1})$ be obtained from $P_i = (x_i, y_i)$ according to

$$x_{i+1} = x_i + m\, y_i\, \delta_i \tag{3}$$

$$y_{i+1} = y_i - x_i\, \delta_i , \tag{4}$$

where $m$ is the parameter for the coordinate system, and $\delta_i$ is an arbitrary value. The angle radius of the new vector in terms of the old are given by

$$A_{i+1} = A_i - \alpha_i \tag{5}$$

$$R_{i+1} = R_i * K_i , \tag{6}$$

where

$$\alpha_i = m^{-1/2} \tan^{-1}[m^{1/2}\, \delta_i\, ] \tag{7}$$

$$K_i = [1 + m\, \delta_i^2]^{1/2} . \tag{8}$$

The angle and radius are modified by quantities which are independent of the coordinate values. Table 1 gives the equations for $\alpha$ and K after applying identities A2 and A5 in Table 2.

TABLE 1

Angles and Radius Factors

| Coordinate System | Angle | Radius Factor |
|---|---|---|
| m | $\alpha_i$ | $K_i$ |
| 1 | $\tan^{-1}\delta_i$ | $(1+\delta_i^2)^{1/2}$ |
| 0 | $\delta_i$ | 1 |
| −1 | $\tanh^{-1}\delta_i$ | $(1-\delta_i^2)^{1/2}$ |

TABLE 2

Mathematical identities

Let $i = \sqrt{-1}$

$$z \equiv \lim_{m\to 0} m^{-1/2} \sin (z\sqrt{m}) \tag{A1}$$

$$z \equiv \lim_{m\to 0} m^{-1/2} \tan^{-1} (z\sqrt{m}) \tag{A2}$$

$$\sinh z \equiv -i \sin (iz) \tag{A3}$$

$$\cosh z \equiv \cos (iz) \tag{A4}$$

$$\tanh^{-1} z \equiv -i \tan^{-1} (iz) \tag{A5}$$

For $n$ iterations we find:

$$A_n = A_o - \alpha \tag{9}$$

$$R_n = R_o * K , \tag{10}$$

where

$$\alpha = \sum_{i=0}^{n-1} \alpha_i , \tag{11}$$

$$K = \prod_{i=0}^{n-1} K_i . \tag{12}$$

The total change in angle is just the sum of the incremental changes while the total change in radius is the product of the incremental changes.

If a third variable z is provided for the accumulation of the angle variations,

$$z_{i+1} = z_i + \alpha_i \tag{13}$$

and the set of difference equations (3), (4), and (13) is solved for $n$ iterations, we find:

$$x_n = K\{x_o \cos(\alpha\ \sqrt{m}) + y_o m^{1/2} \sin(\alpha\ \sqrt{m})\} \tag{14}$$

$$Y_n = K\{y_o \cos(\alpha\ \sqrt{m}) - x_o m^{-1/2} \sin(\alpha\ \sqrt{m})\} \tag{15}$$

$$z_n = z_o + \alpha , \tag{16}$$

where $\alpha$ and $K$ are as in equations (11) and (12). These relations are summarized in FIG. 2 for $m$=1, $m$=0 and $m$=−1 for the following special cases:

1. A is forced to zero: $y_n = 0$;

2. $z$ is forced to zero: $z_n = 0$.

The initial values $x_o$, $y_o$, $z_o$ are shown on the left of each block in FIG. 2 while the final values $x_n$, $y_n$, $z_n$ are shown on the right. The identities given in Table 2 were used to simplify these results.

By the proper choice of the initial values the functions x-z, y/x, sin z, cos z, $\tan^{-1} z$, sinh z, cosh z, and $\tanh^{-1} z$ may be obtained. In addition the following functions may be generated:

$$\tan z = \sin z/\cos z \tag{17}$$

$$\tanh z = \sinh z/\cosh z \tag{18}$$

$$\exp z = \sinh z + \cosh z \tag{19}$$

$$ln\ w = 2\ \tanh^{-1}[y/x], \text{where } x = w+1 \text{ and } y = w-1 \tag{20}$$

$$\sqrt{w} = \sqrt{x^2-y^2}, \text{ where } x = w + (1/4) \text{ and } y = w - (1/4). \tag{21}$$

The angle A of the vector P may be forced to zero by a converging sequence of rotations $\alpha_i$ which at each step brings the vector closer to the positive $x$ axis. The magnitude of each element of the sequence may be predetermined, but the direction of rotation must be determined at each step such that

$$|A_{i+1}| = \| A_i| - \alpha_i| . \tag{22}$$

The sum of the remaining rotations must at each step be sufficient to bring the angle to at least within $\alpha_{n-1}$ of zero, even in the extreme case where $A_i = 0$, $|A_{i+1}| = \alpha_i$. Thus,

$$\alpha_i - \sum_{j=i+1}^{n-1} \alpha_j < \alpha_{n-1}. \tag{23}$$

The domain of convergence is limited by the sum of the rotations:

$$|A_o| - \sum_{j=0}^{n-1} \alpha_j < \alpha_{n-1} \tag{24}$$

$$\max |A_o| = \alpha_{n-1} + \sum_{j=0}^{n-1} \alpha_j. \tag{25}$$

To show that A converges to within $\alpha_{n-1}$ of zero within $n$ steps we first prove the following theorem:

$$|A_i| < \alpha_{n-1} + \sum_{j=i}^{n-1} \alpha_j, \tag{26}$$

which holds for $i \geq 0$.

We proceed by induction on $i$. The hypothesis (26) holds for $i$=0 by (24). We now show that if the hypothesis is true for $i$ then it is also true for $i$+1. Subtracting $\alpha_i$ from (26) and applying (23) at the left side yields

$$-\left[\alpha_{n-1} + \sum_{j=i+1}^{n-1} \alpha_j\right] < -\alpha_i < |A_i| - \alpha_i < \left[\alpha_{n-1} + \sum_{j=i+1}^{n-1} \alpha_j\right]. \tag{27}$$

Application of (22) then yields

$$|A_{i+1}| < \alpha_{n-1} + \sum_{j=i+1}^{n-1} \alpha_j, \tag{28}$$

as was to be shown. Therefore, by induction, the hypothesis holds for all $i \geq 0$.

In particular, the theorem is true for $i$=$n$ so that

$$|A_n| < \alpha_{n-1} . \tag{29}$$

The same scheme may be used to force the angle in $z$ to zero. The proof of convergence proceeds exactly as before except that $A$ is replaced by $z$ in equations (22) through (29). By equation (25) $z$ has the same domain of convergence as $A$,

$$\max |z_o| = \max |A_o| . \tag{30}$$

Note that since $K$ is a function of $\delta_i^2$, hwere $\delta_i = m^{-1/2} \tan[m^{1/2} \alpha_i]$, $K$ is independent of the sequence of signs chosen for the $\alpha_i$. Thus, for a fixed sequence of $\alpha_i$ magnitudes the constant $1/K$ may be used as an initial value to counteract the factor $K$ present in the final values.

The practical use of the algorithm is based on the use of shifters to effect the multiplication by $\delta_i$. If $\rho$ is the radix of the number system and $F_i$ is an array of integers, where $i \geq 0$, then a multiplication of $x$ by

$$\delta_i = \rho^{-F_i} \tag{31}$$

is simply a shift of $x$ by $F_i$ places to the right. The integers $F_i$ must be chosen such that the angles

$$\alpha_{m,F_i} = m^{-1/2} \tan^{-1}(m^{1/2} \rho^{-F_i}) \tag{32}$$

satisfy the convergence criterion (23). The domain of convergence is then given by (25).

Table 3 shows some F sequences, convergence ranges, and radius factors for a binary code.

TABLE 3

Shift Sequences for a Binary Code

| radix $\rho$ | coordinate system m | shift sequence $F_{m,i}$, $i \geq 0$ | domain of convergence max $\|A_o\|$ | radius factor K |
|---|---|---|---|---|
| 2 | 1 | 0,1,2,3,4,i,... | ~1.74 | ~1.65 |
| 2 | 0 | 1,2,3,4,5,i+1,... | 1.0 | 1.0 |
| 2 | −1 | 1,2,3,4,4,5...* | ~1.13 | ~0.80 |

*for $m = -1$ the following integers are repeated: {4, 13, 40, 121, . . . , $k$, $3k + 1$, . . .}

The hyperbolic mode ($m = -1$) is somewhat complicated by the fact that for $\alpha_i = \tanh^{-1}(2^{-i})$ the convergence criterion (23) is not satisfied. However, it can be shown that

$$\alpha_i - \sum_{j=i+1}^{n-1} \alpha_j - \alpha_{3i+1} < \alpha_{n-1} \tag{33}$$

and that, therefore, if the integers {4, 13, 40, 121, ..., $k$, $3k+1$, ...} in the $F_i$ sequence are repeated then (23) becomes true.

The limited domain imposed by the convergence criterion (25) may be extended by means of the prescaling identities shown in Table 4. For example, to calculate the sin of a large argument, we first divide the argument by $\pi/2$ obtaining a quotient Q and a remainder D

where $|D| < \pi/2$. The table shows that only sin D 'r cos D need be calculated and that $\pi/2$ is within the domain of convergence. Note that the sine and cosine can be generated simultaneously by the CORDIC algorithm and that the answer may then be chosen as plus or minus one of these according to Q mod 4. As a second example, to calculate the logarithm of a large argument we first shift the argument's binary point E places until it is just to the left of the most significant non-zero bit. The fraction M then satisfies $0.5 \leq M < 1.0$ and as shown in the table therefore falls within the domain of convergence. The answer is calculated as $\log_e M + E^{**} \log_e 2$.

dent of the absolute size of the argument. To accomplish this for very small arguments it is necessary to keep each storage register in a normalized form; i.e., in a form where there are no leading zeros. It is possible to do this by transforming the iteration equations (3), (4), (13) to a normalized form according to the following substitutions:

$$x \text{ becomes } x' \tag{34}$$

$$y \text{ becomes } y' * 2^{-E} \tag{35}$$

TABLE 4

| Prescaling identities | Domain | Domain of convergence |
|---|---|---|
| $\sin (Q\pi/2 + D) = \begin{cases} \sin D \text{ if } Q \bmod 4 = 0 \\ \cos D \text{ if } Q \bmod 4 = 1 \\ -\sin D \text{ if } Q \bmod 4 = 2 \\ -\cos D \text{ if } Q \bmod 4 = 3 \end{cases}$ | $\lvert D\rvert < \pi/2 = 1.57$ | 1.74 |
| $\cos (Q\pi/2 + D) = \begin{cases} \cos D \text{ if } Q \bmod 4 = 0 \\ -\sin D \text{ if } Q \bmod 4 = 1 \\ -\cos D \text{ if } Q \bmod 4 = 2 \\ \sin D \text{ if } Q \bmod 4 = 3 \end{cases}$ | $\lvert D\rvert < \pi/2 = 1.57$ | 1.74 |
| $\tan (Q\pi/2 + D) = \sin (Q\pi/2 + D)/\cos (Q\pi/2 + D)$ | $\lvert D\rvert < \pi/2 = 1.57$ | 1.74 |
| $\tan^{-1} (1/y) = \pi/2 - \tan^{-1} (y)$ | $\lvert y\rvert < 1.0$ | $\infty$ |
| $\sinh (Q \log_e 2 + D) = 2^Q/2 [\cosh D + \sinh D - 2^{-2Q} (\cosh D - \sinh D)]$ | $\lvert D\rvert < \log_e 2 = 0.69$ | 1.13 |
| $\cosh (Q \log_e 2 + D) = 2^Q/2 [\cosh D + \sinh D + 2^{-2Q} (\cosh D - \sinh D)]$ | $\lvert D\rvert < \log_e 2 = 0.69$ | 1.13 |
| $\tanh (Q \log_e 2 + D) = \sinh (Q \log_e 2 + D)/\cosh (Q \log_e 2 + D)$ | $\lvert D\rvert < \log_e 2 = 0.69$ | 1.13 |
| $\tanh^{-1} (1 - M * 2^{-E}) = \tanh^{-1} (T) + E/2 * \log_e 2$ where $T = (2 - M - M * 2^{-E})/(2 + M - M * 2^{-E})$ | $0.17 < T < 0.75$ for $0.5 \leq M < 1, E \geq 1$ | $(-0.81, 0.81)$ |
| $\exp (Q \log_e 2 + D) = 2^Q (\cosh D + \sinh D)$ | $\lvert D\rvert < \log_e 2 = 0.69$ | 1.13 |
| $\log_e (M * 2^E) = \log_e M + E * \log_e 2$ | $0.5 \leq M < 1.0$ | (0.10, 9.58) |
| $\text{sqrt} (M2^E) = \begin{cases} 2^{E/2} * \text{sqrt} (M) \text{ if } E \bmod 2 = 0 \\ 2^{(E+1)/2} * \text{sqrt} (M/2) \text{ if } E \bmod 2 = 1 \end{cases}$ | $0.5 \leq M < 1.0$<br>$0.25 \leq M/2 < 0.5$ | (0.03, 2.42) |
| $(M_x * 2^{E_x}) * (M_z * 2^{E_z}) = (M_x * M_z) * 2^{E_x + E_z}$ | $0.5 \leq \lvert M_z\rvert < 1.0$ | $(-1.0, 1.0)$ |
| $(M_y * 2^{E_y}) / (M_x * 2^{E_x}) = (M_y/2M_x) * 2^{E_y - E_x + 1}$ | $0.25 \leq \lvert M_y/2M_x\rvert < 1.0$ | $(-1.0, 1.0)$ |

The accuracy at the $n^{th}$ step is determined in theory by the size of the last of the converging sequence of rotations $\alpha_i$, and for large $n$ is approximately equal in digits to $F_{n-1}$. The accuracy in digits may conveniently be made equal to $L$, the length of storage used for each variable, by choosing $n$ such that $F_{n-1} = L$.

In practice the accuracy is limited by the finite length of storage. The truncation of input arguments performed to make them fit within the storage length gives rise to unavoidable error, the size of which depends on the sensitivity of the calculated function to small changes in the input argument. In a binary code, the truncation of intermediate results after each of L iterations gives rise to a total of at most $\log_2 L$ bits of error. This error can be rendered harmless by using $L + \log_2 L$ bits for the storage of intermediate results.

In a normalized floating point number system it is desirable that all L bits of the result be accurate, indepen-

$$z \text{ becomes } z' * 2^{-E} \tag{36}$$

$$\alpha_F \text{ becomes } \alpha_F' * 2^{-F}, \tag{37}$$

where $E$, a positive integer, is chosen such that the initial argument, placed into either the $y$ or $z$ register, is normalized.

The result of the substitutions is:

$$x' \leftarrow x' + my' 2^{-(F+E)} \tag{38}$$

$$y' \leftarrow y' - x' 2^{-(F-E)} \tag{39}$$

$$z' \leftarrow z' + \alpha_F' 2^{-(F-E)}. \tag{40}$$

For simplicity the subscripts $i$ and $i+1$ have been dropped. Instead, $\alpha$ has been expressed as a function of F as in equation (32), and the replacement operator ($\leftarrow$) has been used. The value of $i$ may be initialized such that $F_i = E$:

$$i_{initial} \leftarrow \{i \mid F_i = E\} . \tag{41}$$

The value of $n$ may be chosen such that L significant bits are obtained:

$$n \leftarrow \{n \mid F_{n-1} - E = L\} . \tag{42}$$

Note that $n - i_{initial} \approx L$ and that therefore providing $L + \log_2 L$ bits for the storage of intermediate results is still adequate.

The radius factor $K$ is now a function of $i = i_{initial}$ as well as $m$,

$$K_{m,i} = \prod_{j=i}^{n-1} (1 + m2^{-2j})^{1/2}. \tag{43}$$

Fortunately, not all the reciprocal constants $1/K_{m,i}$ need to be stored since for large values of $i$

$$(1/K_{m,i}) \approx 1 - m\ (2/3)\ 2^{-2i} \tag{44}$$

and therefore all the constants having $i > L/2$ are identical to within $L$ significant bits. Therefore, only $L/2$ constants need to be stored for $m = +1$ and also for $m = -1$. For $m=0$ no constants need to be stored since $K_{0,i} = 1$ for $i \geq 1$.

A similar savings in storage can be made for the angle constants $\alpha_{m,F}$ since for large values of $F$

$$\alpha_{m,F}' \equiv \alpha_{m,F} * 2^F \approx 1 - m(1/3)\ 2^{-2F} \tag{45}$$

and,thus, as for the $K$ constants, only $L/2$ constants need to be stored for $m = +1$ and also for $m = -1$. For $m=0$ no constants need to be stored since $\mathbf{60}_{0,F}' = 1$ for $F \geq 1$.

## GENERAL DESCRIPTION

As shown in FIG. **3**, the FPP includes three identical arithmetic units **10, 12,** and **14** operated in parallel. Each arithmetic unit contains a 64-bit register **16, 18,** or **20,** an 8-bit parallel adder/subtractor **22, 24,** or **26,** and an 8-out-of-48 multiplex shifter **28, 30,** or **32.** The assembly of arithmetic units is controlled by a microprogram stored in a read-only memory (ROM) **34,** which also contains the angle and radius-correction constants. The ROM contains 512 words of 48 bits each and operates on a cycle time of 200 nanoseconds.

The essential aspects of the microprogram used to execute the unified CORDIC algorithm are shown in FIG. **4.** The initial argument and correction constants are loaded into the three registers **16, 18,** and **20,** and $m$ is set to one of the three values 1, 0, $-1$ . If the initial argument is small, it is normalized and E is set to minus the binary exponent of the result, otherwise, E is set to zero. Next, $i$ is initialized to a value such that $F_{m,i} = E$. A loop is then entered and is repeated until $F_{m,i} - E = L$. In this loop the direction of rotation necessary to force either of the angles $A$ or $z$ to zero is chosen; the binary variable $\sigma$, used to control the three adder/subtractors **22, 24,** and **26,** is set to either $+1$ or $-1$; and the iteration equations in block **36** or **38** are executed.

Table 5 gives a breakdown of the maximum execution times for the routines performed by the FPP. The figures in the column marked "data transfers from computer " are the times for operand and operation code transfers between the FPP and the COMP and vary depending upon the COMP used and how it is interfaced with the FPP. The FPP retains the result of each executed function. Thus, the binary functions add, subtract, multiply and divide require only one additional operand to be supplied, and the unary functions do not require any operand transfers. The first operand is loaded via the LOAD instruction, and the final result is retrieved via the STORE instruction.

TABLE 5.—ROUTINES AND MAXIMUM EXECUTION TIMES*

| Routine | Mnemonic | Cordic execution (μsec) | Prescale normalize, misc. (μsec) | Data transfers from computer (μsec) | Total (μsec) |
|---|---|---|---|---|---|
| Load | LDX | 0 | 5 | 25–80 | 30–85 |
| Store | STX | 0 | 0 | 15–70 | 15–70 |
| Add | ADX | 0 | 15 | 25–80 | 40–95 |
| Subtract | SBX | 0 | 25 | 25–80 | 50–95 |
| Multiply | MPX | 60 | 15 | 25–80 | 100–155 |
| Divide | DVX | 60 | 15 | 25–80 | 100–155 |
| Sine | SNX | 70 | 85 | 5–10 | 160–165 |
| Cosine | CSX | 70 | 85 | 5–10 | 160–165 |
| Tangent | TNX | 130 | 85 | 5–10 | 220–225 |
| Arctangent | ATX | 70 | 15 | 5–10 | 90–95 |
| Hyperbolic sine | HSX | 70 | 55 | 5–10 | 130–135 |
| Hyperbolic cosine | HCX | 70 | 55 | 5–10 | 130–135 |
| Hyperbolic tangent | HTX | 130 | 55 | 5–10 | 190–195 |
| Archyperbolic tangent | AHT | 70 | 45 | 5–10 | 120–125 |
| Exponential | EXX | 70 | 55 | 5–10 | 130–135 |
| Natural logarithm | LNX | 70 | 45 | 5–10 | 120–125 |
| Square-root | SRX | 70 | 25 | 5–10 | 100–105 |
| Absolute value | ABX | 0 | 15 | 5–10 | 20–25 |
| Entier | ENX | 0 | 12 | 5–10 | 17–22 |
| Complement | CMX | 0 | 15 | 5–10 | 20–25 |
| Round to nearest integer | FIX | 0 | 6 | 5–10 | 11–16 |
| Round to 24 bits | RNX | 0 | 9 | 5–10 | 14–19 |

*The execution times shown assume all operands are normalized; if otherwise, add 12 microseconds per operand.

## PROGRAMMING INFORMATION

Table 6 lists the operation codes for each of the 20 FPP routines.

TABLE 6

Cordic Floating-Point Processor Operation Codes

| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| TRIPLE LOAD AND STORE | LDX | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| | STX | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| TRIPLE PRECISION FLOATING POINT ARITHMETIC | ADX | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| | SBX | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| | MPX | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| | DVX | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| | ABX | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| | ENX | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| | CMX | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| | CSX | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| | SNX | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| | TNX | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| FUNCTIONS | ATX | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| | HCX | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| | HSX | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| | HTX | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| | AHT | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| | EXX | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| | LNX | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | SRX | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| | FIX | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| | RNX | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |

The FPP receives the data from the COMP in a normalized or un-normalized extended floating-point format and returns data in a normalized extended floating-point format to the COMP. As shown in the upper box of FIG. 5, a typical traditional floating-point format employs 23 bits for the mantissa, 1 bit for the mantissa sign, 7 bits for the exponent, and 1 bit for the exponent sign. This requires two 16-bit words. Note that the second word is split, such that bits 8 through 15 are the eight least significant bits of the mantissa, and the remaining eight bits are the exponent and exponent sign. As shown in the middle box of FIG. 5, the only difference between the traditional double-word floating point format and the extended floating-point format employed by the FPP is the addition of one 16-bit word to the length of the mantissa.

As shown in the lower box of FIG. 5, the conversion from the traditional double-word floating-point format to the extended triple-word floating-point format is accomplished by splitting the second word of the double-word format between bits 8 and 7, and inserting 16 zeros as the least significant bits of the mantissa in the triple-word format. Note that 8 of these zeros are present in the second word of the triple-word format, and the remaining 8 are in the third word. The reverse conversion, from triple- to double-length format, consists simply of truncating the 16 least significant bits of the mantissa. This means removing bits 7 through 0 of the second word, and bits 15 through 8 of the third word.

The conversions between double-word integer and extended floating point formats (not illustrated) are more complex. Briefly, the process is as follows. For conversion to double-word integer, the mantissa is arithmetically shifted right while the exponent value is correspondingly increased (one increment per shift) until the exponent equals +31. If bit 15 is a 1 (implying ½), the quantity in bits 16 through 47 is incremented by 1; this rounds the integer to the nearest whole number. Bits 8 through 15 are set to 0, and bits 16 through 47 comprise the integer value. The reverse conversion, double-word integer to extended floating point, consists of filling in zeros for bits 8 through 15 of the least significant word, setting the exponent to +31, and then normalizing the result.

From the standpoint of the COMP programmer, the FPP effectively adds an extended floating-point accumulator, designated the X register. This register can be loaded from the COMP memory, its contents manipulated, and its contents stored in the COMP memory. Each extended floating-point operand may be contained in three consecutive COMP memory locations, as assumed for the following definitions of the twenty FPP routines:

LDX (LOAD EXTENDED FLOATING POINT). Load the extended floating point number from addressed memory location $m$ (and $m + 1$ and $m + 2$) into the X-register.

STX (STORE EXTENDED FLOATING POINT). Store the extended floating point number in the X-register in addressed memory location $m$ (and $m + 1$ and $m + 2$).

ADX (ADD EXTENDED FLOATING POINT). Add the extended floating point number from addressed memory location $m$ (and $m + 1$ and $m + 2$) to the current value in the X-register. The extended floating point result of the addition occupies the X-register on completion of the instruction.

SBX (SUBTRACT EXTENDED FLOATING POINT). Subtract the extended floating point number in addressed memory location $m$ (and $m + 1$ and $m + 2$) from the current value in the X-register. The extended floating point result of the subtraction occupies the X-register on completion of the instruction.

MPX (MULTIPLY EXTENDED FLOATING POINT). Multiply the extended floating point number in the X-register by the extended floating point number in addressed memory location $m$ (and $m + 1$ and $m + 2$). The extended floating point result of the multiplication occupies the X-register on completion of the instruction.

DVX (DIVIDE EXTENDED FLOATING POINT). Divide the extended floating point number in the X-register by the extended floating point number in addressed memory location $m$ (and $m + 1$ and $m + 2$). The extended floating point result of the division occupies the X-register on completion of the instruction.

ABX (ABSOLUTE VALUE). Calculate the absolute value of the extended floating point value in the X-register; i.e., if the content of the X-register is negative, convert to positive.

ENX (ENTIER). Calculate the entier of the extended floating point value in the X-register. The calculated result replaces the original contents of the X-register.

CMX (COMPLEMENT). Convert the extended floating point value in the X-register to an extended floating point value with unchanged magnitude but opposite sign.

CSX (COSINE). Calculate the cosine of the value in the X-register, where the value is expressed in radians as an extended floating point number. The result of the cosine calculation replaces the original value in the X-register.

SNX (SINE). Calculate the sine of the value in the X-register, where the value is expressed in radians as an extended floating point number. The result of the sine calculation replaces the original value in the X-register.

TNX (TANGENT). Calculate the tangent of the value in the X-register, where the value is expressed in radians as an extended floating point number. The result of the tangent calculation replaces the original value in the X-register.

ATX (ARCTANGENT). Calculate the arctangent of the value in the X-register, where the result is expressed in radians as an extended floating point number in the X-register.

HCX (HYPERBOLIC COSINE). Calculate the hyperbolic cosine of the extended floating point number in the X-register. The result of the hyperbolic cosine calculation replaces the original value in the X-register.

HSX (HYPERBOLIC SINE). Calculate the hyperbolic sine of the extended floating point number in the X-register. The result of the hyperbolic sine calculation replaces the original value in the X-register.

HTX (HYPERBOLIC TANGENT). Calculate the hyperbolic tangent of the extended floating point number in the X-register. The result of the hyperbolic tangent calculation replaces the original value in the X-register.

AHT (ARCHYPERBOLIC TANGENT). Calculate the archyperbolic tangent of the extended floating point number in the X-register. The result of the archyperbolic tangent calculation replaces the original value in the X-register.

EXX (EXPONENTIAL). Calculate the exponential of the extended floating point number in the X-register. The result of the exponential calculation replaces the original value in the X-register.

LNX (NATURAL LOGARITHM). Calculate the natural logarithm of the extended floating point number in the X-register. The result of the logarithm calculation replaces the original value in the X-register.

SRX (SQUARE ROOT). Calculate the square root of the extended floating point number in the X-register. The result of the square root calculation replaces the original value in the X-register.

FIX (ROUND TO NEAREST INTEGER). Round-off the extended floating point number in the X-register to the nearest integer value. The result remains an extended floating point number and replaces the original value in the X-register. The number of bits affected depends on the exponent value.

RNX (ROUND TO 24 BITS). Round-off the extended floating point number in the X-register to 24 bits of precision.

Internally, the FPP processes all data as normalized extended floating point numbers and can convert the input data if the data is not already in this form. As shown in Table 7, normalization is accomplished by shifting the mantissa left to eliminate any zeros between the binary point and the first non-zero bit, while the exponenet is correspondingly reduced by subtracting 1 for each shift. In the upper example of Table 7, there are two zeros between the binary point and the 1 bit. Therefore two left shifts are necessary, and the exponent is reduced from −121 to −123. In the lower example, the number is too small to be normalized, resulting in an underflow condition. The mantissa is shown shifted left seven positions, which reduces the exponent to its smallest possible value, −128. No further left shifts can therefore be made, and there is still one remaining zero between the binary point and the 1 bit. If a number is too large to be represented in the normalized extended floating point format, an overflow condition results.

TABLE 7

| NORMALIZATION | |
|---|---|
| Mantissa (Binary representation) | Binary Exponent (Decimal representation) |
| + .0010000000 . . . 000 | −121 |
| + .1000000000 . . . 000 | −123 |
| UNDERFLOW (Cannot be normalized) | |
| Mantissa (Binary representation) | Binary Exponent (Decimal representation) |
| + .0000000010 . . . 000 | −121 |
| + .0100000000 . . . 000 | −128 |

FIG. 6 defines the ranges of valid binary numbers that can be processed by the FPP. The shaded areas define overflow and underflow ranges. In the VALUE column, the mantissa is enclosed in parentheses, followed by the exponent outside of the parentheses. Note that 0 (middle line of the figure) is represented by all zeros in both the mantissa and the exponent. Positive numbers are shown above this line, and negative numbers are shown below this line. In the far right column, the exponent values are shown increasing in both directions from the zero line, from the smallest representable value (−128) to the largest (+127). Nonrepresentable exponent values, smaller than −128, are also underflow conditions, but this situation is not considered in FIG. 6.

For each finite value of exponent, the mantissa is assumed to go through its complete cycle of valid values. In approximate terms, the positive mantissa cycles from +½ to +1, and the negative mantissa cycles from −½ to −1. More precisely, the positive range is from +½ to the largest possible fractional number under the value of 1. The negative range is from the largest possible fractional number below (more negative than) −½ to −1.

The significance of +½ and −½ in determining mantissa ranges is a result of the normalization requirement, which dictates that there will be a significant digit immediately to the right of the binary point. This automatically eliminates all numbers between +½ and −½, including the exact value of −½, but excluding 0 and +½.

If an error condition results during the execution of an FPP routine, the FPP provides an error code and an error control signal. Table 8 lists the possible types of error that can occur for each of the 20 routines of the FPP.

TABLE 8.—ERROR CONDITIONS

| Routine | Possible errors: Code | Error | Condition (See note 1) | Error effect on X-Register contents |
|---|---|---|---|---|
| LDX | | None | | |
| STX | | None | | |
| ADX | 1 | Underflow | See note 2 | Correct, except exponent sign bit is complemented. |
| | 2 | Overflow | See note 3 | Do. |
| SBX | 1 | Underflow | See note 2, or $x = (1/2)\ 2^{-128}$. | Do. |
| | 2 | Overflow | See note 3, or $x = -2^{128}$. | Do. |
| MPX | 1 | Underflow | See note 2 | Do. |
| | 2 | Overflow | See note 3 | Do. |
| DVX | 1 | Underflow | See note 2 | Do. |

TABLE 8.—ERROR CONDITIONS—Continued

| Routine | Possible errors: Code | Error | Condition (See note 1) | Error effect on X-Register contents |
|---|---|---|---|---|
| | 2 | Overflow | See note 3 | See note 4. |
| | 4 | Divided by zero | Divisor = 0 | Unchanged. |
| ABX | 2 | Overflow | $x = -2^{127}$ | Same as ADX underflow. |
| ENX | | None. | | |
| CMX | 1 | Underflow | $x = (1/2)2^{-128}$ | Same as ADX underflow. |
| | 2 | Overflow | $x = -2^{127}$ | Do. |
| CSX | 3 | No resolution | $\|x\| \geq 2^{38}$ | Unchanged. |
| SNX | 3 | .....Do | $\|x\| \geq 2^{38}$ | Do. |
| TNX | 3 | .....Do | $\|x\| \geq 2^{38}$ | Do. |
| TNX | 4 | Divided by zero | $x = (2k+1)\pi/2$ for $k = 0, \pm 1, \pm 2 \ldots$ | $z = \sin x$. |
| ATX | | None | | |
| HCX | 1 | Underflow | $x < -88$ | Same as ADX underflow. |
| | 2 | Overflow | $x > 88$ | Not predictable. |
| HSX | 1 | Underflow | $x < -88$ | Same as ADX underflow. |
| | 2 | Overflow | $x > 88$ | Not predictable. |
| HTX | | None | | |
| AHT | 5 | Improper variable | $\|x\| \geq 1$ | Unchanged, unless $x = -1$; then contents will be 0. |
| EXX | 1 | Underflow | $x < -87.3$ | Same as ADX underflow. |
| | 2 | Overflow | $x > 87.3$ | Not predictable. |
| LNX | 5 | Improper variable | $x \leq 0$ | Unchanged. |
| SRX | 5 | Improper variable | $x < 0$ | Do. |
| FIX | 2 | Overflow | $x < -2^{31}$, or $x \geq 2^{31}$ | Not predictable. |
| RNX | 2 | Overflow | $z = 2^{127}$ | Same as ADX underflow. |

NOTES: 1. $x$ = Contents of X-register before execution. p1 $z$ = Contents of X-register after execution.

2. Underflow condition if: $0 < z < (\frac{1}{2})2^{-128}$ or $0 > z \geq (-\frac{1}{2})2^{-128}$

3. Overflow condition if: $z \geq 2^{127}$ or $z < -2^{127}$

4. Answers will be correct except for the sign bit of the exponent which will be complemented. If the dividend exponent is +126 or +127 and the divisor exponent is −127 or −128, the resultant exponent will be incorrect. In this case, the exponent value will equal the original value of the dividend exponent, plus two. This produces a rollover to an apparent negative exponent of −128 (if the original was +126) or −127 (if the original was +127). The SBX routine also checks the subtrahend from memory before execution begins. If the subtrahend is at the maximum allowed negative value [i.e., $(-1)2^{127}$ in FIG. 6], an overflow will result when the number is converted to positive form during execution. Similarly, the CMX and ABX instructions check the pre-execution contents of the X-register for the maximum allowed negative value; overflow will result when the number is converted. However, in the SBX, CMX, and ABX routines execution will proceed, and only the sign bit of the exponent will be incorrect (complemented). The RNX routine will indicate an overflow condition if the number is at the maximum allowed positive value and then is rounded upward. This will cause rollover to the maximum negative number. Overflows resulting from the FIX, HCX, HSX, and EXX routines produce results which are generally unpredictable, making it difficult or impossible to reconstruct correct answers.

The FPP add, subtract, multiply, and divide routines will indicate an underflow error condition if the result cannot be normalized, i.e., the result falls in one of the shaded areas immediately adjacent to zero in FIG. 6. If the input data is not normalized, the FPP will normalize the operands before beginning the computation; if the numbers are too small to be successfully normalized, the FPP will attempt normalization as far as possible and then proceed with the computation. The answer will be correct except that the sign bit of the exponent will be incorrect (complemented). The FPP will indicate an underflow error condition for CMX if the pre-execution contents of the X-register (i.e., $x$) equals $(\frac{1}{2})2^{-128}$, for HCX or HSX if $x$ is less than −88, and for EXX if $x$ is less than −87.3. The SBX routine checks if the subtrahend from memory has the minimum allowed positive value (i.e., $(\frac{1}{2})2^{-128}$ in FIG. 6); this would produce an underflow when converted to negative form during execution. However, as in all underflow conditions, the computation will proceed, and only the sign bit of the exponent will be incorrect.

The FPP add, subtract, multiply, and divide routines will indicate an overflow error condition if the result exceeds the largest positive or negative number which can be represented; i.e., the result falls in either the top or bottom shaded areas in FIG. 6. Answers will be correct except for the sign bit of the exponent, which will be complemented. However, the divide routine DVX can produce one exception to this general rule. If the dividend exponent is +126 or +127 and the divisor exponent is −127 or −128, the resultant exponent will be incorrect. In this case, the exponent value will equal the original value of the dividend exponent, plus two. This produces a rollover to an apparent negative exponent of −128 (if the original was +126) or −127 (if the original was +127). The SBX routine also checks the subtrahend from memory before execution begins. If the subtrahend is at the maximum allowed negative value [i.e., $(-1)2^{127}$ in FIG. 6], an overflow will result when the number is converted to positive form during execution. Similarly, the CMX and ABX instructions check the pre-execution contents of the X-register for the maximum allowed negative value; overflow will result when the number is converted. However, in the SBX, CMX, and ABX routines execution will proceed, and only the sign bit of the exponent will be incorrect (complemented). The RNX routine will indicate an overflow condition if the number is at the maximum allowed positive value and then is rounded upward. This will cause rollover to the maximum negative number. Overflows resulting from the FIX, HCX, HSX, and EXX routines produce results which are generally unpredictable, making it difficult or impossible to reconstruct correct answers.

The CSX, SNX, and TNX routines allow the arguments to include multiple rotations of the angle represented by the argument. These rotations use up part of the available floating point bits, leaving fewer bits to express the fractional part of the rotation for the computation. When the number of rota-tions reaches about $2^{38}/2\pi$, there is insufficient resolution to express angles in increments smaller than 90 degrees. The no-resolution error code 3 indicates this condition.

If the divisor for the DVX routine is zero, the division will not be attempted, and the error code 4 will indicate this condition. In the TNX routine, odd numbers of quarter rotations (¼, ¾, etc.) result in a divide-by-zero condition (sin = 1, cos = 0). This is also indicated by the error code 4. The value 1 will remain in the X-register on exit from this routine.

Attempts to calculate the square root (SRX) of negative numbers, or the natural logarithm (LNX) of zero or negative numbers, will not be executed, and will leave the X-register unchanged. Attempts to calculate

the archyperbolic tangent (AHT) of numbers which are ±1 or greater in magnitude also will not be executed and will leave the X-register unchanged with the exception that the attempt to calculate the archyperbolic tangent of −1 will result in clearing the X-register to zero.

Undefined operation codes given to the FPP will indicate error code 6 and will otherwise be ignored.

The ENX, RNX, and FIX are explained in the following paragraphs.

Entier (ENX) is simple truncation of the fractional part of a number. This is accomplished by noting the value of the exponent and saving that number of places in the most significant part of the mantissa. The remalning bits of the mantissa are cleared (to zeros). The effect for positive numbers is to reduce the X-register contents (if there is a fraction) to the nearest integer value and for negative numbers is to increase the X-register contents (even if there is no fraction) to the nearest negative integer value.

Rounding an extended floating point number to 24 bits of significance (RNX) implies that a specific number of bits (16) will be cleared. Before these bits are cleared, however, the most significant bit of those to be cleared is examined. If this bit is a 1, indicating for positive numbers that the part of the number to be cleared represents a value of ½ (or more) of the least significant bit of the saved part, the saved part is incremented by +1. Thus, if the cleared part of a positive number is ½ or greater, the value is rounded upward (not necessarily to an integer); otherwise, the value is rounded downward. If the cleared part of a negative number is −½ or more negative, the value is rounded in the negative direction; otherwise, the value is rounded in the positive direction. Note that if all 23 significant data bits were 1's before execution, rounding upward (incrementing by +1) would cause overflow; unless an overflow condition exists, the instruction will automatically renormalize the number.

Rounding an extended floating point number to the nearest integer (FIX) involves two steps: conversion to integer, and rounding. First, the mantissa is shifted right while the exponent value is incremented, once per shift, until the exponent equals +31. Then the most significant bit of the eight bits to be cleared is examined. If this bit is a 1, the saved part is incremented by +1. For positive numbers, the 1 bit indicates a fraction of ½ or greater; for negative numbers, it indicates a fraction smaller than ½. After rounding, bits 8 through 15 are cleared. If the cleared part of a positive number is ½ or greater, the saved part is rounded to the next higher integer; otherwise the number is reduced to the integer-only value. If the cleared part of a negative number is −½ or more negative, the value is rounded to the next more negative integer; otherwise the number becomes the integer-only value.

## THEORY OF OPERATION

The floating point processor has two basic modes of operation, one for binary function routines, and one for unary function routines. FIGS. **7** and **8** illustrate these two basic modes of operation.

In FIGS. **7** and **8**, the three registers, A, B, C, together comprise what was earlier called, for simplicity, the X-register. Each of the three registers has 48 bits for data (mantissa), and also has 8 bits for an exponent byte (E) and 8 bits for a shift control byte (S). Since the C-register has no facilities for shifting, the C shift control byte is used to receive the operation code from the COMP. The shaded areas indicate the location of operands at the start of the operations.

Referring now to FIG. **7**, an operand quantity $x$, previously loaded into the FPP, is assumed to exist in the FPP B-register. The COMP places an operation code on interface data lines **40** and issues an OPC command to the FPP via line **42**. (An OPC command initiates the reception of an operation code through the 16 bit input port of the FPP and initiates the execution of that operation.) The FPP, which operates under control of firmware programs in the ROM, cycles in a wait mode as long as it is in the ready state, looking for an OPC command. When the ROM program detects the presence of OPC, it loads the operation code data into the S byte of the FPP C-register. Then the microprogram identifies the type of operation by decoding the operation code bits, and branches to the appropriate function routine. The operation code is now no longer needed.

As the microprogram proceeds, the value in the FPP B-register is manipulated according to the algorithm for the particular function, using all three registers. At the end of the routine, the final answer resides in the FPP B-register, and a FLG (Flag) signal is sent back to the COMP. The FLG signal indicates, if following an OPC command, that the FPP is ready for further commands, having completed the last issued command.

If an error occurs during the calculation, or if the operation code is improper, an ERR (Error) signal is sent back to the COMP with the FLG signal. It is the user's option to decide what to do about an error condition. In general, it may be said that the FPP will attempt the calculation, rather than abort, even if input values will result in an error. The FPP will provide the best answer it can, along with the error indication. This allows the programmer some flexibility to reconstruct correct answers from results which normally could not be represented. The exception to this generalization is that divisions by zero will not be attempted.

Referring now to FIG. **8**, the binary function routines take two operands, one that was previously loaded into the FPP, and one that exists in the COMP memory, and operate on these operands. The operations include addition, subtraction, multiplication, and division.

Initially the quantity $x$ is assumed to exist in the FPP B-register. It may have been left there as the result of a previous instruction, or it may have been loaded by a load instruction LDX. The COMP first fetches one word of a three-word operand ($y$) from memory. It then puts this data word on the interface data lines and issues an ENC command to the FPP unit. An ENC command initiates the reception of an operand word through the 16 bit input port and the transmission of a result word through the 16 bit output port.

As mentioned previously, the FPP unit, under control of the microprogram, continuously searches for ENC or OPC commands as long as it is in the ready state. When the microprogram detects the presence of ENC, it loads the data word (in two 8-bit bytes) into the high order third of the FPP C-register.

After the second byte has been loaded, the FPP unit sends a FLG signal back to the COMP, indicating readiness for the next word of $y$. The COMP fetches this next word from its memory and repeats the pro-

cess: the word is placed on the interface data lines **40**, an ENC command is given to the FPP to load these two bytes, and another FLG signal is returned to repeat the process for the third and final time.

At the end of the three-word transfer, the quantity *x* is in the FPP B-register and the quantity *y* is in the FPP C-register. The FPP unit now needs to be told what to do with these numbers. The entire process described above under function routines is now added on. In brief, the procedure is:

a. The COMP issues an operation code and OPC control signal.

b. The FPP unit loads the operation code into the FPP C-register.

c. The ROM program interprets the operation code and branches to the appropriate function routine (add, subtract, etc.).

d. The function routine calculates the answer, and leaves it in the FPP B-register.

e. A final FLG returned to the COMP tells it that the FPP unit is ready for further commands.

f. In case of error, an ERR signal indicates that an error code is present on the data lines to the COMP.

The load instruction LDX operates similarly to the above procedure, except that the operation code simply causes the loaded FPP C-register contents to move up into the FPP B-register.

The OPC command, or ENC command, should only be given (set to 1) to the FPP if FLG from FPP is 1. The ENC command should not go to 0 until FLG from FPP goes to 0. The ENC or OPC command should not go to 1 again until FLG goes to 1.

The following describes in detail the procedures described above, plus a description of the FPP power supply. All logic is positive-true. The high (or true) state ranges from +1.25 to +2.5 volts; the low (or false) state ranges from −0.5 to +0.5 volts.

The block diagrams of FIGS. **9A–D** are referenced throughout this description. Tables **9** and **10** provide supporting information: the detailed coding of the instruction register (IR), definitions of the ROM instructions, and a list of tests used for branching decisions. These tables and FIGS. **9A–D** should be referred to frequently, since definitions will not be given within the descriptive text. The logic diagrams for the FPP are given in FIGS. **11–40**, a wiring diagram is given in FIGS. **42A–F**, and flow charts for the FPP routines are given in FIGS. **44–71**. Specific signal names are given on the block diagram, facilitating direct reference from a function on the block diagram to the comparable function on the logic diagrams, wiring lists, signal indexes, and signal transfers from board to board.

The clock generator for the floating point processor unit operates at a rate of 5 MHz (200-nanosecond period) supplying a 100-nanosecond clock signal, and its complement, to the FPP logic. The clock generator is located on the FPP D-register card. The complemented clock signal ($\overline{\text{Clock}}$) allows the clock cycle to be split, so that an active bit may be loaded into a register in the first 50 nanoseconds and perform its function in the second 50 nanoseconds. This high-speed feature employs a master/slave pair of flip-flops for each bit.

TABLE 9. ROM INSTRUCTIONS

| INSTR | DEFINITION |
|---|---|
| TM1 | Transfer Mode 1<br>C (shifted) to A-adder<br>C (shifted) to B-adder<br>A (shifted) to C-adder |
| TM2 | Transfer Mode 2<br>B (shifted) to A-adder<br>A (shifted) to B-adder<br>D (shifted) to C-adder |
| TM3 | Transfer Mode 3<br>D (shifted) to A-adder<br>A to B-adder<br>B to C-adder |
| TM4 | Transfer Mode 4<br>A (shifted to A-adder<br>B (shifted to B-adder<br>IM* or IL** to C-adder<br>* if Y0 = 0<br>** if Y0 = 1 |
| CP ( ) | Complement TM input to A, B, or C-adder |
| CI ( ) | Inject Carry into A-, B-, or C-adder |
| PCY | Propagate Carry into all adders |
| IRT | Inhibit Result of Test (for diagnostic use only) |
| BI8 | Inject eighth bit (7) into all adders |
| ERR | Error line to computer I/O |
| FLG | Flag to computer I/O |
| RR( ) | Read Register A, B, or C into corresponding adder (add to TM input) |
| RYY | Read Byte Y of all registers (Y = 0 to 5, determined by byte counter) |
| SYY | Store into Byte Y, designated by byte counter |
| SY5 | Store into Byte 5 |
| SYE | Store into Exponent Byte |
| SYS | Store into Shift Control Byte |
| CON | Load Constant into D-register from ROM location specified by S byte of C-register (plus AD9 bit to specify upper or lower half of ROM); then execute remainder of instruction word. (If CIC is used in the same word as CON, CIC will be inhibited if CS is 24 or higher.) |
| BRN | Branch conditionally to line (on current page) specified by BL field (test true) or by BP field (test false); or unconditionally to page and line specified by BP and BL fields (test specification = TRU). |
| IND | Indirect address. Use 8 bits of C-register S byte for next ROM address. (Lower half of ROM, since 9th bit = 0.) |
| JSB | Jump to Subroutine at address specified by BP and BL fields; TS field saved for return line address; return is always to same page on which JSB was given. One nested JSB level permitted. |
| RTN | Return to address saved by most recent JSB instruction; see JSB above. |
| YM1 | Y Minus One (decrement byte counter) |
| YF0 | Y Forced to Zero (clear byte counter) |
| YP1 | Y Plus One (Increment Byte Counter) |
| T ( ) | Test specification. Coded to specify 1 of 16 conditions for true/false test. (Refer to table 4-4). Used with branch instruction BRN. |

TABLE 9. ROM INSTRUCTIONS—Continued

| INSTR | DEFINITION |
|---|---|
| RY5 | Read Byte 5 of all registers |
| RYE | Read Exponent Byte of all registers |
| RYS | Read Shift Control Byte of all registers |
| SR ( ) | Store into Register A, B, or C from output of corresponding adder |

| INSTR | DEFINITION |
|---|---|
| AD9 | Ninth Address bit; used only for ROM dump routine to read low half in conjunction with CCN. |
| BP | Branch Page address field; 5 bits. |
| BL | Branch Line address field; 4 bits. |

TABLE 10. TEST INPUT

| TEST | T03 | T02 | T01 | T00 | TEST INPUT | |
|---|---|---|---|---|---|---|
| T0 | 0 | 0 | 0 | 0 | NONE | Forced TRU output for unconditional branching |
| T1 | 0 | 0 | 0 | 1 | AB8 | Eighth bit (7) of A-Adder output |
| T2 | 0 | 0 | 1 | 0 | $\overline{AOO}$ | A-Adder Output non-zero |
| T3 | 0 | 0 | 1 | 1 | COM | AB8 Compares with (equals) BB8 |
| T4 | 0 | 1 | 0 | 0 | BB8 | Eighth bit (7) of B-Adder output |
| T5 | 0 | 1 | 0 | 1 | $\overline{BOO}$ | B-Adder output non-zero |
| T6 | 0 | 1 | 1 | 0 | CB8 | Eighth bit (7) of C-Adder output |
| T7 | 0 | 1 | 1 | 1 | S80 | A-shifter output = 80 or higher (decimal) |
| T8 | 1 | 0 | 0 | 0 | S40 | A-shifter output = 40 or higher (decimal) |
| T9 | 1 | 0 | 0 | 1 | Y00 | Byte counter = 0 |
| T10 | 1 | 0 | 1 | 0 | Y01 | Byte counter = 1 |
| T11 | 1 | 0 | 1 | 1 | Y04 | Byte counter = 4 (decimal) |
| T12 | 1 | 1 | 0 | 0 | Y05 | Byte counter = 5 (decimal) |
| T13 | 1 | 1 | 0 | 1 | S83 | S80 is true (see above), or CS is 3 (module 4) |
| T14 | 1 | 1 | 1 | 0 | OPC | Opcode signal present from computer I/O |
| T15 | 1 | 1 | 1 | 1 | ENC | Encode signal present from computer I/O |

As shown in FIG. 72, Clock latches the master flip-flop, and $\overline{Clock}$ latches the slave flip-flop. The master flip-flop is loaded with data (Data 1) by Clock, and about 45 nanoseconds later $\overline{Clock}$ transfers the bit to the slave flip-flop. (There is a slight offset between Clock and $\overline{Clock}$.) The output of the slave flip-flop can then be used in the logic without affecting the inputs (such as Data 2) that determined the setting of the master flip-flop.

The FPP ROM has nine input lines (RA0 through RA8), thus allowing 512 addresses ($2^9$), and 48 output lines (R0 through R47), giving a word length of 48 bits.

The ROM constantly reads out whatever contents are enabled by the nine address lines. The ROM output lines are clocked either into the instruction register or if the preceding instruction contained a constant call into the D-register . All words in the ROM are either instructions or constants. For start-up purposes (power-on), ROM is forced to start at address 0.

The FPP ROM contents are listed in Table 11 in the form of mnemonics and constants. Physically, the ROM is contained in 24 microcircuit packages on a single printed circuit card. Each package accepts 8 of the 9 address lines and has 4 of the 48 output lines. (See logic diagram, FIGS. 11A–E). The ninth address bit (RA8) selects either the high half or low half of ROM. The lower rank of 12 packages is enabled when RA8 is 0 and is therefore active for address 0 through 255 (decimal). The upper rank of 12 packages is enabled when RA8 is 1 and is therefore active for address 256 through 511. The output of the two ranks are "or"-tied together.

TABLE 11

```
                          D I S - A S S E M B L Y                    DATE :  6/16/1970
PAGE.LINE LABEL           MNEMONICS/DATA                          QUAL(RTN)   ALT  DIR
  0.  0      5  BRN SYY RYY                                       TRU( 0)      10   13
  0.  1    400  BRN SYY RY5 RRA                                   TRU( 0)      14    0
  0.  2    410  YP1 BRN SYY RYY CIB CPB                           TRU( 0)      31    3
  0.  3    415  JSB SYY RYY                                          ( 4)      19   12
  0.  4    420  JSB SYE SRA RYE TM1                                  ( 1)       0    6
  0.  5   4015  BRN SYY RYY                                       TRU( 0)       5    0
  0.  6    425  YP1 BRN SYY SRC RYY TM3                           Y05(12)       6    2
  0.  7   1300  YM1 BRN SYY RY5 RRA                                TRU( 0)       6    0
  0.  8    430  BRN SYY RYY                                       TRU( 0)      14   12
  0.  9    405  JSB SYY RYY                                          ( 8)       0    6
  0. 10    435  BRN SYY RYY                                       BB8( 4)       8    9
  0. 11   2400  YM1 BRN SYY RYY                                   TRU( 0)      12    0
  0. 12    620  BRN SYS SRC SRB SRA RYS RRC CIC CIB CPA           TRU( 0)      19   15
                TM1
```

```
0.13   4005  JSB SYE SRA RYE RRB                          (14)  15   0
0.14   4010  YM1 BRN SY5 SRC RYE RRC                TRU( 0)  13   2
0.15   4000  JSB SYE SRA RY5 RRB                        (14)  15   7
1. 0    605  YP1 BRN SYY SRB RYY RRB PCY           Y04(11)   0   2
1. 1   2100  JSB SYS SRC RYE CPC TM3                    ( 4)  10   0
1. 2    610  BRN SYY SRB RYY RRB PCY               TRU( 0)  30  13
1. 3   2105  JSB SYS SRC RYE CPC TM3                    ( 6)  10   0
1. 4   2115  JSB SYE SRA RYE TM1                        ( 6)  30  11
1. 5   2110  JSB SYS SRC RYE CPC TM3                    ( 8)  10   0
1. 6   2120  BRN SYY RYY                             TRU( 0)  14  12
1. 7   2130  JSB SYS SRA RY5 RRB CIA                    ( 4)  15  13
1. 8   2125  JSB SYS SRB RYY                            (12)   8  15
1. 9   2160  JSB SYE SRC RYE CPC TM3                    ( 4)   3   0
1.10   2135  BRN SYE SRB RYS TM1                   TRU( 0)  23   4
1.11   2165  JSB SYE SRC RYE CPC TM3                   ( 6)   3   0
1.12   2145  JSB SYE SRA RYE TM1                       (14)  30  11
1.13   2170  YP1 JSB SYE SRC RYE CPC TM3               (12)   3   0
1.14   2150  BRN SYY RYY                           TRU( 0)   0   7
1.15   2175  JSB SYE SRC RYE CPC TM3                   (10)   3   0
2. 0   4400  YM1 JSB SYY RYY                           ( 4)   5  11
2. 1    215  YM1 BRN SYS SRC RYE TM3               TRU( 0)   4   0
2. 2   4410  YF0 BRN SYS SRC SRB SRA RYS RRC RRB BI8 S40( 8)   6   2
              CIB CIA CPB CPA
2. 3    205  BRN SYY RY5 RRB                       TRU( 0)   0  10
2. 4   4405  JSB SYS SRB SRA RYY TM1                    ( 2)  23  13
2. 5    230  YP1 BRN SYE SRC RY5 TM3               TRU( 0)   3  15
2. 6   4415  BRN SYY SRC RY5 RRB                    TRU( 0)  26   0
2. 7    225  YM1 BRN SYS SRC RYE TM3               TRU( 0)   4   6
2. 8   4425  YP1 BRN SYY SRB SRA RYY RRB RRA PCY CPB Y05(12)   8   2
              TM2
2. 9    200  BRN SYE SRB RYE TM1                   TRU( 0)  10  15
2.10   4435  YP1 BRN SYY SRB RYY TM4                Y05(12)  10   0
2.11    235  YM1 BRN SYY SRC RYY                   TRU( 0)  29  11
2.12   4420  BRN SYY RYY                           TRU( 0)  12   9
2.13    245  YM1 BRN SYY RYY                       TRU( 0)   2  14
2.14    250  YM1 BRN SYY RYY TM1                   TRU( 0)  22  13
2.15    255  BRN SYS SRC RYY                       TRU( 0)   8  12
3. 0   3100  BRN SYY RY5 RRB                       CB8( 6)   1   2
3. 1   3105  YF0 BRN SYS SRC SRB RYY CIC CIB        TRU( 0)   3  14
3. 2   3110  YP1 BRN SYS SRC RYE RRC RRB            BN0( 5)   3   5
3. 3   3115  BRN SYE SRB RYE RRB CIB                TRU( 0)  10   5
3. 4   3120  JSB SYS SRB SRA RYE CIB CIA TM1            ( 6)  23  13
3. 5   3125  YM1 BRN SYS SRC RYS RRC TM3            BN0( 5)   1   7
3. 6   3130  YF0 BRN SYS SRC SRB SRA RYS RRC RRB BI8 S40( 8)   8   3
              CIB CIA CPB CPA
3. 7   3135  YF0 BRN SYE SRC RYS RRC CPC TM3       Y01(10)  10  11
3. 8   3140  JSB SYE SRC SRB SRA RYE RRB CIB CIA        ( 9)  22  11
3. 9   3145  BRN SYY RYY                           TRU( 0)  30  11
3.10   3150  YM1 BRN SYY RYS RRC                   TRU( 0)  11   0
3.11   3155  BRN SYE SRC RYE RRC TM3               CB8( 6)  13  12
3.12   3160  YM1 BRN SYE SRB RYE TM1               TRU( 0)   1  13
3.13   3165  BRN SYE SRC RYE CPC TM3               TRU( 0)   3   0
3.14   3175  YP1 BRN SYY SRB RYY TM4               Y05(12)  14   4
3.15   2900  YP1 BRN SYS SRA RYE TM1               TRU( 0)   4  14
4. 0   2600  YF0 BRN SYS SRB SRA RYS RRB RRA CPB CPA CB8( 6)   1   4
              TM1
4. 1   2605  BRN SYS SRC RYS TM3                   S40( 8)   2   5
4. 2   2620  YP1 BRN SYY SRB RYY TM4               Y05(12)   2   3
4. 3   2625  BRN SYY SRB SRA RYS RRB CIB CPB TM1   TRU( 0)   7   2
4. 4   2610  YF0 JSB SYY RYY                            ( 5)  18   9
4. 5   2615  BRN SYY SRB RYY                       TRU( 0)  10  13
4. 6   2630  YM1 BRN SYY SRA RYY TM2               CB8( 6)   7   4
4. 7   2635  YF0 BRN SYS SRB RYS RRB CPB TM1       TRU( 0)   4   8
4. 8   2640  YP1 BRN SYE SRB RYS RRB TM1           BB8( 4)   9  13
4. 9   2645  BRN SYY SRB RYY TM4                   TRU( 0)   4  10
4.10   2665  YP1 BRN SYY RYY RRB BI8               TRU( 0)   4  11
4.11   2650  YP1 BRN SYY SRB RYY PCY TM4           Y05(12)  11  12
4.12   2655  YP1 BRN SYY SRB RYY                   COM( 3)  13   5
4.13   2660  BRN SYS SRC RYY CIC                   TRU( 0)   8  12
```

| | | | | | |
|---|---|---|---|---|---|
| 4.14 | 2670 | YP1 JSB SYY SRB RYY RRB BI8 | (15) | 1 | 0 |
| 4.15 | 2675 | YP1 BRN SYY SRB RYY | TRU( 0) | 4 | 12 |
| 5. 0 | 1000 | JSB SYS SRB RYY CIB | ( 1) | 5 | 15 |
| 5. 1 | 1005 | YP1 BRN SYY SRC SRA RYY TM1 | Y05(12) | 1 | 2 |
| 5. 2 | 1010 | YP1 BRN SYY SRB SRA RYY TM2 | Y05(12) | 2 | 12 |
| 5. 3 | 1015 | BRN SYS SRA RYS RRA CIC CIA CPC | CB8( 6) | 9 | 5 |
| 5. 4 | 1020 | YP1 BRN SYY SRA RYY TM3 | Y05(12) | 4 | 15 |
| 5. 5 | 1030 | YP1 BRN SYY SRC SRB RYY RRC RRB CIB CPB TM2 | TRU( 0) | 5 | 6 |
| 5. 6 | 1035 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB TM2 | Y04(11) | 6 | 8 |
| 5. 7 | 1070 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC TM2 | S80( 7) | 3 | 10 |
| 5. 8 | 1065 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB TM2 | S80( 7) | 3 | 10 |
| 5. 9 | 1025 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC TM2 | Y04(11) | 9 | 7 |
| 5.10 | 1040 | YF0 BRN SYE SRA RYE TM1 | TRU( 0) | 7 | 0 |
| 5.11 | 1045 | BRN SYS SRC RYY | TRU( 0) | 13 | 11 |
| 5.12 | 1050 | BRN SYS SRA RY5 RRC CPA | TRU( 0) | 5 | 3 |
| 5.13 | 1055 | YP1 BRN SYY SRB RYY TM4 | Y05(12) | 13 | 4 |
| 5.14 | 1060 | YF0 RTN CON SYY RY5 RRC RRB RRA | ( 0) | 0 | 0 |
| 5.15 | 1075 | YP1 BRN SYY SRC RYY | Y05(12) | 15 | 11 |
| 6. 0 | 1205 | YF0 BRN SYS SRB RYS RRB BI8 | AN0( 2) | 1 | 4 |
| 6. 1 | 1210 | YM1 BRN SYS SRC RYY CPC | TRU( 0) | 8 | 12 |
| 6. 2 | 1220 | JSB SYS SRA RYY CPA | ( 5) | 5 | 14 |
| 6. 3 | 1215 | BRN SYS SRC RY5 RRB RRA | S83(13) | 10 | 5 |
| 6. 4 | 1235 | JSB SYE SRA RYE CPA TM1 | ( 2) | 5 | 15 |
| 6. 5 | 1240 | BRN SYS SRA RYS RRA CIC CIA CPC | COM( 3) | 8 | 7 |
| 6. 6 | 1245 | YF0 BRN SYE SRC RYE RRA CIC | TRU( 0) | 7 | 7 |
| 6. 7 | 1255 | YP1 BRN SYY SRC SRB RYY RRC RRB CIB CPB TM2 | TRU( 0) | 6 | 9 |
| 6. 8 | 1250 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC TM2 | Y04(11) | 8 | 11 |
| 6. 9 | 1260 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB TM2 | Y04(11) | 9 | 12 |
| 6.10 | 1265 | YP1 BRN SYY SRB RYY TM1 | Y05(12) | 10 | 6 |
| 6.11 | 1270 | YP1 BRN SYY SRC SRB RYY RRC RRB RRA PCY CPC TM2 | S80( 7) | 5 | 3 |
| 6.12 | 1275 | YP1 BRN SYY SRC SRB RYY RRC RRB RRA PCY CPB TM2 | S80( 7) | 5 | 3 |
| 6.13 | 5825 | JSB SY5 SRC RYS RRC | (14) | 8 | 15 |
| 6.14 | 5830 | JSB SYS SRC RY5 RRC | (15) | 19 | 12 |
| 6.15 | 5835 | BRN SYE SRA RYE TM1 | TRU( 0) | 9 | 6 |
| 7. 0 | 1100 | BRN SYY SRA RYE RRB RRA | S40( 8) | 1 | 2 |
| 7. 1 | 1105 | JSB SYS SRC RYE RRC | ( 5) | 31 | 2 |
| 7. 2 | 1110 | BRN SYS SRB SRA RYY | COM( 3) | 3 | 4 |
| 7. 3 | 1115 | BRN SYE SRB RYE RRB TM2 | TRU( 0) | 14 | 12 |
| 7. 4 | 1120 | JSB SYE SRC RYY TM1 | ( 0) | 8 | 15 |
| 7. 5 | 1130 | BRN SYY RYY | TRU( 0) | 10 | 13 |
| 7. 6 | 1145 | YP1 BRN SYS SRC RYY PCY | TRU( 0) | 8 | 12 |
| 7. 7 | 1150 | BRN SYE SRA RYE RRA CIA TM1 | AB8( 1) | 8 | 0 |
| 7. 8 | 1155 | BRN SYE RYE RRC RRB CIC | AB8( 1) | 0 | 9 |
| 7. 9 | 1160 | BRN SYE SRB SRA RYE RRB RRA CIA CPA TM1 | BB8( 4) | 10 | 0 |
| 7.10 | 1165 | BRN SYY RYY CIC CPC | BB8( 4) | 0 | 6 |
| 7.11 | 1170 | BRN SYS SRC RYY TM2 | TRU( 0) | 5 | 14 |
| 7.12 | 4710 | YF0 BRN CON SYE SRC RYY CIC TM1 | CB8( 6) | 14 | 13 |
| 7.13 | 4715 | JSB SYS SRB SRA RYY CIB | (15) | 5 | 13 |
| 7.14 | 4720 | JSB SYS SRB RYY CIB | (15) | 5 | 13 |
| 7.15 | 4730 | BRN SYS SRC SRB SRA RYY CIC CIB CPB | TRU( 0) | 2 | 8 |
| 8. 0 | 2000 | YP1 BRN SYE SRA RYS CIA TM1 | CB8( 6) | 4 | 1 |
| 8. 1 | 2005 | YP1 BRN SYS SRC RYY | TRU( 0) | 8 | 12 |
| 8. 2 | 2010 | YP1 BRN SYY SRC SRA RYY TM3 | Y05(12) | 2 | 3 |
| 8. 3 | 2015 | YP1 BRN SYY SRB RYY | Y05(12) | 3 | 5 |
| 8. 4 | 2020 | JSB SYE SRC RYY CPC | ( 6) | 5 | 11 |
| 8. 5 | 2025 | BRN SYE SRA RYE RRB CPB TM1 | TRU( 0) | 30 | 0 |
| 8. 6 | 2030 | JSB SYS SRB RYE RRB | ( 7) | 5 | 4 |
| 8. 7 | 2035 | JSB SYS SRC SRA RY5 RRB CIC CPA | ( 8) | 19 | 0 |
| 8. 8 | 2040 | YP1 JSB SYE SRC RYE RRC TM3 | ( 9) | 5 | 11 |

```
 8. 9   2045  BRN SYS SRA RYE RRA CIA                          TRU( 0)   10   7
 8.10   2050  BRN SYS SRC SRB SRA RYS RRC RRB RRA BI8          CB8( 6)    2  11
               CIA CPB CPA
 8.11   2055  YP1 BRN SYY SRC SRA RYY PCY CPA TM3              Y05(12)   11   3
 8.12     15  YM1 BRN SYS SRC RYS RRC CIC                     Y01(10)   12  13
 8.13     45  JSB SYY SRA RYS TM1                                (14)   13  12
 8.14     55  BRN SYY RYY                                     TRU( 0)   17   9
 8.15    600  YM1 BRN SYS SRC SRA RY5 RRB CIC                 TRU( 0)   18   1
 9. 0   5700  YP1 BRN SYS SRA RYS TM1                         CB8( 6)    1   4
 9. 1   5705  YF0 BRN SYE SRA RYE TM1                         S40( 8)    2   3
 9. 2   5710  BRN SYS SRB SRA RYY                             S40( 8)    6   5
 9. 3   5715  BRN SYY RYY                                     S40( 8)    7   4
 9. 4   5720  YF0 BRN SYS SRC RYY                             TRU( 0)    8  12
 9. 5   5725  YP1 BRN SYY SRA RYY TM1                         Y05(12)    5   8
 9. 6   5730  YM1 JSB SYY RY5 RRB                                ( 9)   12  13
 9. 7   5735  IND SYS SRC SRB SRA RYY                            ( 0)    0   0
 9. 8   5740  YM1 JSB SYY RY5 RRA                                (10)   12  13
 9. 9   5745  BRN SYY SRB SRA RYY                             BB8( 4)    7  12
 9.10   5750  BRN SYS SRB SRA RYY                            AB8( 1)    6  11
 9.11   5755  JSB SYY SRA RYY                                   (13)   19  12
 9.12   5760  BRN SYS SRB RYY                                TRU( 0)    9  14
 9.13   5765  BRN SYE SRA RYE TM1                            TRU( 0)    6  13
 9.14   5815  JSB SY5 SRC RYS RRC                               (15)    8  15
 9.15   5820  BRN SYS SRC RY5 RRC                            TRU( 0)    9   7
10. 0   2200  BRN SYY RY5 RRB                                CB8( 6)    2   1
10. 1   2205  YM1 BRN SYY RYS RRC CIC                         BN0( 5)    5   8
10. 2   2210  YP1 JSB SYE SRC RYE CPC TM3                        ( 6)    5  11
10. 3   2215  YF0 BRN SYS SRB SRA RYS RRB CIA                 S40( 8)    4   5
10. 4   2220  BRN SYE SRC SRB RYE RRB CIC CIB                 TRU( 0)    8  10
10. 5   2225  YF0 JSB SYS SRA RYY                                (10)    5  11
10. 6   2230  JSB SYS SRB SRA RYE CIB CIA TM1                    ( 3)   23  13
10. 7   2240  BRN SYS SRA RYE TM1                             TRU( 0)    8  10
10. 8   2260  YF0 BRN SYS SRC RYS RRC TM3                     CB8( 6)    2   9
10. 9   2265  YP1 BRN SYY RYS RRC CPC                         TRU( 0)    8   0
10.10   2270  BRN SYE SRC SRA RYY CIA                        TRU( 0)   30  12
10.11     65  BRN SYY RYY                                    TRU( 0)   17  10
10.12    100  BRN SYY SRA RYY TM2                            ENC(15)   11  12
10.13    105  JSB SYY RYY                                       (12)   13  12
10.14    150  BRN SYY SRC RYY FLG TM4                        Y00( 9)   12  13
10.15    210  YP1 BRN SYY SRB RYY TM1                        Y05(12)   15  13
11. 0   3200  YM1 BRN SYE SRC RYS RRC TM3                    CB8( 6)    2   1
11. 1   3205  YF0 BRN SYS SRC RY5 CPC TM3                    TRU( 0)   27  15
11. 2   3210  JSB SYE SRA RYS RRA CIA TM1                        ( 3)    5  11
11. 3   3215  JSB SYS SRB RYE RRB CIB                            ( 4)    5   4
11. 4   3220  JSB SYS SRC SRA RY5 RRB CPC CPA                    ( 5)   19   0
11. 5   3225  YM1 JSB SYE SRC RYE RRC CPC                        ( 6)    5  11
11. 6   3230  BRN SYS SRB SRA RYE CPB TM1                     TRU( 0)   11   7
11. 7   3235  BRN SYE SRB RY5 RRA TM2                         TRU( 0)   11   8
11. 8   3240  BRN SYE SRC RYE TM3                             COM( 3)   13   9
11. 9   3245  BRN SYE SRA RYE TM1                             CB8( 6)   10  11
11.10   3250  JSB SYS SRC SRB SRA RYS RRC BI8 CIB CIA           (12)   22  11
               CPB CPA
11.11   3255  BRN SYE SRB SRA RYE CIB CIA CPB CPA TM1         TRU( 0)   11  10
11.12   3260  YP1 BRN SYY SRB SRA RYY RRB RRA PCY CPA         Y05(12)   12  15
               TM2
11.13   3265  BRN SYS SRC RYE TM3                            TRU( 0)   27  15
11.14   3270  BRN SYS SRA RYE RRA                            TRU( 0)   23   0
11.15   3275  YP1 BRN SYY SRC RYY TM3                        Y05(12)   15  14
12. 0   2405  YF0 JSB SYE SRA RYY TM2                            ( 1)    5  11
12. 1   2410  JSB SYS SRB RYY CIB                               ( 2)    5  13
12. 2   2415  YP1 JSB SYY RYY                                   ( 3)    5  11
12. 3   2420  BRN SYE SRC RYE TM3                             BN0( 5)    9   4
12. 4   2425  BRN SYS SRA RYE CIA TM1                         CB8( 6)    5   6
12. 5   2430  BRN SYE SRB RYY                                S80( 7)   12  11
12. 6   2435  JSB SYS SRB SRA RYE CIB CIA CPB CPA TM1            ( 7)   23  13
12. 7   2440  YF0 BRN SYS SRC SRB RYS RRC RRB BI8 CPB         S40( 8)   10   9
12. 8   2445  BRN SYS SRC SRB RYS RRC BI8 CPB                 TRU( 0)   12  10
12. 9   2450  BRN SYE SRB RYE RRB CIB                         TRU( 0)   14  12
12.10   2455  BRN SYS SRA RY5 RRB CPA                        TRU( 0)   27   0
```

```
12.11   2460  BRN SY5 SRA RYY                                  TRU( 0)   12   8
12.12   2465  YP1 BRN SYY SRB SRA RYY TM4                      Y05(12)   12   8
12.13   5800  BRN SYS SRB SRA RYS RRB RRA CPB CPA              COM( 3)   14  15
12.14   5805  YF0 RTN SYY RYY TM4                                ( 0)    0   0
12.15   5810  YF0 RTN SYY RYY CPB CPA TM4                        ( 0)    0   0
13. 0   4600  BRN SYY RY5 TM1                                 CB8( 6)    1   4
13. 1   4605  BRN SYS SRA RYS RRA CIC CIA CPC                  BN0( 5)    3   5
13. 2   4610  YM1 JSB SYS SRB RYS RRB CPB TM1                     ( 7)    5  11
13. 3   4615  BRN SYS SRA RYY CPA                              TRU( 0)    5  10
13. 4   4620  BRN SYS SRA RYS RRA CIB CIA CPB                  TRU( 0)   13   6
13. 5   4625  YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC          Y05(12)    5   0
              TM2
13. 6   4630  YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB          Y05(12)    6   0
              TM2
13. 7   4635  YP1 BRN SYY SRA RYY TM3                          Y05(12)    7   8
13. 8   4640  YP1 BRN SYY SRB RYY TM4                          Y05(12)    8   9
13. 9   4645  JSB SYE SRC RYS CIC TM3                             (10)    5  11
13.10   4650  JSB SYS SRC RYS RRC CIC                            ( 0)    5  14
13.11   1230  BRN CON SYY RYY                                 TRU( 0)    7  11
13.12     10  YF0 BRN SYS SRB SRA RYY BI8 CIB CIA CPB         OPC(14)   13  12
13.13     20  BRN SYY SRB RYE RRB                             TRU( 0)   25   9
13.14   8910  BRN SYY RYY                                     TRU( 0)   31   6
13.15   8915  YP1 BRN SYY SRC SRB SRA RYY                     Y05(12)   15  14
14. 0    300  BRN SYS SRA RY5 RRB                             AN0( 2)   14   1
14. 1    305  BRN SYY SRB SRA RYE RRB RRA                     BN0( 5)    7   2
14. 2    310  BRN SYY SRB RYY RRB CIB CPB TM2                 COM( 3)    4   3
14. 3    315  BRN SYS SRA RYY TM2                             BB8( 4)    5   4
14. 4    320  BRN SYS SRA RYY                                 AB8( 1)    3   1
14. 5    325  BRN SYY RY5 RRB RRA                             S40( 8)    6  14
14. 6    330  BRN SYY SRB SRA RYY                             COM( 3)    9  10
14. 7    335  YP1 BRN SYY SRB SRA RYY TM2                     Y05(12)    7   8
14. 8    340  BRN SYE SRB SRA RYY RRB RRA                     AN0( 2)   14   2
14. 9    345  YP1 BRN SYY SRB RYY RRB PCY TM2                 Y05(12)    9  12
14.10    350  YP1 BRN SYY SRB RYY RRB PCY TM2                 Y05(12)   10  11
14.11    355  BRN SYY RYY                                     COM( 3)   13  12
14.12    360  YF0 JSB SYS SRB SRA RYY                            (14)    8  15
14.13    365  YF0 JSB SYS SRC RYY                                (12)   31   1
14.14    370  BRN SYY RYY                                     TRU( 0)   10  13
14.15    555  YF0 BRN SYS SRA RYY TM2                         TRU( 0)   15  15
15. 0   4100  BRN SYY RYE TM3                                 BN0( 5)    2   1
15. 1   4105  BRN SYY RYY                                     CB8( 6)   12   3
15. 2   4110  JSB SY5 SRB RY5 RRB BI8                            ( 6)   26  12
15. 3   4115  JSB SYE SRC RYY                                    ( 4)    5  11
15. 4   4120  JSB SYS SRB RYY CIB                                ( 5)    5  13
15. 5   4125  BRN SYY SRC RYE CIC CPC TM3                      TRU( 0)    2   0
15. 6   4130  JSB SYS SRB RYY                                    (11)    8  15
15. 7   4135  BRN SYY RY5 RRB                                 BB8( 4)    8  12
15. 8   4140  BRN SYY RYY                                     BN0( 5)   12   9
15. 9   4145  JSB SY5 SRB RY5 RRB BI8                             (10)   26  12
15.10   4150  BRN SYE SRC SRB RYE CIB TM3                     TRU( 0)    2  10
15.11   4160  BRN SYY RY5 RRB                                 TRU( 0)   21  14
15.12   4165  YM1 BRN SYS SRC RYY                             TRU( 0)    8  12
15.13   4170  BRN SYE SRC RYE TM3                             BB8( 4)   14  12
15.14   4175  YM1 BRN SYS SRC RY5 RRB BI8 CIB CPB             TRU( 0)   18  11
15.15    570  YP1 BRN SYY SRB RYY RRB TM1                     TRU( 0)    1   0
16. 0   9015  0.0000001 10101111 10000000 11000000 01000000 00010111
16. 1   8001  0.1001111 10101011 00000000 00011100 11110010 01001110
16. 2   8002  0.1000100 11111110 10010110 10100111 01110101 00011001
16. 3   8003  0.1000000 10101110 10101101 00111100 11111101 11000111
16. 4   8004  0.1000000 00101100 11001101 01111111 11100000 10100001
16. 5   8005  0.1000000 00001100 10101111 00001111 10000001 10110001
16. 6   8006  0.1000000 00000100 10101100 11111001 01110110 01011101
16. 7   8007  0.1000000 00000000 10101100 10101110 10100110 11000101
16. 8   8008  0.1000000 00000000 00101100 10101100 11001101 01100101
16. 9   8009  0.1000000 00000000 00001100 10101100 10101111 00001111
16.10   8010  0.1000000 00000000 00000100 10101100 10101100 11111001
16.11   8011  0.1000000 00000000 00000000 10101100 10101100 10101111
16.12   8012  0.1000000 00000000 00000000 00101100 10101100 10101101
16.13   8013  0.1000000 00000000 00000000 00001100 10101100 10101101
```

```
16.14   8014  0.1000000 00000000 00000000 00000100 10101100 10101101
16.15   8015  0.1000000 00000000 00000000 00000000 10101100 10101101
17. 0   8016  0.1000000 00000000 00000000 00000000 00101100 10101101
17. 1   8017  0.1000000 00000000 00000000 00000000 00001100 10101101
17. 2   8018  0.1000000 00000000 00000000 00000000 00000100 10101101
17. 3   8019  0.1000000 00000000 00000000 00000000 00000000 10101101
17. 4   8020  0.1000000 00000000 00000000 00000000 00000000 00101101
17. 5   8021  0.1000000 00000000 00000000 00000000 00000000 00001101
17. 6   8022  0.1000000 00000000 00000000 00000000 00000000 00000001
17. 7   8023  0.1000000 00000000 00000000 00000000 00000000 00000000
17. 8   9020  1.0000000 00000000 00000000 00000000 00000000 00000000
17. 9     35  YF0 BRN SYY RYY FLG ERR                          OPC(14)   14  12
17.10    120  BRN SYY RYY FLG                                  ENC(15)   12  11
17.11    130  YM1 BRN SYY SRA RYY FLG                          TRU( 0)   25  15
17.12    160  BRN SYY RYY FLG                                  OPC(14)   10  13
17.13    180  YF0 BRN SYS SRC RYY FLG TM4                      TRU( 0)    9   0
17.14     25  BRN SYY RYY FLG ERR                              ENC(15)    9  11
17.15   3425  YF0 BRN SYS SRB SRA RYE RRB RRA CIB CPB TM1       TRU( 0)   23   6
18. 0    505  YM1 BRN SYY RY5 RRB                              BN0( 5)    3   5
18. 1    510  BRN SYS SRC RYS RRC CIC CPC TM3                  BB8( 4)    2   3
18. 2    515  BRN SYY RYY TM4                                  Y00( 9)    4   9
18. 3    520  BRN SYY RYY CPB TM4                              Y00( 9)    0   9
18. 4    525  YM1 BRN SYY RY5 RRB                              BN0( 5)    2   5
18. 5    530  YP1 BRN SYS SRC SRA RYS RRC CIC CPA              BB8( 4)    6   7
18. 6    535  BRN SYS SRB RYS RRB CPB CPA TM2                  AB8( 1)    8   6
18. 7    540  BRN SYS SRB RYS RRB CPB TM2                      AB8( 1)    8   7
18. 8    545  YF0 BRN CON SYS SRC RYS RRC TM3                  TRU( 0)    0  12
18. 9    560  YP1 BRN SYY SRB RYY                              Y05(12)    9  10
18.10    565  BRN SYE SRC SRB RYY                              TRU( 0)   27  14
18.11   4700  YM1 BRN SY5 SRA RYE TM1                          BN0( 5)   13  12
18.12   4705  YP1 BRN SYY RYY TM1                              TRU( 0)    7  12
18.13   4725  YF0 BRN SYY RYY                                  TRU( 0)   10  13
18.14   5530  YP1 BRN SYY SRB RYY TM4                          Y05(12)   14  15
18.15   5535  BRN SY5 SRB RY5 RRB BI8                          TRU( 0)   31   2
19. 0   2300  BRN SYS SRB SRA RYS RRB RRA CIA CPB              BB8( 4)    3   1
19. 1   2305  BRN SYY RYS RRC CIC CPC                          BB8( 4)    2   5
19. 2   2310  YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC TM2       Y05(12)    2   0
19. 3   2315  BRN SYY RYS RRC CIB CPB                          BB8( 4)    4   5
19. 4   2320  YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB TM2       Y05(12)    4   0
19. 5   2325  BRN SYY RY5 RRB                                  CB8( 6)    6   7
19. 6   2330  BRN SYS SRC SRA RYS RRA CPC CPA                   BB8( 4)    3   1
19. 7   2335  YP1 BRN SYY SRA RYY TM1                          Y05(12)    7   8
19. 8   2340  YM1 BRN SYS SRB RYE CIB TM3                      TRU( 0)    1  10
19. 9   2345  YF0 RTN SYS SRB SRA RYY CPB                        ( 0)    0   0
19.10   2355  YM1 BRN SYY SRB RYY TM4                          Y00( 9)   10   9
19.11   2360  BRN SYY SRC RYY                                  TRU( 0)    8  11
19.12   2365  YP1 BRN SYY SRB SRA RYY TM2                      Y05(12)   12  13
19.13   2370  RTN SYE SRC SRB RYE TM3                            ( 0)    0   0
19.14   2375  YP1 BRN SYY SRB SRA RYY RRB RRA PCY CPB TM2       Y05(12)   14  13
19.15    550  YM1 BRN SYY SRC RYY TM2                          TRU( 0)   14  15
20. 0   7000  0.1001101 10111010 01110110 11010100 00100001 10101111
20. 1   7001  0.1101101 11101100 10101101 10001011 00111011 11100001
20. 2   7002  0.1111010 11100110 00111011 11111111 01011010 10000111
20. 3   7003  0.1111110 10101110 10000111 01100100 01110000 11000001
20. 4   7004  0.1111111 10101010 11101001 00001011 11111111 11011000
20. 5   7005  0.1111111 11101010 10101110 10010011 00011110 01101100
20. 6   7006  0.1111111 11111010 10101010 11101001 00111011 01101000
20. 7   7007  0.1111111 11111110 10101010 10101110 10010011 11011101
20. 8   7008  0.1111111 11111111 10101010 10101010 11101001 00111110
20. 9   7009  0.1111111 11111111 11101010 10101010 10101110 10010100
20.10   7010  0.1111111 11111111 11111010 10101010 10101010 11101001
20.11   7011  0.1111111 11111111 11111110 10101010 10101010 10101111
20.12   7012  0.1111111 11111111 11111111 10101010 10101010 10101011
20.13   7013  0.1111111 11111111 11111111 11101010 10101010 10101011
```

```
20.14   7014  0.1111111 11111111 11111111 11111010 10101010 10101011
20.15   7015  0.1111111 11111111 11111111 11111110 10101010 10101011
21. 0   7016  0.1111111 11111111 11111111 11111111 10101010 10101011
21. 1   7017  0.1111111 11111111 11111111 11111111 11101010 10101011
21. 2   7018  0.1111111 11111111 11111111 11111111 11111010 10101011
21. 3   7019  0.1111111 11111111 11111111 11111111 11111110 10101011
21. 4   7020  0.1111111 11111111 11111111 11111111 11111111 11101011
21. 5   7021  0.1111111 11111111 11111111 11111111 11111111 11111011
21. 6   7022  0.1111111 11111111 11111111 11111111 11111111 11111111
21. 7   7023  0.1111111 11111111 11111111 11111111 11111111 11111111
21. 8   4200  YP1 BRN SYY SRB SRA RYY TM4                    Y05(12)    8   9
21. 9   4205  BRN SYS SRB SRA RYY CIA CPA                    AB8( 1)   11  10
21.10   4215  YP1 BRN SYY SRA RYY PCY CPA TM4                Y05(12)   10  12
21.11   4220  YP1 BRN SYY SRB SRA RYY RRB RRA PCY CPA        Y05(12)   11  13
              TM2
21.12   4225  BRN SYE SRC SRB RYE TM3                        TRU( 0)    7  15
21.13   4230  BRN SYE SRC SRB RYE CIC CPC TM3                TRU( 0)    2   2
21.14   4235  BRN SYS SRB SRA RYS CIB TM1                    BN0( 5)   15   8
21.15   4240  BRN SYY RYY                                    TRU( 0)   15  12
22. 0   3300  BRN CON SYS SRC SRB SRA RYS RRC RRB RRA        CB8( 6)    4   1
              CIC CIB CIA
22. 1   3305  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        TRU( 0)   22   2
              CIB CIA CPB CPA TM2
22. 2   3315  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        Y04(11)    2   3
              PCY CPB CPA TM2
22. 3   3325  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        S83(13)    0   7
              PCY CPB CPA TM2
22. 4   3310  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        TRU( 0)   22   5
              CIC CPC TM2
22. 5   3320  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        Y04(11)    5   6
              PCY CPC TM2
22. 6   3330  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA        S83(13)    0   7
              PCY CPC TM2
22. 7   3335  BRN SYS SRC RYS RRC BI8                        S40( 8)    8  12
22. 8   3340  BRN SYY RY5 RRC                                CB8( 6)    9   0
22. 9   3345  BRN SYY RYY                                    CB8( 6)    4   1
22.10   3350  YP1 BRN SYY SRB RYY RRC                        Y05(12)   10   0
22.11   3355  YP1 BRN SYY SRC SRA RYY TM3                    Y05(12)   11  10
22.12   3360  RTN SYS SRC SRB SRA RYE RRC CIA CPA              ( 0)    0   0
22.13   3365  BRN SYS SRC RYY RRC                            AN0( 2)   15  14
22.14   3370  YF0 BRN CON SYY RYY                            TRU( 0)   29   9
22.15   3375  AD9 YF0 BRN CON SYY RYY                        TRU( 0)   29   9
23. 0   3400  YM1 BRN SYS SRB RYS RRC CPB                    S40( 8)    1   2
23. 1   3405  BRN SYE SRB SRA RYE RRB RRA CIB CIA            TRU( 0)   17  15
23. 2   3410  YF0 BRN SYS SRA RYY CIA CPA                    CB8( 6)    3   5
23. 3   3415  YP1 BRN SYY SRA RYY                            Y05(12)    3   7
23. 4   3420  BRN SYS SRA RYE RRA CIA CPA TM1                BN0( 5)    8  10
23. 5   3430  YP1 BRN SYY SRB SRA RYY PCY CPA TM4             Y05(12)    5   7
23. 6   3435  BRN SYY RYY CIA CPA                            TRU( 0)   23   5
23. 7   3440  BRN SYS SRB SRA RYY CIB CPB                    TRU( 0)   19  14
23. 8   3445  BRN SYE SRB RYY CIB                            S80( 7)   12   9
23. 9   3450  BRN SYY RY5 RRB                                TRU( 0)   14  11
23.10   3455  YP1 BRN SYY SRB RYY TM1                        Y05(12)   10  11
23.11   3460  BRN SYY SRA RYY CIA                            TRU( 0)    7   2
23.12   3465  YP1 BRN SYY SRB RYY RRB PCY TM2                Y05(12)   12   9
23.13   2065  YP1 BRN SYS SRC SRB RYS RRC RRB TM3            S40( 8)   15  14
23.14   2060  RTN SYS SRA RYY CPA                              ( 0)    0   0
23.15   2070  YP1 RTN CON SYS SRB RYS RRB                      ( 0)    0   0
24. 0   8100  0.0110001 10010101 11111000 10010110 00001000 01110000
24. 1   8101  0.1000110 01001111 10101001 11101010 10110100 00001100
24. 2   8102  0.1000001 01100010 10111011 11101010 00000100 01010001
24. 3   8103  0.1000000 01010110 00100100 01110010 01111010 10111011
24. 4   8104  0.1000000 00010101 01100010 00101011 01001101 11010110
24. 5   8105  0.1000000 00000101 01010110 00100010 01000110 10111100
24. 6   8106  0.1000000 00000001 01010101 01100010 00100010 10110100
24. 7   8107  0.1000000 00000000 01010101 01010110 00100010 00100100
24. 8   8108  0.1000000 00000000 00010101 01010101 01100010 00100010
24. 9   8109  0.1000000 00000000 00000101 01010101 01010110 00100010
24.10   8110  0.1000000 00000000 00000001 01010101 01010101 01100010
```

```
24.11   8111  0.1000000 00000000 00000000 01010101 01010101 01010110
24.12   8112  0.1000000 00000000 00000000 00010101 01010101 01010101
24.13   8113  0.1000000 00000000 00000000 00000101 01010101 01010101
24.14   8114  0.1000000 00000000 00000000 00000001 01010101 01010101
24.15   8115  0.1000000 00000000 00000000 00000000 01010101 01010101
25. 0   8116  0.1000000 00000000 00000000 00000000 00010101 01010101
25. 1   8117  0.1000000 00000000 00000000 00000000 00000101 01010101
25. 2   8118  0.1000000 00000000 00000000 00000000 00000001 01010101
25. 3   8119  0.1000000 00000000 00000000 00000000 00000000 01010101
25. 4   8120  0.1000000 00000000 00000000 00000000 00000000 00010101
25. 5   8121  0.1000000 00000000 00000000 00000000 00000000 00000101
25. 6   8122  0.1000000 00000000 00000000 00000000 00000000 00000001
25. 7   8123  0.1000000 00000000 00000000 00000000 00000000 00000000
25. 8   8124  0.1000000 00000000 00000000 00000000 00000000 00000000
25. 9     30  BRN SYY SRB RYY RRB TM4                          TRU( 0)   25  10
25.10     40  BRN SYY SRB RYY RRB PCY                          TRU( 0)   25  11
25.11     50  YM1 BRN SY5 SRA RYY TM1                          TRU( 0)   25  12
25.12     70  YM1 BRN SYS SRB RYS RRB                          TRU( 0)   25  13
25.13     80  YP1 BRN SYE SRC RYY TM1                          TRU( 0)   25  14
25.14     90  YF0 RTN SYE SRC RYE RRC TM1                         ( 0)    0   0
25.15    140  YM1 BRN SYY SRC RYY FLG TM4                      TRU( 0)   10  14
26. 0   4300  BRN CON SYS SRC SRB SRA RYS RRC RRB RRA           BB8( 4)    2   1
              CIC CIB CIA
26. 1   4305  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           TRU( 0)   26   3
              CIC CPC TM2
26. 2   4310  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           TRU( 0)   26   4
              CIB CIA CPB CPA TM2
26. 3   4315  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           Y04(11)    3   5
              PCY CPC TM2
26. 4   4320  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           Y04(11)    4   6
              PCY CPB CPA TM2
26. 5   4325  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           S83(13)    0   7
              PCY CPC TM2
26. 6   4330  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           S83(13)    0   7
              PCY CPB CPA TM2
26. 7   4335  BRN SYS SRC RYS RRC BI8                           S40( 8)    8  10
26. 8   4340  BRN SYY RY5 RRB                                   CB8( 6)    9   0
26. 9   4345  BRN SYY RYY                                       BB8( 4)    2   1
26.10   4350  YP1 BRN SYY SRB RYY TM1                          Y05(12)   10  11
26.11   4355  RTN SYS SRC SRB SRA RYY BI8 CIC CIA CPB              ( 0)    0   0
              CPA
26.12   4360  BRN SYS SRB RYY CIB                              TRU( 0)   26  13
26.13   4365  YP1 BRN SYY SRA RYY TM2                          Y05(12)   13  14
26.14   4370  BRN SY5 SRA RY5 RRA BI8                          TRU( 0)    5  15
26.15   9035  0.1011000 10111001 00001011 11111011 11101000 11101000
27. 0   2500  BRN CON SYS SRC SRB SRA RYS RRC RRB RRA           BB8( 4)    2   1
              CIC CIB CIA
27. 1   2505  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           TRU( 0)   27   3
              CIC CIA CPC CPA TM2
27. 2   2510  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           TRU( 0)   27   4
              CIB CPB TM2
27. 3   2515  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           Y04(11)    3   5
              PCY CPC CPA TM2
27. 4   2520  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           Y04(11)    4   6
              PCY CPB TM2
27. 5   2525  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           S80( 7)    0   7
              PCY CPC CPA TM2
27. 6   2530  YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA           S80( 7)    0   7
              PCY CPB TM2
27. 7   2535  YP1 BRN SYY SRB RYY TM1                          Y05(12)    7   8
27. 8   2540  BRN SY5 SRA RYE RRA                              TRU( 0)   27   9
27. 9   2545  BRN SYY RY5 RRB RRA                              TRU( 0)   14  11
27.10   5100  BRN SYY RYS RRC                                  BB8( 4)   12  11
27.11   5105  BRN SYE SRB RYS RRB TM1                          CB8( 6)   14  13
27.12   5110  BRN SYE SRB RYS RRB TM1                          CB8( 6)   13  14
27.13   5115  BRN SYS SRC RYE TM3                              COM( 3)   15  14
27.14   5120  RTN SYS SRC SRB SRA RYY CIC                         ( 0)    0   0
27.15   5125  BRN SYY RYS RRC BI8                              TRU( 0)    7   6
28. 0   7100  0.1100100 10000111 11101101 01010001 00010000 10110100
```

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 28. 1 | 7101 | 0.1110110 10110001 10011100 00010101 10000110 11101101 | | | | | | | | |
| 28. 2 | 7102 | 0.1111101 01101101 11010111 11100100 10110010 00000011 | | | | | | | | |
| 28. 3 | 7103 | 0.1111111 01010110 11101010 01101010 10110000 10111110 | | | | | | | | |
| 28. 4 | 7104 | 0.1111111 11010101 01101110 11011100 10110011 11111000 | | | | | | | | |
| 28. 5 | 7105 | 0.1111111 11110101 01010110 11101110 10100101 11011001 | | | | | | | | |
| 28. 6 | 7106 | 0.1111111 11111101 01010101 01101110 11101101 11001010 | | | | | | | | |
| 28. 7 | 7107 | 0.1111111 11111111 01010101 01010110 11101110 11101010 | | | | | | | | |
| 28. 8 | 7108 | 0.1111111 11111111 11010101 01010101 01101110 11101111 | | | | | | | | |
| 28. 9 | 7109 | 0.1111111 11111110 11110101 01010101 01010110 11101111 | | | | | | | | |
| 28.10 | 7110 | 0.1111111 11111111 11111101 01010101 01010101 01101111 | | | | | | | | |
| 28.11 | 7111 | 0.1111111 11111111 11111111 01010101 01010101 01010111 | | | | | | | | |
| 28.12 | 7112 | 0.1111111 11111111 11111111 11010101 01010101 01010101 | | | | | | | | |
| 28.13 | 7113 | 0.1111111 11111111 11111111 11110101 01010101 01010101 | | | | | | | | |
| 28.14 | 7114 | 0.1111111 11111111 11111111 11111101 01010101 01010101 | | | | | | | | |
| 28.15 | 7115 | 0.1111111 11111111 11111111 11111111 01010101 01010101 | | | | | | | | |
| 29. 0 | 7116 | 0.1111111 11111111 11111111 11111111 11010101 01010101 | | | | | | | | |
| 29. 1 | 7117 | 0.1111111 11111111 11111111 11111111 11110101 01010101 | | | | | | | | |
| 29. 2 | 7118 | 0.1111111 11110111 11111111 11111111 11111101 01010101 | | | | | | | | |
| 29. 3 | 7119 | 0.1111111 11111111 11111111 11111111 11111111 01010101 | | | | | | | | |
| 29. 4 | 7120 | 0.1111111 11111111 11111111 11111111 11111111 11010101 | | | | | | | | |
| 29. 5 | 7121 | 0.1111111 11111111 11111111 11111111 11111111 11110101 | | | | | | | | |
| 29. 6 | 7122 | 0.1111111 11111111 11111111 11111111 11111111 11111101 | | | | | | | | |
| 29. 7 | 7123 | 0.1111111 11111111 11111111 11111111 11111111 11111111 | | | | | | | | |
| 29. 8 | 7124 | 0.1111111 11111111 11111111 11111111 11111111 11111111 | | | | | | | | |
| 29. 9 | 2715 | BRN SYY SRA RYY TM3 | | | | | | | TRU( 0) | 29 14 |
| 29.10 | 9040 | 0.0000000 00000000 00000000 00000000 00000000 10000000 | | | | | | | | |
| 29.11 | 2700 | YM1 BRN SYY SRB RYY TM1 | | | | | | | Y00( 9) | 11 12 |
| 29.12 | 2705 | YM1 BRN SYY RYY | | | | | | | TRU( 0) | 29 13 |
| 29.13 | 2710 | BRN SYE SRB RYS RRB CPB | | | | | | | TRU( 0) | 14 12 |
| 29.14 | 2720 | YP1 BRN SYY SRB RYY TM3 | | | | | | | Y05(12) | 9 15 |
| 29.15 | 2725 | BRN SYE SRB RYY RRB | | | | | | | TRU( 0) | 10 13 |
| 30. 0 | 1900 | BRN SYS SRA RY5 RRC CPA | | | | | | | BN0( 5) | 1 4 |
| 30. 1 | 1905 | BRN SYS SRA RY5 RRC | | | | | | | TRU( 0) | 30 4 |
| 30. 2 | 1910 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPB TM2 | | | | | | | Y05(12) | 2 4 |
| 30. 3 | 1915 | YP1 BRN SYY SRC SRB RYY RRC RRB PCY CPC TM2 | | | | | | | Y05(12) | 3 4 |
| 30. 4 | 1935 | BRN CON SYS SRC SRB SRA RYS RRC RRB RRA CIC CIB CIA | | | | | | | CB8( 6) | 5 6 |
| 30. 5 | 1940 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA CIC CIA CPC CPA TM2 | | | | | | | BB8( 4) | 7 3 |
| 30. 6 | 1945 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA CIB CPB TM2 | | | | | | | BB8( 4) | 8 2 |
| 30. 7 | 1950 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA PCY CPC CPA TM2 | | | | | | | Y04(11) | 7 9 |
| 30. 8 | 1955 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA PCY CPB TM2 | | | | | | | Y04(11) | 8 10 |
| 30. 9 | 1960 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA PCY CPC CPA TM2 | | | | | | | S80( 7) | 4 11 |
| 30.10 | 1965 | YP1 BRN SYY SRC SRB SRA RYY RRC RRB RRA PCY CPB TM2 | | | | | | | S80( 7) | 4 11 |
| 30.11 | 1970 | BRN SYS SRC SRB SRA RYE RRC | | | | | | | TRU( 0) | 19 12 |
| 30.12 | 1975 | YP1 BRN SYY SRA RYY TM3 | | | | | | | Y05(12) | 12 11 |
| 30.13 | 5500 | YM1 BRN SYY SRB RYY TM4 | | | | | | | Y01(10) | 13 14 |
| 30.14 | 5505 | BRN SYS SRB RY5 RRB | | | | | | | TRU( 0) | 30 15 |
| 30.15 | 5510 | BRN SYY RYS RRB RRA | | | | | | | TRU( 0) | 31 0 |
| 31. 0 | 5520 | BRN SYY SRB RYY | | | | | | | COM( 3) | 1 2 |
| 31. 1 | 5525 | BRN SYS SRC SRB RYS RRC CIC CIB | | | | | | | TRU( 0) | 18 14 |
| 31. 2 | 5540 | BRN SYS SRB RYE RRB | | | | | | | TRU( 0) | 27 10 |
| 31. 3 | 5545 | YP1 BRN SYY SRB RYY PCY CPB TM1 | | | | | | | Y05(12) | 3 4 |
| 31. 4 | 5550 | BRN SYS SRC RY5 RRB | | | | | | | COM( 3) | 2 5 |
| 31. 5 | 5555 | BRN SYY RYY | | | | | | | BN0( 5) | 2 1 |
| 31. 6 | 8800 | BRN SYY SRC SRB SRA RYY RRC RRB RRA CIC CIB CIA | | | | | | | TRU( 0) | 31 7 |
| 31. 7 | 8805 | BRN SYY RYY RRC RRB RRA | | | | | | | AN0( 2) | 8 6 |
| 31. 8 | 8810 | BRN SYY SRB RYY TM1 | | | | | | | BN0( 5) | 9 11 |
| 31. 9 | 8815 | YP1 BRN SYY SRC SRB SRA RYY CPC CPB CPA | | | | | | | BN0( 5) | 10 11 |
| 31.10 | 8820 | BRN SYY RYY | | | | | | | Y00( 9) | 6 12 |

```
31.11   8825  BRN SYY RYY FLG ERR                          TRU( 0)   31  11
31.12   8830  BRN SYY SRC RYY RRC PCY TM1                 TRU( 0)   31  13
31.13   8835  BRN SYY SRB SRA RYY RRB RRA PCY TM4         Y05(12)   15  14
31.14   8840  YF0 BRN SYY RYY                            AN0( 2)    6  12
31.15   8845  YP1 BRN SYY RYY RRC RRB RRA TM1            TRU( 0)   31  12
```

The instruction register is clocked to load the ROM output every 200 nanoseconds. The data word is loaded into the master latch of the instruction register by Clock and into the slave latch by $\overline{\text{Clock}}$. At $\overline{\text{Clock}}$ the contents of the instruction register are used to control the logic (see block diagram). Loading the instruction register occupies one clock cycle (time intervals 1 and 2), as shown in FIG. 73.

As soon as the instruction is in the slave latch of the instruction register, execution begins. A typical execution might read a pair of byte operands, add them during Clock (2) and store the result during the next Clock (3) into the master latch of the specified register. The next $\overline{\text{Clock}}$ (4) would transfer the stored result from master to slave, where it may be used (read) by the next instruction. Notice that there is a time overlap, and the second instruction has already been loaded from ROM (3) into the master latch of the instruction register and execution has begun (4).

The detailed coding of the instruction register, plus definitions of the instruction fields, are given in FIG. 74. Physically, the register is split up and located on three separate cards: ROM address card, D-register card, and FPP interface card.

The floating point algorithms require considerable flexibility for branching from one area of ROM to another. The following paragraphs describe the various modes employed to specify the next address within a current instruction. The addressing modes are shown in Table 12.

Each microinstruction specifies where the next micro-instruction is to be obtained by selecting one of two addresses, dependent on a test condition. One of 16 conditions may be selected for the test by the TS field of the instruction. These conditions are numbered T0 through T15, as shown in Table 10.

TABLE 12. ROM ADDRESSING MODES

| MODE | NEXT ADDRESS PAGE | LINE |
|---|---|---|
| Unconditional Branching (TS = 0) | BP | BL |
| Conditional Branching | | |
| TS True | MP (Current Page) | BL |
| TS False | MP (Current Page) | BP |
| Indirect and Constants | | |
| IND | CS (4–7) Bit 8 = 0 | CS (0–3) |
| CON | CS (4–7) Bit 8 = AD9 | CS (0–3) |
| JSB and Return | | |
| JSB | BP | BL |
| (JSB and RTN complement JAR after execution) | (If JAR = 0, save MP in FP) (If JAR = 1, save MP in GP) | (If JAR = 0, save TS in FL) (If JAR = 1, save TS in GL) |
| RTN | If JAR = 0: GP If JAR = 1: FP | If JAR = 0: GL If JAR = 1: FL |

FIGS. 9A–D show the sources of many of the test inputs; for example, OPC and ENC from the computer, selected outputs of the A, B, C adders, and certain count values of the byte counter YC. These signals are applied as one input to a three-input gate in the conditional branching logic block, as shown in FIGS. 9A–D. (This gate represents a series of gates performing this function.) The second input to the gate is the decoded test number, and the third is the BRN signal (also decoded from the instruction register) which must be true for all branching instructions. If the test is true, bits 0 through 3 of the ROM address are taken from the BL field; if the test is false, these bits are taken from the BP field. Bits 4–8 of the ROM address are taken from the MP register, which contains the page number of the current microinstruction. Thus, conditional branches may be made only to microinstructions on the current page.

Unconditional branch microinstructions give the next ROM address, regardless of test conditions. The page and line of the next microinstruction are given by the BP and BL fields, respectively. The transfer is accomplished by coding BRN and TRU in the microinstruction. This enables the first of the three gates in the unconditional branching logic block, which in turn enables BP and BL onto the page and line address lines.

The jump to subroutine instruction JSB is an unconditional branch with two special provisions: a return address is stored, allowing return from subroutine completion to this address, and register switching to allow one level of JSB nesting.

The decoded JSB signal gates BP and BL onto the page and line address lines in the unconditional branching logic block. The JSB signal also stores the current page value into either the GP or FP register, depending on the current state of the JAR flip-flop, and loads a return address line value from the TS field into either the GL or FL register, depending on the JAR flip-flop state. The TS field can be used to specify the return address line since conditional branches are not allowed with a JSB microinstruction. Thus, a return address is stored in either GP or GL or FP and FL. Since the stored page value is always the current page value, subroutine returns must be to the same page that contained the JSB call. Furthermore, since only four of the five page bits are stored, both the call and return addresses must be in the lower half of ROM (addresses 0 through 255). The subroutine itself, however, may be located on any page, specified in the BP field.

After the return address has been stored, the JAR flip-flop toggles to its complementary state. The initial state of JAR does not matter, as it is immaterial whether the G or F pair of registers is the first selected. The toggling of JAR after each occurrence of JSB insures that the remaining one of the F or G registers will be used to store a second return address. The return from subroutine instruction RTN retrieves a return address from either G or F depending on $\overline{\text{JAR}}$ and branches to that address. RTN also toggles the JAR flip-flop, insuring that a second return address will be retrieved from the remaining one of the F or G registers.

Each word in ROM is either an instruction or constant. Instructions are loaded into the instruction register, and constants are loaded into the D-register (48

bits). To obtain a constant, the ROM program must first load the address of the desired constant into the CS byte of the C-register, and then issue an instruction containing CON in the BC field. The ninth bit of CS is forced to a 1, thus, constants must be in the upper half of ROM (addresses 256 through 511).

The CON signal reads the CS byte onto the ROM address lines, disables Clock for one cycle, and enables a special clock TS that loads the addressed ROM contents into the D-register. See the D-register logic diagram. When the CON bit (R21) is detected, it is loaded by TS into the master CON latch. At Clock, the CON bit is transferred to the slave latch and inverted to disable Clock. The low input to pin 6 (input control code = 01) causes the master latch to clear, so that at the next Clock CON will go false. This, inverted to true by U92B, reenables Clock. The net result is that one Clock pulse has been inhibited, temporarily halting program execution while the constant is being read.

Note, however, that the TS clock has continued to run, and this clock, enabled during the interval that CON is high, loads ROM bits R0 through R47 (the constant) into the D-register.

If a series of constants is called, the CIC bit (R38) may be used to increment the constant address in CS. The purpose of the CKC flip-flop (which, like the CON flip-flop, is clocked by TS) is to assume the function of the CIC flip-flop (disabled in the absence of Clock) during the CON cycle. Note that this feature applies only for CS addresses below 24 (decimal). These 24 locations provide a table that results in a numerical convergence after 25 steps. The S24 signal inhibits CIC for addresses 24 or higher in order to conserve ROM space. The program may continue to call for constants and keep issuing the CIC command, but in effect the contents of location 24 will continue to be read on each further call.

IND reads the 8-bit CS byte onto the ROM address lines. For IND alone, the low half of ROM is addressable (0 to 255) since the ninth address bit (RA8) cannot be controlled by the 8-bit CS register. For CON, however, U61A forces the ninth bit to a 1 (since AD9 is normally 0), so that constants will always be read from the high half of ROM (256 to 511). For certain purposes (such as the ROM dump routine), AD9 can be made true, so that CON can also read the lower half of ROM.

The floating point processor contains three nearly identical arithmetic sections, each typically consisting of a register, a shifter, and an adder. The A-register/shifter/adder will be discussed in detail first; differences for the B and C sections will be described later.

The FPP A-register accommodates 48 bits of data in six separately controllable bytes (A0 through A5). In addition, there is an exponent byte (AE) and a shift byte (AS).

The A-shifter provides a means of selecting any eight consecutive data bits from the 48 bits stored in the FPP A-register starting at any bit position.

The A-adder adds an 8-bit byte, selected from the A-register, to another 8-bit byte, selected from the C register, the B-shifter, the D-shifter, or the A-shifter.

With reference to the block diagram of FIGS. 9A–D, the logic will be discussed left to right across the A-register/shifter/adder block. On the left side of the block is a series of four transfer mode gates (each representing eight separate gates for the complete byte). If transfer mode 1 is selected in the ROM instruction, one of the C-register bytes (on the C50 through C57 lines) will be transferred as an input to the A-adder. (The C byte number selected will be the same as the A byte number selected.) Similarly, if transfer mode 2, 3, or 4 is selected, the gates will transfer a shifted B byte (on the B70 through B77 lines), a shifted D byte (on the D70 through D77 lines) or a shifted A byte (on the A70 through A77 lines).

The output of the Transfer Mode gates is applied to a true/complement network consisting of an "and"/"nor" pair of gates for each bit. If the CPA instruction bit is true, each bit is complemented before being routed to the A-adder on the A90 through A97 lines.

The other input to the A-adder (lines A60 through A67) is enabled if the RRA bit of the instruction register is true. The input consists of one of the FPP A-register bytes on the A50 through A57 lines (byte selection described below).

The result of the addition appears on the A00 through A07 lines, with a possible carry saved in the CY bit register. The carry may be used (propagated) by a PCY instruction in the next cycle. It is also possible to inject a carry (actually an increment by one) by means of a CIA signal. PCY and CIA are functions of the Special field of the instruction word, as is BI8, which can force a one on the eighth bit (A97) of the transferred input to the adder.

Tests which can be made on the A00 through A07 output are: eighth bit true or false, eighth bit of A does or does not equal eighth bit of B, and adder output is zero or non-zero.

The adder output is applied to all eight byte positions of the A-register (the data is stored in a master register at Clock time). At Clock time, the data will be transferred into a byte position (slave register) which is selected by one of eight enabling signals: AY0 through AY5, AYE, or AYS. The enabling signal is derived from the SR, SY, and YC fields of the instruction register. The SR field specifies the A-register (SRA), and SY either specifies byte AS, AE, A5 or else enables the YC field (byte counter) to select one of the six data bytes, A0 through A5. The byte counter produces an octal output on the ROM address card, consisting of signals Y0, Y1, Y2, which is decoded on the FPP interface card to produce the SY0 through SY5 signals. The decoder is not enabled if the SY field specifies SY5 (store in A5 byte), SYS (store in AS byte) or SYE (store in AE byte).

The bytes stored in the FPP A-register can be selectively read out by read signals derived from the RY and YC fields of the instruction register. The RY field either specifies byte AS, AE, or A5 (by RYS, RYE, or RY5 signals), or else enables the decoded byte count from the YC field to select one of the six data bytes (by the RY0 through RY5 signals). The selected byte is routed via the A50 through A57 lines to the A-adder.

In addition to the selective byte reading described in the preceding paragraph, provision is also made to read any adjacent eight bits in the data portion of the A-register. Byte boundaries are ignored, and the register is looked at as a 48-bit data register. Selection is accomplished by the A-shifter, under control of the shift byte (AS) in the A-register.

The shifter may be viewed as an addressable reader. (see FIG. 75.) The numerical value of the shift byte (decimal) enable points to the least significant bit of the desired 8-bit series. This bit and the next higher

seven bits are read out to the transfer mode gates.

As shown in FIG. **75**, special cases occur when the shift byte points to bit positions higher than **40**. (Seven of the eight bits of AS are used for addressable reading, so AS can point to values as high as 127.) When the AS value is between 41 and 47 (inclusive), one or more bits selected at the high end of the series of eight will be nonexistent. These non-existent bits are referred to as phantom bits (P); provision is made to fill these bit positions on the output lines (A**70** through A**77**) with either zeros or copies of the sign bit (bit **47**). The desired choice is made by controlling bit 7 of the shift byte (AS7): if 0, signs will be copied (arithmetic shifting); if 1, zeros will be filled in (logical shifting). Note that when AS is 48 or higher, all of the selected bits will be phantom bits.

Details of the selection process are shown in FIG. **76**. The six least significant bits of AS are decoded octally into two sets of selection signals, designated **SW0** through **SW7** and **SV0** through **SV7**. (**AS6**, if true, would result in the all-phantom condition, so it is not decoded but is simply "or"-tied with **SW6** and **SW7**; see next paragraph.) The **SW0** through **SW5** signals accomplish a preselection of 15 out of the 48 register bits, and the **SV0** through **SV5** signals select 8 out of the 15 preselected bits. These final eight bits are routed out on the A**70** through A**77** lines.

Refer to the A-shifter logic diagram for details on the generation of phantom bits. Note that **U50C** gates sign bit **47** to the higher order **SW5** positions if AS7 is 0. If AS7 is a 1, the output of **U50C** is 0. (Final selection of one or more of these bits is made by the **SV0** through **SV7** signals.) For the all-phantom condition, the shifter network is ignored completely (all zeros on the A**70–77** lines); instead, a true or false TSA signal is sent to the complementing networks. Gate **U50D** is enabled by **SW6, SW7** or **AS6**, and will provide a true output if phantom signs are desired (AS7 = 0) and the sign bit happens to be a 1. Otherwise, if the sign bit is 0 or if phantom zeros are desired (AS7 = 1), TSA will be false. Depending on the transfer mode selected, TSA will affect one of the three complementers (A, B, or C) by inverting the existing all-zero output to all ones (TSA true) or will leave the data as all zeros (TSA false). The result is eight copies of the sign (1 or 0), or eight zeros.

For microprogramming purposes, it is advantageous to have the pointer in AS keep in step with the byte counter. This means that whenever the byte counter is incremented or decremented to enable the next higher or lower byte position, the shift pointer should also change value to enable the next higher or lower series of eight bits. The **AY8** adder performs this function.

In order for the AS value to point to a new series of eight bits, its value must increase by 8 when **YP1** increments the byte counter and must decrease by 8 when **YM1** decrements the byte counter. Furthermore, when the byte counter rolls over from 5 to 0 (incrementing, modulo 6) or from 0 to 5 (decrementing), the AS value must change correspondingly: return to its original value or go to the original value plus 40 (i.e., 5 × 8), respectively.

Referring to the ROM address card logic diagram, FIGS. **14A–R**, note that when **YP1** increments the byte counter (via **U35E**), it also increments the **AY8** adder. Since the **AY8** adder operates on bits 3 through 6 of the AS register (rather than 0 through 3), each increment adds 8 to the contents of AS, via the **AP3** through **AP6** lines. Similarly, when **YM1** decrements the byte counter (by adding all 1's via **U35D/C, U33B**, and **U32A**), it also decrements the **AY8** adder decrementing AS by 8 via the **AP3-6** lines.

Gates **U35E, U33A**, and **U32D** cause the byte counter (and **AY8** and **BY8** adders) to act as modulo **6** counters when incrementing. When the count of 5 is detected by **U24A** and **U15D**, the next **YP1** will inject a quantity which, when added to 5, will produce zero. For the 3-bit byte counter this quantity is 3 (via **U35E** and **U33A**). For the 4-bit **AY8** and **BY8** adders this quantity is 11 (all three gates).

To achieve modulo **6** when decrementing, gates **U33B** and **U32A** are disabled at the count of zero, and allow **U35D** and **U35C** to inject a quantity of 5. This reverts the byte counter to the count of 5 and adds 40 to the AS register via the **AP3** through **AP6** lines. (Incidentally, the **AP3-6** lines are disabled when AS is originally loaded, by the SYSA signal.)

The method by which the byte counter is forced to zero (YFO) is to add the current value of Y to its complement (**U35C, U33C, U16B**) and inject a carry (**U35A**). For the 4-bit adders, **U32B** injects the necessary one-bit for the most significant bit position.

The FPP B-register/shifter/adder is identical to the A section described above, with only signal nomenclature changes and a different assignment of inputs for the transfer modes.

The C-register/adder does not have an associated shifter. Instead, the third shifter is assigned to the D-register. The D-shifter is controlled in parallel with the A-shifter by the AS shift byte.

Since the CS byte is used for indirect addressing of ROM, the CS output is routed to the ROM address card. Also the **S24** signal is made available to the conditional branching test logic.

On the C-adder card, the CSX line is open 0, whereas on the A- and B-adder cards ASX and BSX are enabled by tying to +4.75 volts. This disables the CP input lines to the shift byte (CS), since these lines are not used in the C arithmetic section.

As mentioned above, data is transferred into or out of the FPP in three successive 16-bit words. It was also stated that the COMP sends 16 bits of data with every ENC, and the FPP returns 16 bits of data with every FLG, whether or not data is actually used at either end. Data to the FPP is loaded into the C-register (and transferred to the B-register if a load opcode follows), and is sent from the B-register. Referring to table **13**, the process is as follows:

TABLE 13. X REGISTER TRANSFER SEQUENCE

| ENC | INPUT | OUTPUT |
|---|---|---|
| No. 1 | IM0-7 → C5 | B5 → A4 |
| | IL0-7 → C4 | A4 (FLG) |
| No. 2 | IM0-7 → C3 | B3 → A2 |
| | IL0-7 → C2 | B2 (FLG) |
| No. 3 | IM0-7 → C1 | B1 → A0 |
| | IL0-7 → Convert to FPP format → C0 | B0 Convert to FPP format (FLG) |

On the first ENC, the entry routine first loads the high order eight bits (IM0 through IM7) into C5 while, simultaneously, B5 is transferred to A4 and read out to the output lines (A50 through A57). Then the byte

counter is decremented (pointing to byte 4). The low order input bits (IL0-7) are loaded into C4, and B4 is read out to the B50 through B57 lines. A FLG signal is issued to the COMP, telling it that it can store the 16 bits from A4 and B4.

On the second ENC, the byte counter decrements to 3, and IM0-7 is loaded into C3, while B3 is transferred to A2. Decrementing to count 2 allows C2 to be loaded, and A2 and B2 to be read out (with FLG).

On the third ENC, the byte counter decrements to 1, and IM0-7 is loaded into C1. Then, when the byte counter decrements to 0, a format conversion occurs which moves the exponent sign bit to the proper position. (Internally in the FPP, this bit is in the most significant bit position; externally in the computer, it is in the least significant bit position.) Byte B1 is now transferred to A0, and A0 and B0 are read out (with FLG).

## POWER SUPPLY

The power supply of the floating point processor generates two regulated dc supply voltages for all logic circuits in the unit: +4.75 volts and −2 volts. (A third dc voltage, +10 volts, is also generated, but this supply is used only within the power supply itself.)

FIG. 77 illustrates the power supply circuits in simplified form. The 115- or 230-volt ac input is stepped down to a nominal 12 volts ac and rectified by a pair of silicon-controlled rectifiers (SCR). The inductor/-capacitor filtered output is 6.75 volts dc, referenced to ground such that the positive line is +4.75 volts and the negative line is −2 volts with respect to ground.

The full 35-ampere current capacity is available to the +4.75-volt load, and up to 35 amperes is available to the +2-volt load. Since the −2volt load is less than the +4.75-volt load, the difference current is diverted through the −2-volt shunt regulator. This regulator acts in the same way as would a Zener diode. A variable amount of current is drawn through the shunt in order to maintain a constant −2-volt level.

The level of the +4.75 voltage is maintained constant by controlling the conduction time of the SCR. The +4.75-volt level is detected by a differential amplifier, which compares the voltage with a Zener diode reference. The difference output is used to control the slope of a ramp voltage, which is synchronized to the 120 Hz rectified line frequency. When the ramp reaches the trigger level of a unijunction transistor in the ramp generator, the ramp terminates, generating a positive pulse of about 10 volts amplitude and 20 microseconds duration. This pulse triggers the SCR's, which will then continue to conduct for the remainder of the half cycle. As shown in FIG. 77 (note examples of ramp slope and rectified sine wave), a variance of ramp slope has the net result of altering the conduction time (shaded area). Consequently, the energy delivered to the LC filter will be increased or reduced proportionately, thus providing the means of controlling the output dc level.

Referring now to FIGS. 40 A–J, input ac power is applied to power line assembly A1. This snap-in module contains the ac line connector, line fuse, rf interference filter, 115/230V line voltage switch, and terminals for connection of the front panel POWER switch, power-on indicator lamp (DS1), and power transformer. Relay K1 is inserted in series with the transformer primary, so that power will be turned off if either the computer loses power (−23.8V drops) or the ambient temperature in th FPP unit rises too high.

Sensing of the +4.75-volt level is made from a point on the backplane bus. Due to the high currents involved, bus resistance itself will drop the dc level slightly; power is therefore applied to the bus at two points, and the sense line is connected to a point that represents an average value.

The sensed +4.75 voltage is applied to a differential amplifier at Q1/Q2, which compares a divided sample (R20/R21) to a presettable reference level from resistor R25 (+4.75V ADJ). Any voltage difference between the bases of Q1 and Q2 is amplified and applied to Q3, altering the charging rate of ramp capacitor C30. When Q3 has charged C30 to the triggering level of unijunction transistor Q4, Q4 discharges C30 to the −2-volt clamping level. The sharp negative transition at the base of Q5 turns on Q5 for about 20 microseconds, dependent on circuit constants, and the resultant positive pulse is applied through emitter follower Q6 to the SCR trigger inputs (CR5, CR6). Diode CR22 limits the pulse amplitude to +10 volts and protects Q6; CR8 protects the SCR's (which are non-conducting before the pulse arrives).

The positive pulse turns on CR5 or CR6 (depending on the ac cycle polarity), charging filter capacitors C11, C12, and C13 through inductor L4. At the end of the half cycle, ac polarity reverses and the SCR ceases conduction. Since the other SCR will not begin its conduction until triggered, neither SCR is conducting at this time. The inductive field of L4 begins to collapse, building up a reverse voltage which could be destructive if protection were not provided. Diode CR7 provides this protection by coming into conduction when the reverse voltage exceeds the −2-volt level, and provides a current path back to the filter capacitors. Thus, even when both SCR's are off, the inductor still delivers current to the load. When the next SCR is triggered, it abruptly puts out a positive voltage to the inductor, and thus reverse biases CR7. In summary: CR7 conducts when the SCR's are not conducting.

As explained above, the timing of the SCR trigger accomplishes the voltage regulating function.

The primary purpose of Q7 is to synchronize the unijunction oscillator to twice the line frequency. A secondary function is to inhibit the triggering of unijunction transistor Q4 when the crowbar is on, thus reducing current delivered to the Crowbar, CR80. When the input voltage (pulsating dc from the input to L4) is in excess of +9 volts, Q7 is saturated (on), providing a low impedance path for the Q3 collector current, diverting it from C30. Thus the unijunction oscillator is held in the off state. (Note that a positive input from the crowbar, via Q29, could permanently hold the oscillator in this off state.) Then, when the pulsating voltage drops below +8 volts, Q7 is cut off, and the current from the Q3 collector is shunted to ramp capacitor C30. This results in a voltage ramp on the emitter of Q4, the slope of which (as discussed earlier) is determined by the collector current of Q3. The start of the ramp is therefore determined by the on-to-off transition of Q7, which occurs twice for each cycle of the line.

The −2-volt sense voltage referred to above is applied through a presettable divider to the base of Q18. The bottom end of the divider is held constant by a Zener diode reference. The −2-volt adjustment resistor is set so that the Q18 base is at zero volts when the −2-volt output is at its nominal value. This zero-volt-level

is compared with the zero-volt ground at the emitter of Q19. Any difference is amplified by Q19, Q20, and Q21, altering the flow of shunt current through Q22. The direction of change (more or less current) is such as to maintain a fixed voltage value on the −2-volt sense line. As mentioned earlier (paragraph 4—94), the circuit acts like a Zener diode in maintaining a fixed voltage output. About 5 amperes is passed through Q22.

Transistors Q8 and Q9 are normally conducting. When an unusual current drain increases the dc voltage drop across inductor L4 to a specific level (determined by the selected values of R40 through R44), Q8 and Q8 will be biased off. Under this condition, CR35 clamps the unijunction input to a level that is below the trigger point. No pulses are therefore applied to the SCR's, and no further conduction occurs. Both +4.75-volt and −2-volt outputs are thus cut off.

There are three separate circuits involved in detecting and acting on out of limit dc voltage conditions. These three circuits (−2V limit sense, +4.75V limit sense, and crowbar) are discussed together under the current heading.

FIG. 78 illustrates the actions that occur when either the +4.75 or −2 voltages go out of limits. When the +4.75 voltage (applied to Q23/Q24 bases) rises too high, to a level set by R91, Q24 will conduct and activate the power fail circuit (discussed later under paragraph 4–110). Similarly, if the +4.75 voltage drops below a negative limit set by R92, Q23 will conduct and activate the power fail circuit. In the −2V limit sense circuit, if the −2 voltage (applied to the top of the divider), becomes too positive, to a level set by R61, Q14 will conduct and activate the power fail circuit. The negative limit sensing circuit uses a normally conducting emitter follower (Q13). When the −2 voltage becomes too negative, Q15 will conduct and activate the power fail circuit.

If the +4.75 voltage becomes excessively positive (above about +6 volts), or if the −2 voltage becomes excessively negative (more than about −3 volts), the crowbar circuit triggers and cuts off both supplies.

The crowbar circuit uses an SCR diode (CR80). When the −2-volt level goes more negative than the breakdown level of CR82 (normally an effective open circuit), CR82 causes Q30 (and Q31) to conduct. Or, if the +4.75-volt level goes more positive than the breakdown level of CR81, Q31 will again be caused to conduct. This is because both emitter and base voltages increase together as the +4.75 voltage rises; then CR81 breaks down and holds the base low. When, from either cause, Q31 conducts, SCR diode CR80 is triggered, effectively short-circuiting the +4.75V and −2V supplies together. This protects logic circuits from overvoltage damage. To prevent the rectifiers from delivering any more current to this short circuit, Q29 (which goes into conduction when the SCR triggers) inhibits the sync amplifier. Transistor Q7 is driven into saturation, thus preventing further trigger pulses to SCR rectifiers CR5 and CR6.

Diodes CR60 and CR61 rectify a sample of the transformer secondary output, and the resulting pulsating direct voltage is applied to two RC filters. One filter (R70, C50) has a short time constant, and the other (R71, R72, C51) has a long time constant. The filters are isolated from each other by CR63. As a result (see FIG. 79), a dc voltage representing the peak value of the rectified ac is present at the emitter of Q16, and a partially filtered waveform is present at the base. Normally (see half-cycle number 1), the exponential decay is not sufficient to cause conduction of Q16 before the next half-cycle restores the C50 charge. If, however, at least two half-cycles are missed (assume ac power is lost at the end of half-cycle number 2), the base voltage will drop to the point where conduction of Q16 will occur. With Q16 conducting, Q17 will also be turned on, thus activating the power fail circuit.

When any of the previously discussed voltage sensing circuits indicate a failure, Q26 is caused to conduct. (Note that four of the sources, Q14, Q15, Q17, and Q23, require inversion by Q25, whereas the Q24 source does not.) The conduction of Q26 in turn causes the other four transistors in the power fail circuit to conduct. The EPF signal (normally low) goes high to initiate a power fail interrupt in the computer. A few milliseconds later, EP0 (normally high) goes low; when power is restored, EP0 will go high again, initiating a restart sequence in computers which have the restart option installed and enabled.

Several circuits in the power supply require a +10-volt operating voltage. To supply this, the transformer secondary is rectified by CR40 and CR41, filtered by C45, and regulated by Q10. The control for Q10 is the differential amplifier consisting of Q11 and Q12. The reference voltage provided by CR42 is compared with a divided sample of the +10V output, and any difference is applied as a correction signal to the base of Q10.

## INTERFACE INFORMATION

ENC: Initiates the reception of an operand word through the 16 bit input port and the transmission of a result word through the 16 bit output port.

FIRST OPERAND WORD: FPP receives through the 16 bit input port the 16 most significant magnitude bits (including sign) of a 48 bit floating point operand (reception initiated by the 1st, 4th, 7th, etc. ENC command after the last preceding OPC command).

SECOND OPERAND WORD: FPP receives through the 16 bit input port the 16 middle significant magnitude bits of a 48 bit floating point operand (reception initiated by the 2nd, 5th, 8th, etc. ENC command after the last preceding OPC command).

THIRD OPERAND WORD: FPP receives through the 16 bit input port the 8 least significant magnitude bits and the 8 exponent bits of a 48 bit floating point operand (reception initiated by the 3rd, 6th, 9th, etc., ENC command after the last preceding OPC command).

OPC: Initiates the reception of an operation code through the 16 bit input port and initiates the execution of that operation.

OPERATION CODE: FPP receives through the 16 bit input port an 8 bit operation code contained in bits 0–7 (reception initiated by the OPC command).

FLG: Indicates if following and ENC command that the reception of an operand word is complete and that the transmission of a result word has begun, or if following an OPC command that the execution of an operation is complete.

FIRST RESULT WORD: FPP transmits through the 16 bit output port the 16 most significant magnitude bits (including sign) of a 48 bit floating point result

(transmission initiated by the 1st, 4th, 7th, etc. ENC command after the last preceding OPC command).

SECOND RESULT WORD: FPP transmits through the 16 bit output port the 16 middle significant magnitude bits of a 48 bit floating point result (transmission initiated by the 2nd, 5th, 8th, etc. ENC command after the last preceding OPC command).

THIRD RESULT WORD: FPP transmits through the 16 bit output port the 8 least significant magnitude bits and the 8 exponent bits of a 48 bit floating point result (transmission initiated by the 3rd, 6th, 9th, etc. ENC command after the last preceding OPC command).

ERR: Indicates that an error or special condition has been encountered during the execution of an operation and that the transmission of an error code has begun.

ERROR CODE: FPP transmits through the 16 bit output port an 8 bit error code contained in bits 8–15 (transmission initiated by an error or special condition encountered during the execution of an operation).

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A7 | | TEST CARD | 28480 | 02152-60008 |
| A7C1- | | | | |
| A7C4 | 46 | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 150D225X9020A2-DYS |
| A7C5- | | | | |
| A7C11 | 10 | C:FXD CER DISC 1000 PF +80-20% 1000VDCW | 56289 | C067B102E102ZE19-CDH |
| A7R13 | 9 | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R14 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R16 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R18 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R21 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R23 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A7R25- | | | | |
| A7R36 | 88 | RESISTOR NETWORK | 28480 | 1810-0047 |
| A7U11 | 31 | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A7U13 | 99 | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U14 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U15 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U21 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A7U23 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U24 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U25 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U31 | 11 | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A7U32 | 114 | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A7U33 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U34 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U35 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U41 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A7U42 | 10 | IC:CTL DUAL RANK J-K FLIP-FLOP | 07263 | SL3464 |
| A7U43 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U44 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A7U51 | 22 | INTEGRATED CIRCUIT: | 28480 | 1820-0186 |
| A7U52 | | IC:CTL DUAL RANK J-K FLIP-FLOP | 07263 | SL3464 |
| A7U53 | 2 | INTEGRATED CIRCUIT:TTL 6 BIT COMP | 28480 | 1820-0250 |
| A7U54 | | INTEGRATED CIRCUIT:TTL 6 BIT COMP | 28480 | 1820-0250 |
| A7U55 | 4 | IC:TTL QUAD | 28480 | 1820-0141 |
| A7U61 | 24 | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A7U62 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A7U63 | | IC:TTL QUAD | 28480 | 1820-0141 |
| A7U64 | | IC:TTL QUAD | 28480 | 1820-0141 |
| A7U65 | | IC:TTL QUAD | 28480 | 1820-0141 |
| A8 | | ROM ADDRESS CARD | 28480 | 02152-60005 |
| A8C1- | | | | |
| A8C4 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 150D225X9020A2-DYS |
| A8Q1 | 10 | TSTR:SI PNP | 80131 | 2N3640 |
| A8Q2 | | TSTR:SI PNP | 80131 | 2N3640 |
| A8R1 | | | | |
| A8R9 | 42 | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A8R10- | | | | |
| A8R38 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A8U11 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U12 | 28 | IC:CTL TRIPLE 2-2-3-INPUT AND GATE | 07263 | SL3456 |
| A8U14 | 3 | INTEGRATED CIRCUIT:BINARY FULL ADDER | 01295 | SN7483N |
| A8U15 | | INTEGRATED CIRCUIT:BINARY FULL ADDER | 01295 | SN7483N |
| A8U16 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A8U17 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U21 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U22 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U23 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U25 | | INTEGRATED CIRCUIT:BINARY FULL ADDER | 01295 | SN7403N |
| A8U26 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A8U27 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U32 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U33 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U34 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A8U35 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A8U36 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U38 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U37 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U41 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U42 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A8U43 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U44 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U46 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U47 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U51 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U52 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U53 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U54 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A8U55 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A8U56 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U57 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U61 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U62 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U63 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U64 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U65 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U66 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A8U67 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A8U72 | | IC:CTL TRIPLE 2-?-3-INPUT AND GATE | 07263 | SL3456 |
| A8U73 | | IC:CTL TRIPLE 2-?-3-INPUT AND GATE | 07263 | SL3456 |
| A8U74 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U75 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U76 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U77 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U81 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U82 | | IC:CTL TRIPLE 2-?-3-INPUT AND GATE | 07263 | SL3456 |
| A8U83 | | IC:CTL TRIPLE 2-?-3-INPUT AND GATE | 07263 | SL3456 |
| A8U84 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U85 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U86 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U87 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U91 | | INTEGRATED CIRCUIT: | 28480 | 1820-0186 |
| A8U92 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U93 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U94 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U95 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A8U96 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A8U97 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A9 | | ROM CARD | 28480 | 02152-60007 |
| A9C1 - A9C4 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 15CD225X9020A2-DYS |
| A9R1 | 2 | R:FXD FLM 3.32K 0.5% 1/4W | 28480 | 0698-5615 |
| A9R10 | | R:FXD FLM 3.32K 0.5% 1/4W | 28480 | 0698-5615 |
| A9U13 | 1 | INTEGRATED CIRCUIT | 28480 | 94-19-144 |
| A9U15 | 1 | INTEGRATED CIRCUIT | 28480 | 82-19-144 |
| A9U23 | 1 | INTEGRATED CIRCUIT | 28480 | 93-19-144 |
| A9U25 | 1 | INTEGRATED CIRCUIT | 28480 | 81-19-144 |
| A9U35 | 1 | INTEGRATED CIRCUIT | 28480 | 136-22-144 |
| A9U43 | 1 | INTEGRATED CIRCUIT | 28480 | 139-22-142 |
| A9U45 | 1 | INTEGRATED CIRCUIT | 28480 | 135-22-144 |
| A9U53 | 1 | INTEGRATED CIRCUIT | 28480 | 90-19-144 |
| A9U55 | 1 | INTEGRATED CIRCUIT | 28480 | 78-19-144 |
| A9U63 | 1 | INTEGRATED CIRCUIT | 28480 | 138-22-144 |
| A9U65 | 1 | INTEGRATED CIRCUIT | 28480 | 77-19-144 |
| A9U73 | 1 | INTEGRATED CIRCUIT | 28480 | 88-19-144 |
| A9U75 | 1 | INTEGRATED CIRCUIT | 28480 | 76-19-144 |
| A9U83 | 1 | INTEGRATED CIRCUIT | 28480 | 87-19-144 |
| A9U85 | 1 | INTEGRATED CIRCUIT | 28480 | 75-19-144 |
| A9U95 | 1 | INTEGRATED CIRCUIT | 28480 | 134-22-144 |
| A9U95 | 1 | INTEGRATED CIRCUIT | 28480 | 137-22-143 |
| A9U97 | 3 | INTEGRATED CIRCUIT:LEVEL TRANSLATOR | 04713 | MC10102P |
| A9U103 | 1 | INTEGRATED CIRCUIT | 28480 | 85-19-144 |
| A9U105 | 1 | INTEGRATED CIRCUIT | 28480 | 133-22-144 |
| A9U107 | | INTEGRATED CIRCUIT:LEVEL TRANSLATOR | 04713 | MC10102P |
| A9U113 | 1 | INTEGRATED CIRCUIT | 28480 | 84-19-144 |
| A9U115 | 1 | INTEGRATED CIRCUIT | 28480 | 72-19-144 |
| A9U125 | 1 | INTEGRATED CIRCUIT | 28480 | 71-19-144 |
| A9U126 | | INTEGRATED CIRCUIT:LEVEL TRANSLATOR | 04713 | MC10102P |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A10 | | D REGISTER CARD | 28480 | 02152-60004 |
| A10C1 | 3 | C:FXD MICA 63 PF 5% | 28480 | 0140-0192 |
| A10C4 | 1 | C:FXD CER 1000PF 5% 50VDCW | 28480 | 0160-2588 |
| A10C6 - A10C9 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 1500225X9020A2-DYS |
| A10L1 | 1 | COIL:MOLDED CHOKE 0.20 UH 10% | 28480 | 9140-0105 |
| A10Q1 - A10Q3 | 3 | TSTR:SI NPN | 80131 | 2N708 |
| A10Q4 - A10Q7 | | TSTR:SI PNP | 80131 | 2N3640 |
| A10R5 | 1 | R:FXD COMP 1000 OHM 5% 1/4W | 01121 | CB 1025 |
| A10R6 - A10R9 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A10R10 - A10R12 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A10R13 | | | | |
| A10R22 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A10R23 - A10R25 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A10R26 | | | | |
| A10R33 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A10R34 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A10R35 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A10U11 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U14 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U15 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U21 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U24 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U25 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U34 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U35 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U44 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U51 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U53 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A10U54 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10U55 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U61 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U62 | | IC:CTL DUAL BASE J-K FLIP-FLOP | 07263 | SL3464 |
| A10U63 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U64 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U65 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U72 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10U73 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10U74 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U75 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U81 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U83 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U84 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U85 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A10U92 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A10U93 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10U94 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10U95 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A10Y1 | | CRYSTAL:QUARTZ 10MC/S 0.005% | 28480 | 0410-0035 |
| A11 | 3 | SHIFTER CARD | 28480 | 02152-60001 |
| A11C1- A11C4 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 1500225X9020A2-DYS |
| A11R1 - A11R6 | 26 | R:FXD COMP 470 OHM 5% 1/4W | 01121 | CB 4715 |
| A11U12 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A11U13 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A11U15 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U16 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A11U17 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U21 | | | | |
| A11U25 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U26 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A11U31 | | | | |
| A11U35 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U36 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A11U37 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U41 | | | | |
| A11U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U46 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A11U47 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U51 | | | | |
| A11U55 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U56 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A11U57 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U61 | | | | |
| A11U65 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U66 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A11U67 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U71 | | | | |
| A11U75 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U76 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A11U77 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U81 | | | | |
| A11U85 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U86 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A11U87 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U91 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U92 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U93 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A11U94 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U95 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U96 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A11U97 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A11U103 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A11U104 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| | | A13 SAME AS A11<br>A15 SAME AS A11 | | |
| A12 | 3 | ADDER CARD | 28480 | 02152-60002 |
| A12C1 - | | | | |
| A12C4 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 15CD225X9020A2-DYS |
| A12Q1 | | TSTR:SI PNP | 80131 | 2N3640 |
| A12Q2 | | TSTR:SI PNP | 80131 | 2N3640 |
| A12R1 - | | | | |
| A12R12 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A12R13 - | | | | |
| A12R31 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A12U3 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U4 | | IC:CTL TRIPLE 3-2-3-INPUT AND GATE | 07263 | SL3456 |
| A12U5 | | IC:CTL TRIPLE 2-2-3-INPUT AND GATE | 07263 | SL3456 |
| A12U6 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U11 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U12 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U14 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U16 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U21 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U22 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U23 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U24 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A12U25 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U26 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U31 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U32 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U33 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U34 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A12U35 | | IC:CTL TRIPLE 2-2-3-INPUT AND GATE | 07263 | SL3456 |
| A12U36 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U41 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U42 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U43 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U44 | | IC:CTL TRIPLE 2-2-3-INPUT AND GATE | 07263 | SL3456 |
| A12U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U46 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U51 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U52 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U53 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A12U54 | | IC:CTL TRIPLE 2-2-3-INPUT AND GATE | 07263 | SL3456 |
| A12U55 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U56 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U61 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U62 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U63 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U64 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A12U65 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U66 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U71 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U72 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U73 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U74 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A12U75 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U76 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U81 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U82 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U83 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U84 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U85 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A12U86 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U91 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U92 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U93 | | IC:CTL QUAD 2-INPUT AND GATE | 07263 | SL3462 |
| A12U94 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U95 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U96 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A12U101 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U102 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U103 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U104 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U105 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A12U106 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U111 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U113 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U113 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U114 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12J115 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U116 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U121 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U122 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U123 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A12U124 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A12U125 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A12U126 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17 | | I/O INTERFACE CARD | 28480 | 02152-60017 |
| A17C1 | | C:FXD CER DISC 1000 PF +80-20% 1000VDCW | 56289 | C067B102E102ZE19-CDH |
| A17C2 | | C:FXD CER DISC 1000 PF +80-20% 1000VDCW | 56289 | C067B102E102ZE19-CDH |
| A17C3 - A17C6 | | C:FXD ELECT .2 UF 10% 20VDCW | 56289 | 1500225X9020A2-DYS |
| A1701 | | TSTR:SI PNP | 80131 | 2N3640 |
| A1702 | | TSTR:SI PNP | 80131 | 2N3640 |
| A17R1 - A17R16 | | R:FXD COMP 470 OHM 5% 1/4W | 01121 | CB 4715 |
| A17R17 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A17R18 | | R:FXD COMP 470 OHM 5% 1/4W | 01121 | CB 4715 |
| A17R19 | | R:FXD COMP 100 OHM 5% 1/4W | 01121 | CB 1015 |
| A17R20 - A17R22 | | R:FXD COMP 470 OHM 5% 1/4W | 01121 | CB 4715 |
| A17R23 - A17R28 | | RESISTOR NETWORK | 28480 | 1810-0047 |
| A17R29 - A17R40 | | R:FXD COMP 330 OHM 5% 1/4W | 01121 | CB 3315 |
| A17U21 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U24 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U25 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A17U31 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U32 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A17U33 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A17U34 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A17U35 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A17U41 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U42 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17U43 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A17U44 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U45 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U51 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U52 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17U53 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17U54 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U55 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U61 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U62 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17U63 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A17U64 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A17U65 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15964 |
| A17U71 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15961 |
| A17U72 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15965 |
| A17U73 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15966 |
| A17U74 | | INTEGRATED CIRCUIT: CTL | 07263 | SL15963 |
| A17U75 | | INTERGRATED CIRCUIT: | 28480 | 1820-0186 |
| A3 | | POWER SUPPLY REGULATOR CARD | 28480 | 02125-60009 |
| A3C25 | 1 | C:FXD ELECT 50 UF 10% 6VDCW | 56289 | 150D606X9006B2 |
| A3C26 | 1 | C:FXD MY 0.001 UF 10% 200VDCW | 56289 | 192P10292-PTS |
| A3C30 | 1 | C:FXD MY 0.022 UF 10% 200VDCW | 56289 | 192P22392-PTS |
| A3C31 | 1 | C:FXD CER 0.47 UF +80-20% 25VDCW | 56289 | 5C1187S-CKL |
| A3C32 | 3 | C:FXD ELECT 100UF +100%-10% 15VDCW | 56289 | 30107G015DD4 |
| A3C40 | 1 | C:FXD ELECT 200 UF +75-10% 25VDCW | 28480 | 0180-2144 |
| A3C45 | 1 | C:FXD ELECT 350 UF +75-10% 50VDCW | 28480 | 0180-2217 |
| A3C46 | | C:FXD ELECT 100UF +100%-10% 15VDCW | 56289 | 30107G015DD4 |
| A3C50 | 1 | C:FXD ELECT 1.0UF 10% 35VDCW | 56289 | 150D105X9035A2-DYS |
| A3C51 | 1 | C:FXD ELECT 50 UF +75-10% 50VDCW | 56289 | 30D506G050CD2-DSM |
| A3C60 | 1 | C:FXD CER 0.05 UF +80-20% 100VDCW | 51418 | TA |
| A3C61 | | C:FXD ELECT 100UF +100%-10% 15VDCW | 56289 | 30107G015DD4 |
| A3C70 | 1 | C:FXD ELECT 10.0 UF 10% 20VDCW | 56289 | 150D106X9C2CB2-76 |
| A3C71 | 1 | C:FXD ELECT 5.8 UF 10% 35VDCW | 56289 | 150D685X9035B2-DYS |
| A3C72 | 3 | C:FXD CER 0.1 UF +80-20% 50VDCW | 56289 | 5C50C1S-CKL |
| A3C73 | | C:FXD CER 0.1 UF +80-20% 50VDCW | 56289 | 5C50C1S-CKL |
| A3CR20 | 2 | DIODE BREAKDOWN:4.64V 5% | 28480 | 1902-3082 |
| A3CR21 | 10 | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR22 | 6 | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR30 | 1 | DIODE:BREAKDOWN 8.25V 5% | 04713 | SZ10939-158 |
| A3CR35 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR40 | | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR41 | | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR42 | | DIODE BREAKDOWN:4.64V 5% | 28480 | 1902-3082 |
| A3CR50 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR60 | | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR61 | | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR62 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR63 | | DIODE:SILICON 0.75A 200 PIV | 28480 | 1901-0158 |
| A3CR70 - A3CR73 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR81 | 1 | DIODE:BREAKDOWN 6.19V 5% | 04713 | SZ10939-122 |
| A3CR82 | 1 | DIODE:BREAKDOWN 7.61V 5% | 04713 | SZ10939-14 |
| A3CR95 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A3CR96 | | DIODE:SILICON 30MA 30WV | 07263 | FDG1088 |
| A301 | 15 | TSTR:SI NPN | 80131 | 2N3054 |
| A302 | | TSTR:SI NPN | 80131 | 2N3054 |
| A303 | 10 | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A304 | 1 | TSTR:SI | 80131 | 2N2646 |
| A305 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A306 | 3 | TSTR:SI PNP | 80131 | 2N3638A |
| A307 | | TSTR:SI NPN | 80131 | 2N3054 |
| A308 | | TSTR:SI NPN | 80131 | 2N3054 |
| A309 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3010 | 1 | TSTR:SI PNP | 02735 | 38640 |
| A3011 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3012 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3013 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3014 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3015 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3016 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3017 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3018 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3019 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3020 | | TSTR:SI PNP | 80131 | 2N3638A |
| A3023 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3024 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3025 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3026 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3027 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3028 | | TSTR:SI NPN | 80131 | 2N3C54 |
| A3029 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3030 | | TSTR:SI PNP | 80131 | 2N3638A |
| A3031 | 1 | TSTR:SI PNP(SELECTED FROM 2N1132) | 28480 | 1853-0001 |
| A3032 | | TSTR:SI NPN | 80131 | 2N3054 |
| A3033 | | TSTR:SI PNP(SELECTED FROM 2N3702) | 28480 | 1853-0020 |
| A3R20 | 1 | R:FXD MET FLM 511 OHM 1% 1/8W | 14674 | C4 |
| A3R21 | 1 | R:FXD MET FLM 3.83K 1% 1/8W | 91637 | MFF-1/10-32 |
| A3R22 | 1 | R:FXD MET FLM 26.1K OHM 1% 1/8W | 75042 | CEA |

## PARTS LIST

| Reference Designation | Qty | Description | Mfr Code | Mfr Part Number |
|---|---|---|---|---|
| A3R23 | 4 | R:FXD MET FLM 10.0K 1% 1/8W | 14674 | C4 |
| A3R24 | 1 | R:FXD MET FLM 4.15 OHM 1% 1/8W | 91637 | MF-1/10-32 |
| A3R25 | 7 | R:VAR FLM 500 OHM 10% LIN 1/2W | 28480 | 2100-1788 |
| A3R26 | 1 | R:FXD MET FLM 2.15K 1% 1/8W | 14674 | C4 |
| A3R27 | 2 | R:FXD MET FLM 1.96K OHM 1% 1/8W | 14674 | C4 |
| A3 | | POWER SUPPLY REGULATOR CARD | 28480 | 02152-60009 |
| A7 | | TEST CARD | 28480 | 02152-60008 |
| A8 | | ROM ADDRESS CARD | 28480 | 02152-60005 |
| A9 | | ROM CARD | 28480 | 02152-60007 |
| A10 | | D REGISTER CARD | 28480 | 02152-60004 |
| A11 | | SHIFTER CARD | 28480 | 02152-60001 |
| A12 | | ADDER CARD | 28480 | 02152-60002 |
| A13 | | SHIFTER CARD | 28480 | 02152-60001 |
| A14 | | ADDER CARD | 28480 | 02152-60002 |
| A15 | | SHIFTER CARD | 28480 | 02152-60001 |
| A16 | | ADDER CARD | 28480 | 02152-60002 |
| A17 | | I/O INTERFACE CARD | 28480 | 02152-60017 |
| A109 | | EAU TIMING CARD | 28480 | 02152-60012 |
| A110 | | EAU LOGIC CARD | 28480 | 02152-60011 |
| A209 | | EAU I/O CARD | 28480 | 02152-60013 |
| B1 | 2 | FAN:TUBEAXIAL 115V 60 HZ | 23480 | 3160-0072 |
| B2 | | FAN:TUBEAXIAL 115V 60 HZ | 23480 | 3160-0072 |
| C7 | 1 | C:FXD CER 2.2 UF 20% 25VDCW | 56289 | 5C152C2S-CML |
| C11 | 1 | | | |
| C13 | 3 | C:FXD ELECT 160,000 UF +75-10% 10VDCW | 56289 | 36D164G010DF2A-DQB |
| C62 | | C:FXD ELECT 2.2 UF 10% 20VDCW | 56289 | 15CD225X9020A2-DYS |
| C80 | 1 | C:FXD ELECT 330 UF 10% 6VDCW | 28480 | 0180-1714 |
| CR5 | 1 | DIODE: SCR. 40RC510 | 00000 | OBD |
| CR6 | 1 | DIODE: SCR. 40RC510 | 00000 | OBD |
| CR7 | 1 | DIODE: 1N1192AR | 00000 | OBD |
| CR8 | 1 | DIODE:SILICON 100MA/1V | 07263 | FD 2387 |
| CR80 | 1 | THYRISTOR:SCR 50V 25A | 28480 | 1884-0046 |
| CR90 | 1 | DIODE:SILICON 100PIV 1N3209R | 04713 | 1N3209R |
| CR91 | 1 | DIODE:SILICON 1N3209 | 04713 | 1N3209 |
| DS1 | 1 | LIGHT:INDICATOR SELECTED NE-2H | 28480 | 1450-0419 |
| L3 | 1 | RF CHOKE ASSEMBLY | 28480 | 02152-60023 |
| L4 | 1 | CHOKE 3MH | 28480 | 5081-0372 |
| Q21 | 1 | TRANSISTOR SI PNP | 04713 | SS9320 |
| Q22 | 1 | TSTR:SI NPN | 80131 | 2N3055 |
| R1 | | R:FXD COMP 33K OHM 5% 1/4W | 01121 | CB 3335 |
| R2 | 2 | R:FXD MET FLM 61.9 OHM 1% 1/8W | 28480 | 0757-0276 |
| R5 | 2 | R:FXD MET FLM 40 OHM 1% 1/8W | 28480 | 0757-0346 |
| R6 | | R:FXD MET FLM 10 OHM 1% 1/8W | 28480 | 0757-0346 |
| R7 | 1 | R:FXD COMP 2.7 OHM 10% 1/4W | 01121 | CB 27G1 |
| R8 | 1 | R:FXD MET FLM 100 OHM 1% 1/8W | 14674 | C4 |
| R82 | | R:FXD MET FLM 61.9 OHM 1% 1/8W | 28480 | 0757-0276 |
| R122 | 1 | R:FXD WW 0.1 OHM 5% 25W | 28480 | 0811-2510 |
| R130 | 1 | R:FXD FLM 10 OHM 1% 1/8W | 28480 | 0757-0346 |
| S1 | 1 | SWITCH:TOGGLE DPST (ON-NONE-OFF) | 04009 | 81024-GT |
| T1 | 1 | TRANSFORMER | 28480 | 5080-0378 |

I claim:

1. A floating point CORDIC processor for calculating trigonometric, hyperbolic, and linear elementary functions, said floating point CORDIC processor comprising:

input means for receiving input information and input control signals;

output means for providing output information and output control signals;

first, second, and third arithmetic units coupled in parallel for performing floating point CORDIC calculations, each of said first, second, and third arithmetic units including an adder-subtractor, a data register, and a fixed plural bit shifting unit;

coupling means for selectively intercoupling the adder-subtractors, the data registers, and the fixed plural bit shifting units of the first, second, and third arithmetic units;

storage means for storing a plurality of floating point CORDIC routines and a plurality of tables of uniquely determined floating point CORDIC constants; and

control means coupled to the first, second, and third arithmetic units, to the storage means, and to the coupling means, said control means being responsive to the input control signals and to the input information for selecting different ones of the floating point CORDIC routines and associated floating point CORDIC constants stored in the storage means and for selectively enabling different portions of the coupling means.

2. A floating point CORDIC processor as in claim 1 wherein:

said tables of uniquely determined floating point CORDIC constants stored in the storage means include tables of plural-bit rotation and distortion constants for use in performing trigonometric and hyperbolic floating point CORDIC calculations;

said control means includes first logic means coupled to the first, second, and third arithmetic units for

automatically reselecting a plural-bit rotation or distortion constant when, within the accuracy of the floating point CORDIC processor, the bits of that constant are identical to the bits of the next plural-bit rotation or distortion constant to be selected; and

said control means includes second logic means coupled to the first, second, and third arithmetic units for automatically reselecting a prescribed set of plural-bit distortion constants for converging hyperbolic floating point CORDIC rountines.

**3.** A floating point CORDIC processor as in claim **1** wherein said coupling means comprises:

first coupling means for intercoupling the adder-subtractor of the first arithmetic unit with the data register and the fixed plural-bit shifting unit of the first arithmetic unit, with the fixed plural-bit shifting unit of the second arithmetic unit, with the adder-subtractor and the fixed plural-bit shifting unit of the third arithmetic unit for transmitting information and control signals therebetween;

second coupling means for intercoupling the adder-subtractor of the second arithmetic unit with the adder-subtractor and the fixed plural-bit shifting unit of the first arithmetic unit, with the data register and the fixed plural-bit shifting unit of the second arithmetic means, and with the adder-subtractor of the third arithmetic unit for transmitting information and control signals therebetween; and

third coupling means for intercoupling the adder-subtractor of the third arithmetic unit with the data register and the fixed plural bit shifting unit of the third arithmetic unit for transmitting information and control signals therebetween.

**4.** Apparatus for shifting in parallel a fixed plural number of consecutive bits beginning with any bit position from an ordered set of bits, said apparatus comprising:

first logic means for defining a first plurality of overlapping groups of consecutive bits from the ordered set of bits, each of these groups comprising a predetermined number of consecutive bits beginning with a different bit position in the ordered set of bits, the number of bits in each of these groups being less than the number of bits in the ordered set of bits and being greater than the fixed plural number of consecutive bits to be shifted in parallel from the ordered set of bits;

first decoding means coupled to the first logic means for selecting any one of the first plurality of overlapping groups of consecutive bits from the ordered set of bits in response to an input control signal;

second logic means coupled to the first logic means for defining a second plurality of overlapping groups of consecutive bits from the group of consecutive bits selected by the first decoding means, each of these groups comprising a predetermined number of consecutive bits beginning with a different bit position in the group of consecutive bits selected by the first decoding means, the number of bits in each of these groups being less than the number of bits in the group of consecutive bits selected by the first decoding means and being equal to the fixed plural number of consecutive bits to be shifted in parallel from the ordered set of bits;

second decoding means coupled to the second logic means for selecting any one of the second plurality of overlapping groups of consecutive bits in response to an input control signal; and

output means coupled to the second logic means for out-putting in parallel the group of consecutive bits selected by the second decoding means.

**5.** Apparatus for shifting in parallel a fixed plural number of consecutive bits beginning with any bit position from an ordered set of bits, said apparatus comprising:

logic means for defining a plurality of overlapping groups of consecutive bits from the ordered set of bits, each of these groups comprising a predetermined number of consecutive bits beginning with a different bit position in the ordered set of bits, the number of bits in each of these groups being less than the number of bits in the ordered set of bits and being equal to the fixed plural number of consecutive bits to be shifted in parallel from the ordered set of bits;

decoding means coupled to the logic means for selecting any one of the plurality of overlapping groups of consecutive bits from the ordered set of bits in response to an input control signal; and

output means coupled to the logic means for outputting in parallel the group of consecutive bits selected by the decoding means.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

# CERTIFICATE OF CORRECTION

PATENT NO. : 3,766,370

DATED : October 16, 1973

INVENTOR(S) : John S. Walther

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

Column 1, lines 27-28, "the present CORDIC processor" should read -- it --;

Column 2, lines 36-37, "CORDIC floating point" should read -- floating point CORDIC --; line 40, "CORDIC floating point" should read -- floating point CORDIC --; line 44, "CORDIC floating point" should read -- floating point CORDIC --; lines 51-52, "CORDIC floating point" should read -- floating point CORDIC --;

Column 3, line 3, "D-shifter" should read -- A-adder of FIGS. 9A-D --; line 11, "multiplexor" should read -- transfer gates --; line 12, "shifter" should read -- adder --; line 23, "multiplexor" should read -- transfer gates --; line 24, "D-shifter" should read -- C-adder --; line 27, "CORDIC floating point" should read -- floating point CORDIC --; line 31, "CORDIC floating point" should read -- floating point CORDIC --; line 35, "CORDIC floating point" should read -- floating point CORDIC --; line 39, "CORDIC floating point" should read -- floating point CORDIC --; line 42, "CORDIC floating point" should read -- floating point CORDIC --; lines 44-45, "entier/fix/load/-load I/" should read -- entier, fix, load, load I, and --; line 45, "CORDIC floating point" should read -- floating point CORDIC --; lines 48-49, "addition/subtraction/ absolute/" should read -- addition, subtraction, absolute, and --; lines 49-50, "CORDIC floating point" should read -- floating point CORDIC --; line 52, "overflow/" should read -- overflow and --; line 53, "CORDIC floating point" should read -- floating point CORDIC --; line 56, "multiply/" should read -- multiply and --; line 57, "CORDIC floating point" should read -- floating point CORDIC --; line 61, "CORDIC floating point" should read -- floating point CORDIC --;

UNITED STATES PATENT AND TRADEMARK OFFICE

# CERTIFICATE OF CORRECTION

PATENT NO. : 3,766,370

DATED : October 16, 1973

INVENTOR(S) : John S. Walther

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

lines 64-65, "sine/cosine/tangent/hyperbolic" should read -- sine, cosine, tangent, and hyperbolic --; lines 65-66, "CORDIC floating point" should read -- floating point CORDIC --;

Column 4, line 2, "CORDIC floating point" should read -- floating point CORDIC --; line 5, "artangent" should read -- arctangent --; line 6, "CORDIC floating point" should read -- floating point CORDIC --; line 10, "CORDIC floating point" should read -- floating point CORDIC --; line 12, "A-C" should read -- A-D --; lines 13-14, "sine/hyperbolic" should read -- sine and hyperbolic --; lines 16-17, "log/hyperbolic arctangent/" should read -- log, hyperbolic arctangent and --; line 18, "CORDIC floating point" should read -- floating point CORDIC --; line 21, "CORDIC floating point" should read -- floating point CORDIC --; line 25, "CORDIC floating point" should read -- floating point CORDIC --; line 29, "CORDIC floating point" should read -- floating point CORDIC --; line 31, "CORDIC floating point" should read -- floating point CORDIC --; line 34, "CORDIC floating point" should read -- floating point CORDIC --; line 37, "shifter" should read -- shifters --; line 38, "CORDIC floating point" should read -- floating point CORDIC --; line 40, "CORDIC floating point" should read -- floating point CORDIC --; line 42, "CORDIC floating point" should read -- floating point CORDIC --;

Column 5, line 56, after "angle" insert -- and --;

Column 6, at approximately line 26, in Equation (A4), "(is)" should read -- (iz) --; at approximately line 28, in Equation (A5), "$\tanh^{-1}z$" should read -- $\tanh z^{-1}$ --; at approximately line 59, in Equation (15), "$Y_n$" should read -- $y_n$ --;

UNITED STATES PATENT AND TRADEMARK OFFICE

# CERTIFICATE OF CORRECTION

PATENT NO. : 3,766,370

DATED : October 16, 1973

INVENTOR(S) : John S. Walther

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

Column 8, line 20, "hwere" should read -- where --;

Column 11, at approximately line 38, in Equation (45), "$\alpha_{m,F}$'" should read -- $\alpha'_{m,F}$ --;

Column 12, line 1, "$60_{0.F}$'" should read -- $\alpha'_{0,F}$ --;

Column 15, at approximately line 50, "exponenet" should read -- exponent --;

Column 16, at approximately line 13, line 3 of Table 7, "Binary" should read -- (Binary --;

Column 17, line 26, after "tion." delete "pl"; at approximately line 38, after "if the" (second occurrence) insert -- original was +127). --;

Column 18, line 55, "rota-tions" should read -- rotations --;

Column 24, in line 1 of Table 11, the date "6/16/1970" should read -- 4/17/1970 --;

Column 26, line 22, "S40( 8) 6 2" should read -- S40( 8) 6 12 --;

Column 34, line 46, "TRU( Ø) 1 1Ø" should read -- TRU( Ø) 19 1Ø --;

Column 44, line 68, after "(decimal)" delete "enable";

UNITED STATES PATENT AND TRADEMARK OFFICE

# CERTIFICATE OF CORRECTION

PATENT NO. : 3,766,370

DATED : October 16, 1973

INVENTOR(S) : John S. Walther

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

Column 46, line 40, "0" should read -- (0) --;

Column 50, line 60, "and ENC" should read -- an ENC --;

Column 63, at approximately line 62, "plural bit" should read -- plural-bit --; at approximately lines 64-65, "plural bit" should read -- plural-bit --.

Signed and Sealed this

*fourth* Day of *May 1976*

[SEAL]

*Attest:*

**RUTH C. MASON**
*Attesting Officer*

**C. MARSHALL DANN**
*Commissioner of Patents and Trademarks*