THEORY OF OPERATION

INTRODUCTION

The reader is assumed to be acquainted with general digital theory for computers. It is also assumed that the reader is familiar with the programming material. He need not be a programmer but he should understand what the individual instructions accomplish.

The broad concepts and the detailed schematics of the KENBAK-1 computer are given. The material is divided into these sections:

Glossary of Signals

Block Diagram

State Diagram

Definition of Circuit Elements

Kenbak-1 Computer Circuit Schematics

The Glossary of Signals is presented first to emphasize that it does exist and that it can be a useful working tool in studying other aspects of the presentation. However, it is a reference section and not a fundamental mode for presenting ideas. The local reference in the Glossary is the schematic page number where that signal originates.

GLOSSARY

Name	Local Reference	Definition or Comment
A	(20)	One of the two inputs to the adder/subtractor, see also B.
A7	(03)	The most significant address bit which selects one of the two delay lines which constitute the memory.
B	(20)	One of the two inputs to the adder/subtractor. During subtraction B is subtracted from A.
BD	(21)	A flipflop used to detect whether any bits in the A, B, or X registers are 1. Used in evaluating JC.
BM	(23)	An AND of the conditions 1. Skip or Set instructions, 2. The bit time (BT) of the selected bit, and 3. State SL. In the state diagram, BM is used in broad sense of 1 to mean Bit Manipulation instructions.
BT	(18)	The bit manipulation instructions designate a specific bit time. BT is true during that time.
BU	(05)	The eight data input switches are scanned by T0 through T7 and 0Red together to produce the serial signal BU.
C	(19)	The carry flipflop in the adder.
CL	(04)	True when the Clear pushbutton is depressed.
CM	(02)	The comparison between the memory address register and the desired address is true. CM is only valid during T7.
$\overline{CP}$	(01)	Major system clock pulse. A bit time starts with the negative going transition of $\overline{CP}$.
CR	(19)	Carry to the next bit before being delayed by C.

GLOSSARY

CT	(25)	Output of multivibrator.
CY	(21)	A flipflop which holds the carry from the previous byte.
$\overline{C7}$	(01)	Clock pulse that occurs only during T7. Has same phase as $\overline{CP}$.
$\overline{DA}$	(04)	Display Address (from pushbutton of that name). Goes to ground when button is pushed.
$\overline{DD}$	(04)	Display Data (from Read Memory pushbutton) after being cleaned up. Goes to ground when button is pushed.
DEX		Index address mode.
DLO	(24)	Output of memory delay line 0.
DL1	(24)	Output of memory delay line 1.
$\overline{EA}$	(04)	Enter Address (from Set Address pushbutton) after being cleaned up. Goes to ground when button is pushed.
ED	(22)	Automatic processing should End after the current instruction is finished.
$\overline{EN}$	(04)	Enter Data (from Store Memory pushbutton) after being cleaned up. Goes to ground when button is pushed.
F1/F2	(01)	Inputs to delay line clock drivers.
$\overline{GO}$	(04)	Signal from Start Pushbutton after being cleaned up. Goes to ground when button is pushed.
$\overline{HT}$	(04)	Signal from Stop Pushbutton. Goes to ground when button is pushed.
IMMED		Immediate or constant address mode.
IN	(04)	True during byte time when location 377 (Input) is being read from memory.
IND		Indirect address mode.

GLOSSARY

I0/I7	(14)	Generally the I register is the instruction register. I7 is the most significant bit and I0 is the least significant bit. As an instruction register, I holds the first byte of an instruction.
J		Any Jump instruction. See also JD, JI, JM, JP.
JC	(21)	Jump Conditions are true; the Jump should be made.
JD		A Jump instruction with direct addressing (not indirect).
JDLO	(24)	Input to delay line 0.
JDL1	(24)	Input to delay line 1.
JI		A Jump instruction with indirect addressing.
JM		A Jump and Mark instruction.
JP		A straight Jump instruction (no Mark).
JSO/JS4	(10)	Inputs to 5 state flipflops.
$\overline{KP}$	(22)	This signal is ground if a skip is required in the bit test instructions.
$\overline{KS}$	(06)	Shift control to K register. At ground, $\overline{KS}$ indicates a shift.
K0/K7	(07)	The K Register controls the data lamps. K7 is the most significant bit.
LC	(02)	Carry flipflop for memory address (L) register. When LC is a 1 at T7 the next address is 177 (and 377).
$\overline{LS}$	(18)	Shift control signal for left shifting.
L0/L7	(02)	Memory address register. Read serially via L0.

MEM		Memory address mode.
MR	(03)	Memory read. Is the output of DLO or DL1 as selected by A7.
OF	(19)	Overflow as detected by the adder.
OV	(21)	Overflow delayed for one byte time.
PF	(22)	Four is to be added to the P register (Plus Four).
PO	(22)	One is to be added to the P register (Plus One).
PT	(22)	Two is to be added to the P register (Plus Two).
QB	(09)	State to wait for Start button to be released.
QC	(09)	Idle state for manual operations.
QD	(09)	State to seek address comparison for manual operations.
QE	(09)	State to execute manual operations.
QF	(09)	State to wait for control buttons to be released.
$\overline{R}$	(03)	Desired address in logic complement form.
RR	(18)	Control signal for right shifts and rotates.
SA	(09)	State to seek P register.
SB	(09)	State to read and update P register.
SC	(09)	State to seek instruction.
SD	(09)	State to read first byte of instruction.
SE	(09)	State to read second byte of instruction.
SF	(09)	State to seek indirect address.
SG	(09)	State to read indirect address.
SH	(09)	State to seek X register.

SJ	(09)	State to do index address computation.
SK	(09)	State to seek operand.
SL	(09)	State to read operand and execute bit manipulation instructions.
SM	(09)	State to seek A, B, or X register.
SN	(09)	State to execute change to A, B, or X (also P) register.
SP	(09)	State to load I with A, B, or X register.
SQ	(09)	State to modify P in Jump and Mark instructions.
SR	(09)	State to seek address for storing in memory.
SS	(09)	State to store data in memory.
ST	(09)	State to seek P register.
SU	(09)	State to seek A or B register.
SUM	(19)	Output of adder.
SV	(09)	State to read A or B register.
SW	(09)	State to perform shift/rotates in W register.
SX	(09)	State to seek A or B register.
SY	(09)	State to store W register in A or B register.
SZ	(09)	State to evaluate jump conditions.
S0/S5	(09)	Five flipflop state register.
T0/T7	(01)	Bit time counter. T0 is true during first bit time of a byte.
TM		A transfer to memory, i.e., a Store A, B, or X instruction.
TS	(21)	True for the byte time after SN in addition and subtraction. Controls writing of OV and CY.

GLOSSARY

TX	(18)	Intermediate signal used in shifts, rotates, and bit manipulation instructions.
UT	(04)	True during time location 200 (Output) is being read from memory.
WD	(23)	Write data. Data to be written in the memory.
WT	(23)	Write time. Time to write in the memory.
W0/W7	(17)	W register. W0 is least significant bit. A general utility register.

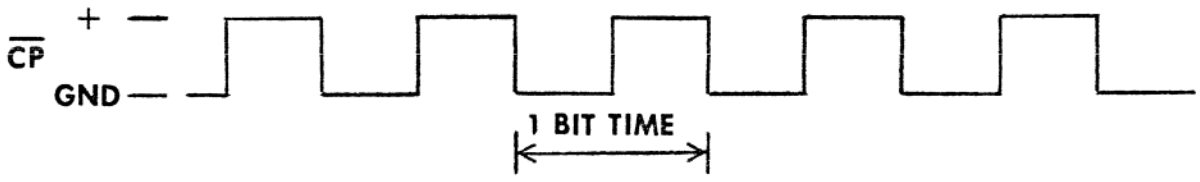
BLOCK DIAGRAM

The major logical elements of the system are shown in the block diagram on page ____. The KENBAK-1 computer differs internally in its structure from the majority of computers in these two ways:

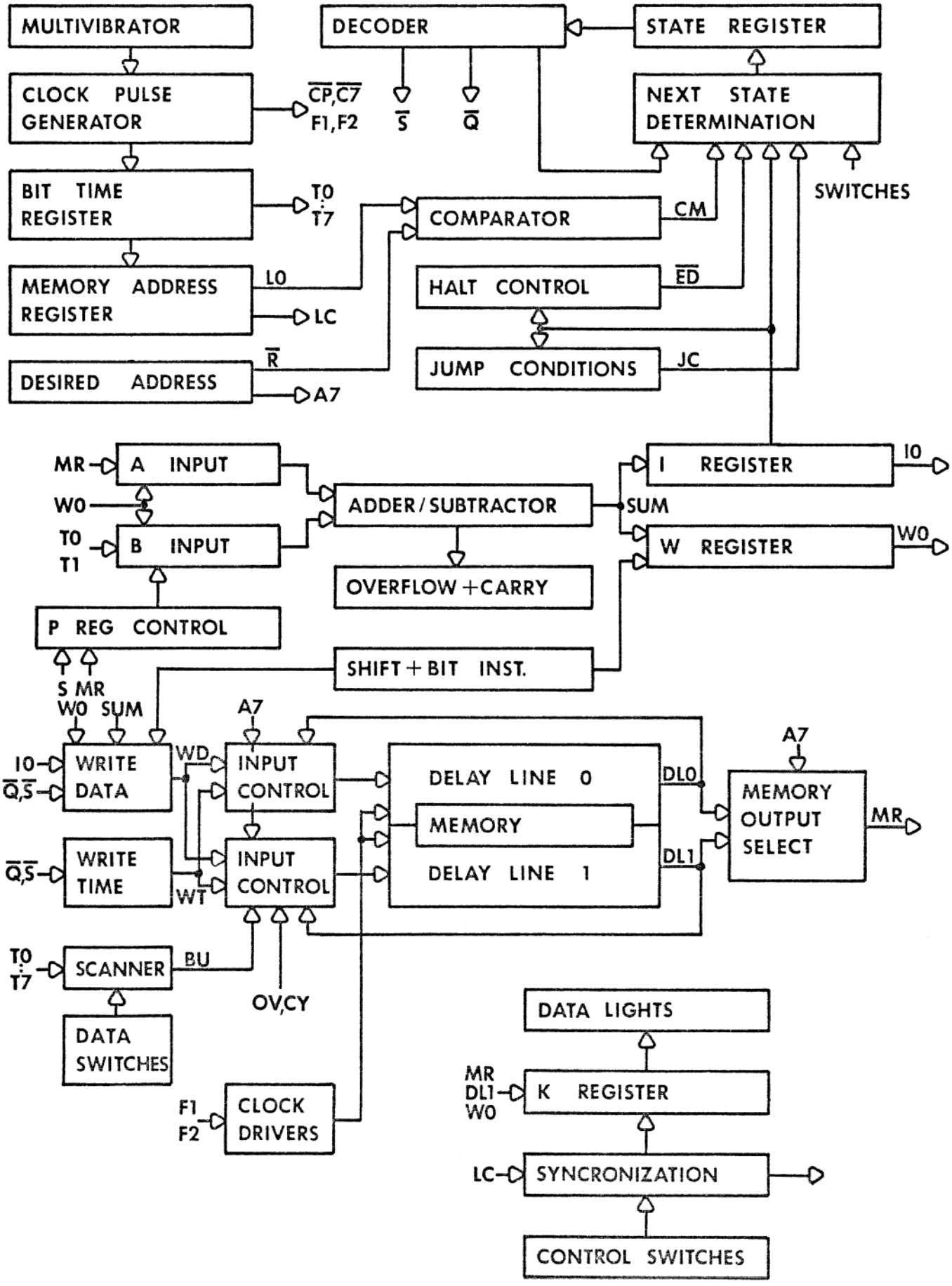
1. The computer is serial. Data is read, or written, from the memory one bit at a time. Thus it takes eight clock times to read a byte from the memory.

2. The programming registers (A, B, X and P) do not exist as actual hardware registers. Instead, a fewer number of hardware registers are time-shared to provide the necessary data manipulation.

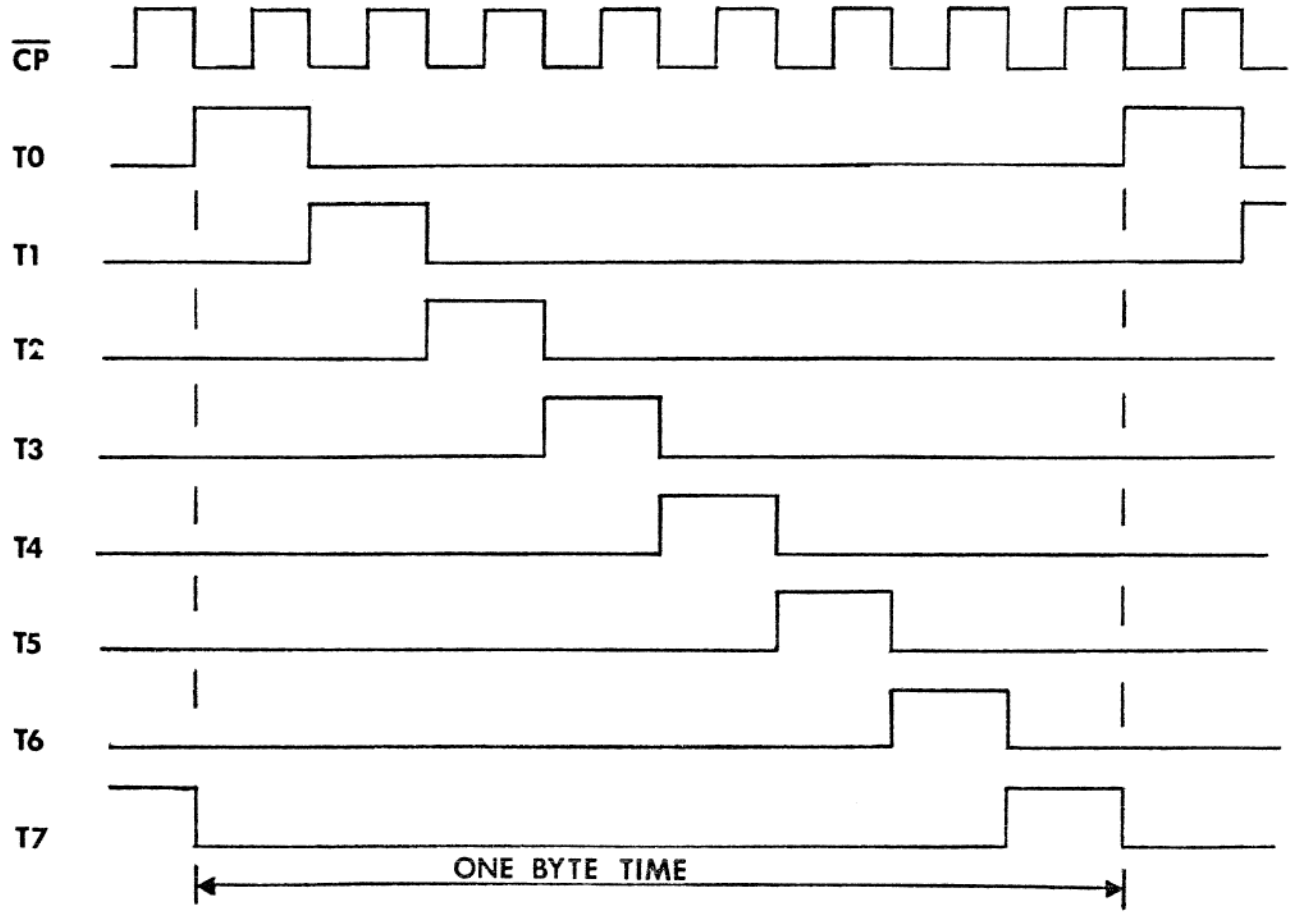
In the block diagram, the source of the clock signals is an RC multivibrator which runs at twice the fundamental system frequency. The clock pulse generator produces the basic system clock, $\overline{CP}$, which is a square wave. One bit time lasts from one positive-to-ground transition to the next such transition:



There are eight bit times: T0, T1, T2, T3, T4, T5, T6, and T7. These are generated by the bit time register in which a single bit shifts. Following T7, T0 is generated

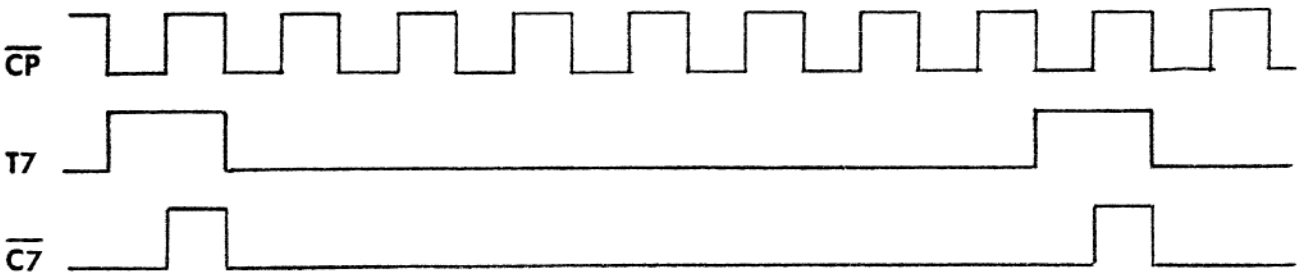

MULTIVIBRATOR
CLOCK PULSE GENERATOR
BIT TIME REGISTER
MEMORY ADDRESS REGISTER
DESIRED ADDRESS
DECODER
STATE REGISTER
NEXT STATE DETERMINATION
SWITCHES
COMPARATOR
HALT CONTROL
JUMP CONDITIONS
A INPUT
B INPUT
ADDER / SUBTRACTOR
OVERFLOW + CARRY
I REGISTER
W REGISTER
P REG CONTROL
SHIFT + BIT INST.
WRITE DATA
WRITE TIME
INPUT CONTROL
INPUT CONTROL
DELAY LINE 0
MEMORY
DELAY LINE 1
MEMORY OUTPUT SELECT
SCANNER
DATA SWITCHES
CLOCK DRIVERS
DATA LIGHTS
K REGISTER
SYNCRONIZATION
CONTROL SWITCHES

again and the cycle repeats indefinitely.

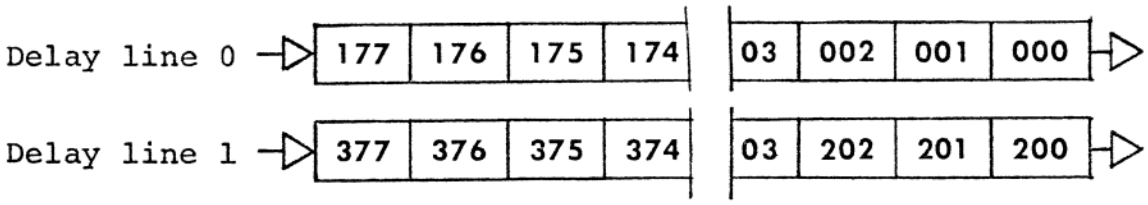


A byte time lasts eight bit times, from the start of T0 to the next start of T0.

It is useful to have a clock pulse which only occurs during T7 bit time. This is $\overline{C7}$:



The memory for the computer consists of two delay lines or shift registers, each of 1,024 bits. Therefore each will hold 128 bytes of information. Bytes with addresses 000 to 177 (octal) are in delay line 0 and bytes with addresses 200 to 377 (octal) are in delay line 1.



After byte 177 is read, byte 000 will appear in delay line 0. At the same time that byte 000 is read from delay line 1, byte 200 is read from delay line 2. Circulation of the information within the delay lines in continuous. It never stops.

Eight bits are required to address a byte since $2^8 = 256$. The most significant address bit, A7, selects the delay line. The seven least significant address bits select a position or a time slot within the delay lines.

The memory address register (the L register) shifts serially and counts one each byte time. The counting, or adding one, is a serial process.

Whenever reading or writing is to occur, the desired address is set up as the serial signal R. When the seven least significant bits of R are equal to the seven least significant bits of the memory address register, then the

next byte time is when the data for the desired address will be at the input and output of the memory. The most significant bit of R is used to determine A7, the delay line selection signal. The memory address register is, in effect, one byte time ahead of the memory.

The comparison between the memory address register and R is summarized by the signal CM during T7. Comparison exists if CM is true at this time. Address comparison does not exist if CM is false at this time.

As an instruction is executed, the computer proceeds through a series of steps. For example, the computer has to find address 003 (the P register), then it has to read and update the value of P, then it has to find the instruction, read the instruction, etc. While it is doing these things, the computer is in certain states. The fact that the computer is in a given state defines what it is to do while it is in that state. The data read from the memory and other conditions including the current state of the computer defines the next state. Thus the computer advances from state to state.

The computer has 29 valid states and 3 invalid states. Should an invalid state occur, which it shouldn't except when power is turned on, a valid state is generated next. The 32 total states result from the combinations of five flipflops, S4, S3, S2, S1, S0. The five signals are applied to decoders which generate 32 outputs. Only one of these

outputs can be true at any given time. This is the current state of the computer.

A state always starts and ends with the end of T7 (or the beginning of T0). It may last one byte time or it may last for many byte times.

The names for the decoded state outputs are two letter symbols which start with S or Q. No signal other than a state signal is assigned a two letter code starting with S or Q.

The nature of the decoder is such that the single true output is at ground while all of the rest of the decoded state signals are positive. Hence the outputs of the decoder are labelled $\overline{Si}$ and $\overline{Qi}$ to conform to positive logic.

The next state of the computer and when it advances are determined by many factors:

1. The current state,
2. Has the desired address been found,
3. Is a halt required,
4. The instruction and its addressing mode,
5. Are the jump conditions met,
6. The panel or control switches.

There is a serial adder/subtractor with two inputs, A and B. In subtraction, B is subtracted from A. Overflow and carry out of a byte are detected, and in addition and subtraction instructions, used to update the OF and CA bits

in locations 201, 202, and 203.

Most of the time, the I register is used as the instruction register to hold the first byte of the instruction. It is also used to hold the data for storing into the memory and for manual console operations. Except when information is being transferred in or out serially, the I register does not shift.

The W register is a general purpose working register which holds data, addresses and operands. It will be discussed at length later. Shift operations, for the Shift and Rotate instructions, are performed in it. Except for some bit times during these instructions, the W register is always shifting.

The memory (two delay lines) have their outputs connected to their inputs so that data is retained or held. New information is entered in different ways, but the usual way is through the use of the Write Data and Write Time signals. New data, equal to Write Data, is entered into the delay line specified by the A7 address bit when Write Time is true.

The input control for delay line 1 allows some special information to be entered into it alone. The states of the eight data switches are converted to a serial signal, BU. This is entered into address 377. The Clear pushbutton will erase the contents of address 377 to all zeroes. The overflow

and carry bits are entered into location 201, 202, 203 by direct entry through delay line 1 input control.

The actual delay lines used are MOS shift registers which require two clock signals out of phase with each other. One occurs during the even bit times and the other occurs during the odd bit times.

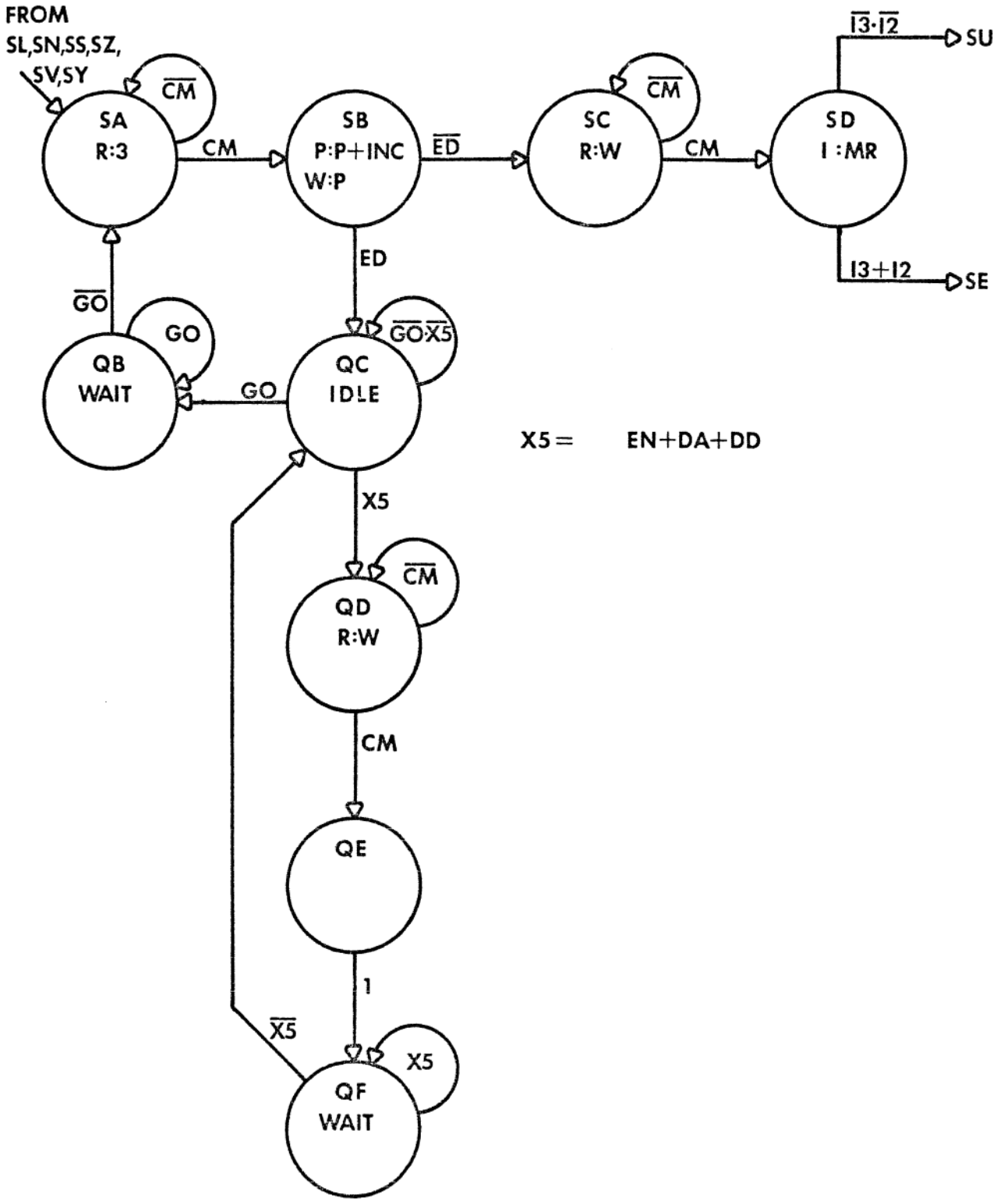
Several of the front panel control switches require some "cleaning up" before they can be used. After the contact bounce is eliminated they are used in the next state determination to control the reading and entry of data, the display of data, and starting and stopping of the computer. The data to be displayed in the lights may originate in the W register or in the memory.

STATE DIAGRAM

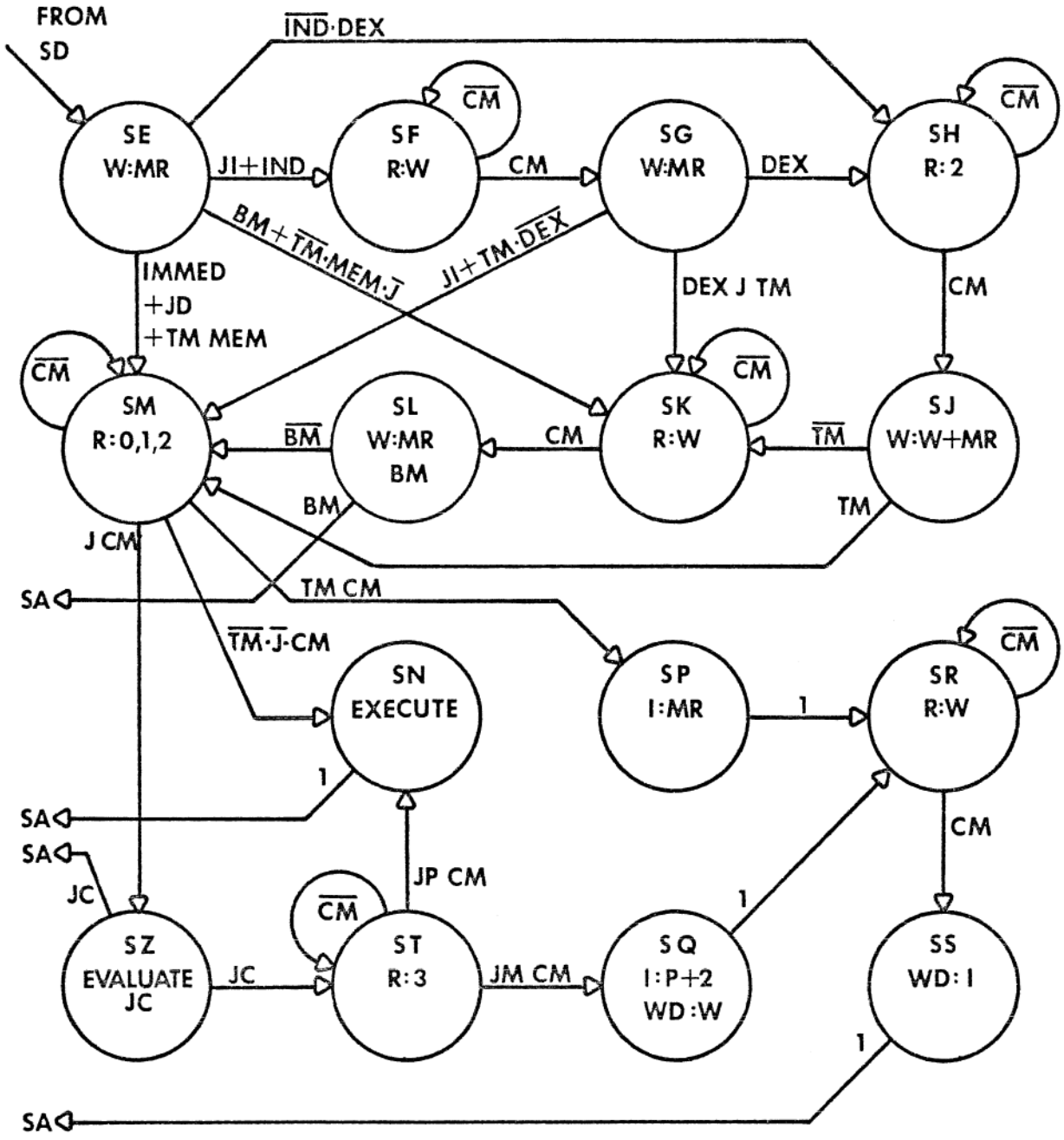
The state diagram for the computer is presented on pages 25, 26, and 27. State names which start with S pertain to automatic operations of the computer or the Run mode. State names which pertain to manual operations start with the letter Q. Of the three pages of state diagrams, the first page pertains to manual operations and automatic operations which are common to one and two byte instructions. The second page involves only two byte instructions and the third page is for one byte instructions.

A state lasts for one or more byte times. It always starts with the beginning of a T0 and it always ends with the end of T7. A state is represented in these diagrams as a circle. Directed lines, the state transitions, interconnect the states. Each line is labelled with logical conditions under which the transition occurs. All of the lines which leave a state must have a logical sum which is equal to 1. Otherwise the computer might become trapped in a state with no valid exit. Also, if more than one line leaves a state, the logical product of any two must be 0. Otherwise the logical transition would not be clearly defined.

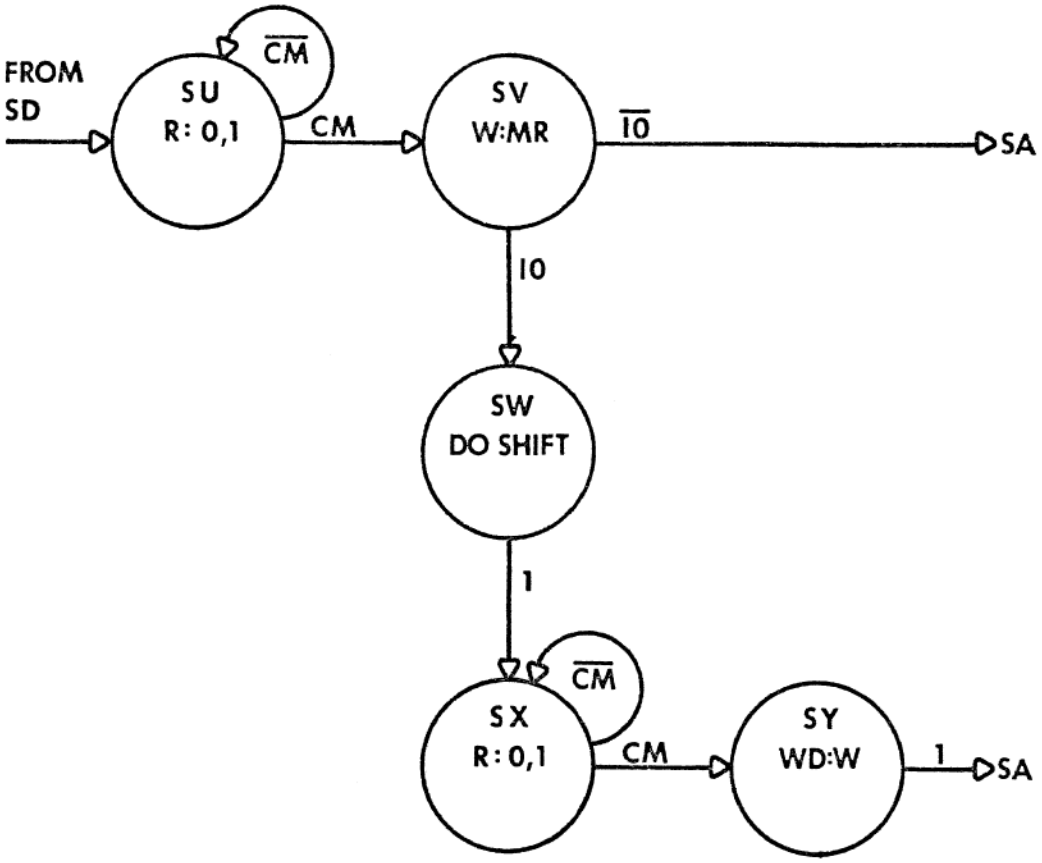
COMMON AND MANUAL OPERATIONS



TWO BYTE INSTRUCTIONS



ONE BYTE INSTRUCTIONS



SA

When the computer finishes any instruction, the SA state results. This may be regarded as the start of the next instruction. In state SA, the desired address is 3 which is the address of the P register. Therefore the function of state SA is to locate the P register. If, during T7, the comparison signal, CM, is false, state SA will be regenerated for another byte time. If the comparison signal is true, then the next state will be SB.

SB

State SB lasts one byte time. While it is true, the P register is being read from the memory. The value of the P register being read is the address of the instruction which was last completed. The P register increment control circuits, through signal PO, have a value which tells how many memory locations this last instruction occupied. This increment is added to the old contents to obtain the address of the next instruction. The sum is written back into the memory and is also transferred to the W register. At the conclusion of state SB, location 3 in the memory and the W register both contain the address of the next instruction.

If signal ED is true at the end of SB, this indicates the computer should stop, either because the last instruction was a Halt or because the Stop button was depressed.

Then the next state is QC. If the signal ED is false, the computer continues its automatic operations by going to state SC.

SC

The function of state SC is to find the next instruction. The desired address is in the W register and this is compared to the memory address register until a comparison is found. When CM is true at T7, the state register advances to state SD.

SD

During state SD, a one byte time state, the first byte of the instruction is transferred to the I register. The data path for this is through the A input of the adder. If I3 + I2 is true during T7, the instruction has two bytes. In this case the next state is SE. If I3 + I2 is not true, the instruction being read from the memory consists of only one byte and then the next state is SU. (The signal to use is I3 + I2, and not I2 + I1, because the instruction still has one bit position to shift in the I register. This will occur at the same time that SD ends.)

SE

During SE, which lasts one byte time, the second byte of the instruction is transferred to the W register. There is one exception to this rule. If the instruction is Store Constant (also called Store Immediate), the W register is

loaded with the address of the second byte of the instruction and not with the contents of the second byte. This is accomplished through the adder inputs.

	A	B	SUM
Store Immediate	WO	T0	Address of second byte
Not Store Immediate	MR	0	Contents of second byte

Since the W register contained the address of the first byte of the instruction, in the Store Immediate instructions the sum will be the address of the second byte. In all other cases the sum will be equal to the contents of the second byte. In either case it is the sum which is transferred to the W register.

Depending upon the addressing mode and the type of instructions, state SE may be followed by four other states. For indirect addressing, SF follows. For indexed addressing (without indirect addressing) SH is next. For direct memory addresses to find operands, state SK follows. If no operand is to be found, state SM follows.

Generally, regardless of the addressing mode, all of the two byte instructions will involve state SM which is a search for the A, B, or X register. If the instruction will change A, B, or X, then during SM the W register will contain the operand. If A, B, or X is to be stored in memory, then the W register will contain the address of

the location where A, B, or X is to be stored. If the instruction is a Jump, then W will contain the target address during SM.

The bit manipulation instructions are two bytes and have memory addressing, but they do not advance to state SM because they test or modify a location in the memory and not the A, B, or X registers.

SF

State SF is a search for an indirect address which is in the W register. When found, state SG results.

SG

During SG, the contents of the indirect address location are transferred to the W register. If indexing is required also, state SH follows. For a Jump Indirect or a Store without indexing, then SM is next. Otherwise state SK for finding the operand follows.

SH

In state SH a search for the X register, address 2, is made. When found, the state counter advances to SJ. State SH is preceeded by SE if there is indexing without indirect addressing. It is preceeded by SG if indirect addressing is also involved.

SJ

During SJ, a one byte time state, the contents of the X register are added to the contents of W. The result is

stored in W. For Store instructions, state SM follows. For other instructions, SK is the next state.

SK

On entering state SK, the W register contains the address of the operand. This may have been directly from the instruction or it may have been derived from indirect and indexing operations. Using this operand address, a search is made for the operand. When found, state SL follows.

SL

For the non-bit manipulation instructions, the W register is loaded with the operand. At the end of the byte time, for these instructions, state SM is next. For the Set 0 or Set 1 instructions, the designated bit is set during SL. For the Skip on 0 and Skip on 1 instructions, the P register increment control is set as necessary. The bit manipulations are then finished and state SA is the next state.

SM

State SM is a search for the A, B, or X register as defined by the instruction. (For Jump Unconditional instructions, a search is made for the A register.)

For instructions which modify the A, B, or X registers, the W register contains the operand. State SN follows.

For instructions which will store A, B, or X in the memory, the W register contains the address where the data is to be stored. State SP follows.

For the Jump instructions, the W register contains the target address. The target address is the jump address or the "mark" address. State SZ follows.

SN

The instructions which change A, B, or X do so during SN. The W register contains the operand. A, B, or X is read from the memory. The required operation is performed and the results are stored into A, B, or X. One byte time is required and then these instructions are finished. State SA is next.

In the Jump instructions (see SZ and ST), the address in W is transferred to the memory to become the new value of the P register.

SP

To store A, B, or X in the memory, the I register is loaded with the contents of A, B, or X. Since this destroys the instruction code in I, all of the remaining parts of the instruction can only be dependent upon the state. Upon leaving SP for state SR, the I register contains a byte to be stored in the memory and the W register contains the address where it is to be stored.

SQ (See after ST)

SR

In state SR the address corresponding to the contents of the W register is sought. When found, state SS is generated.

SS

In state SS the contents of the I register are transferred to the memory. This state is used in the Store instructions or in the Jump and Mark instructions. In all cases it concludes the instruction and state SA follows at the end of the byte time.

SZ (Out of sequence)

In the Jump instructions, after the A, B, or X register is located in state SM, state SZ follows. The jump conditions are evaluated during this byte time while A, B, or X is being read from the memory. If no jump is to be made, state SA follows SZ. If the jump is to be made, state ST follows.

ST

State ST is a search for the P register, address 3. In the Jump instructions without "marking", state SN follows after address 3 is found. For the Jump and Mark instructions, state SQ follows.

SQ (Out of sequence)

Two things occur during SQ. The I register is loaded with the value of the P register plus 2. This is the return address. At the same time the P register,

address 3, is set to the value of the target address in the W register. This target address is where the return address will be deposited. The P register increment control is set to add one so that the next instruction will be taken from the address following the target address. After state SQ, states SR, SS, and SA, all discussed previously, follow.

SU

(The five remaining S states pertain to one byte instructions.) If the instruction read in state SD is a one byte instruction, then state SU follows. A search is made for address 0 or 1 (A or B registers). When found, the state register advances to SV.

SV

In state SV, the A or B register is transferred to the W register. For the Halt and No-op instructions this state is the last and state SA follows.

SW

Shifts and rotates are executed in state SW which lasts one byte time. Following SW is state SX.

SX

A search for address 0 or 1 (A or B) is made. When found, the next state is SY.

SY

The shifted or rotated value of the original A or B register is written back into location 0 or 1. This finishes the Shift and Rotate instructions and the next state is SA.

SZ (See after SS)

QA (Not used)

QB (See after QF)

QC

When the computer is halted, state QC is the idle state. If one of the control switches (Store Memory, Display Address, or Read Memory) is pushed, the control advances to state QD.

During IN time of QC, the I register is always loaded with the contents of location 377. If the Set Address switch is pushed, generating EA, the contents of the I register are transferred to the W register.

If the Start button is pushed, control advances to state QB.

QD

In state QD the address which is equal to the contents of the W register is sought. When found, state QE is next.

QE

In state QE, the action depends on which button was pushed:

Store Memory	(EN)	:	Transfer I to Memory, Add 1 to W
Display Address	(DA)	:	Transfer W to K
Read Memory	(DD)	:	Transfer Memory to K, Add 1 to W

State QF follows after one byte time.

QF

The state register control waits at QF until the control buttons are released. When they are released, state QC is generated.

QB (Our of order)

State QB "remembers" that Start was depressed but prevents control from advancing until Start is released. When Start is released, state SA is generated.

DEFINITION OF LOGIC ELEMENTS

The logical circuits for the computer are implemented with 14 types of integrated circuits. Each type has a distinctive schematic symbol. In the following pages these symbols are defined.

An integrated circuit package has 14 leads (16 leads for the type 7442). Pin 7 is ground and pin 14 is +5 volts (for type 7442, pin 8 is ground and pin 16 is +5 volts). A package may contain a number of independent circuits. Each circuit is defined by its input and output pins and a typical input-output relationship is given as a logical expression.

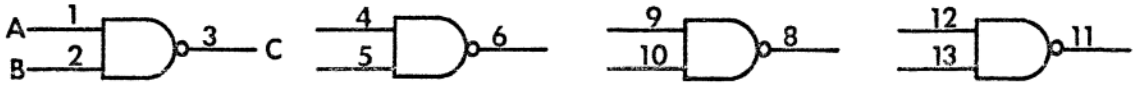
Additional information on these circuits can be found in manufacturers' data sheets or handbooks. Some manufacturers of these circuits include

Motorola Semiconductor
National Semiconductors
Texas Instruments
Signetics
Fairchild Semiconductor

though this is not an exhaustive list. Each manufacturer has a distinctive nomenclature for his units though they all generally involve the four digit 74-- number.

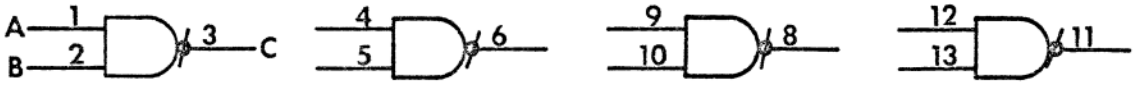
In the actual schematics, the input pins may be ordered differently from the definition.

QUAD 2—INPUT NAND GATE — 7400



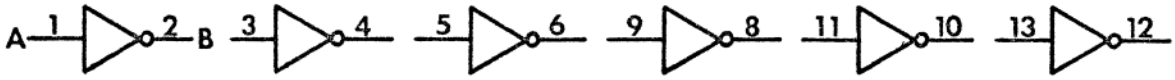
$C = \overline{A} + \overline{B}$

QUAD 2—INPUT NAND GATE, OPEN COLLECTOR — 7403



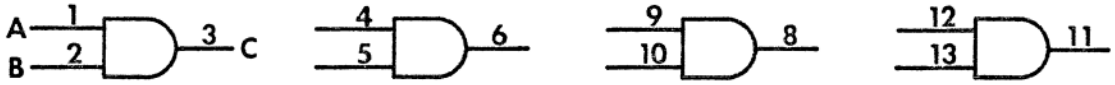
$C = \overline{A} + \overline{B}$ WITH EXTERNAL LOAD TO +5 V

HEX INVERTER — 7404



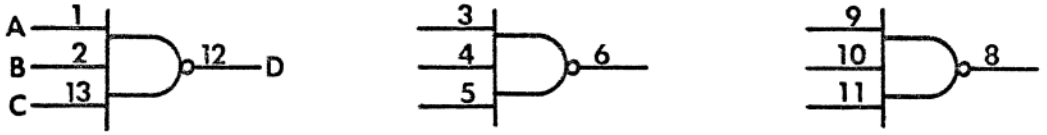
$B = \overline{A}$

QUAD 2—INPUT AND GATE — 7408



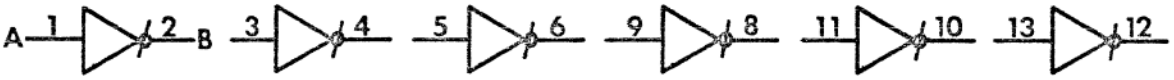
$C = A \cdot B$

TRIPLE 3—INPUT NAND GATE — 7410



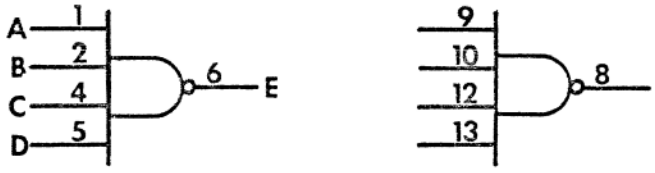
$D = \overline{A} + \overline{B} + \overline{C}$

HEX INVERTER BUFFER — 7416



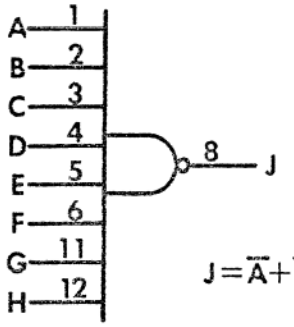
$B = \overline{A}$ WITH EXTERNAL LOAD TO +5 V

DUAL 4—INPUT NAND GATE — 7420



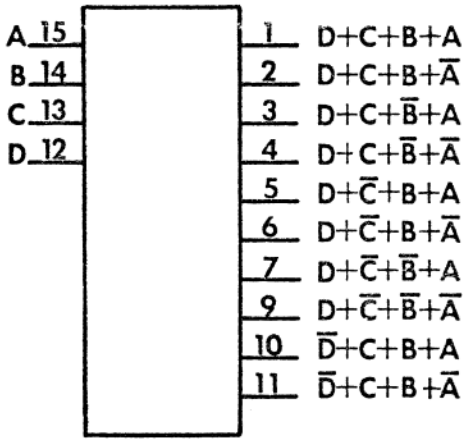
$E = \overline{A} + \overline{B} + \overline{C} + \overline{D}$

8—INPUT NAND GATE — 7430

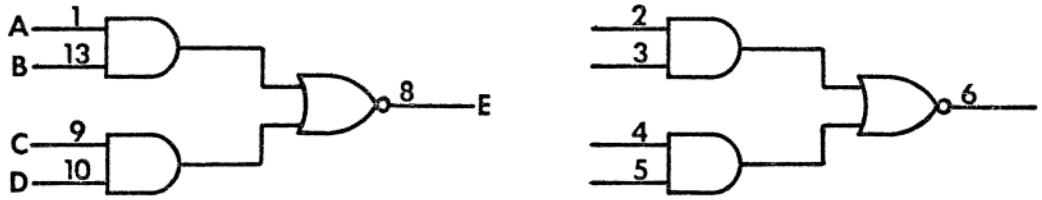


$J = \overline{A} + \overline{B} + \overline{C} + \overline{D} + \overline{E} + \overline{F} + \overline{G} + \overline{H}$

BCD TO DECIMAL DECODER — 7442

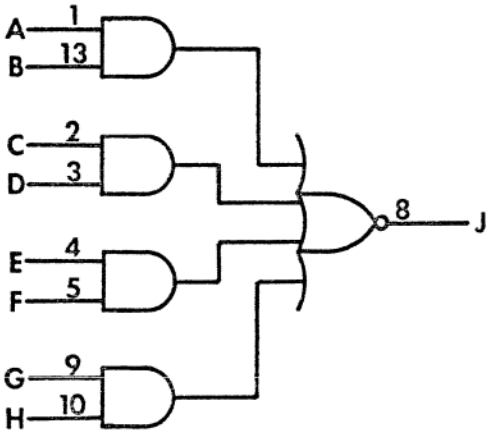


DUAL 2—WIDE 2—INPUT AND—OR—INVERT GATES — 7451



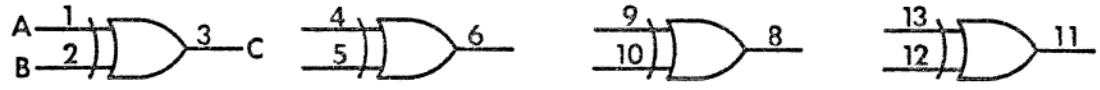
$E = \overline{A \cdot B + C \cdot D}$

4—WIDE 2—INPUT AND—OR—INVERT GATE — 7454



$J = \overline{A \cdot B + C \cdot D + E \cdot F + G \cdot H}$

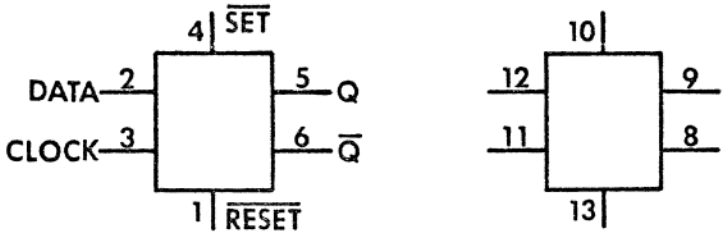
QUAD 2—INPUT EXCLUSIVE OR GATES — 7486, 3021



$C = A \cdot \overline{B} + \overline{A} \cdot B$

DUAL TYPE D FLIPFLOP

7474
7479

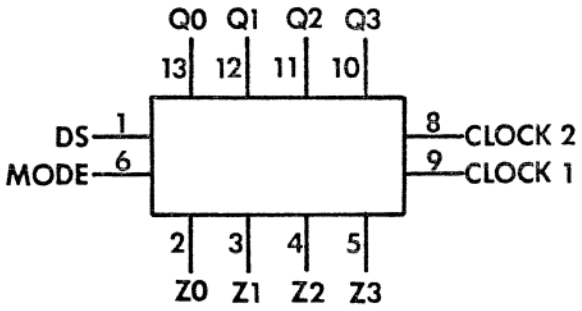


Q IS EQUAL TO DATA JUST PRIOR TO LAST 0 TO 1 CLOCK TRANSITION

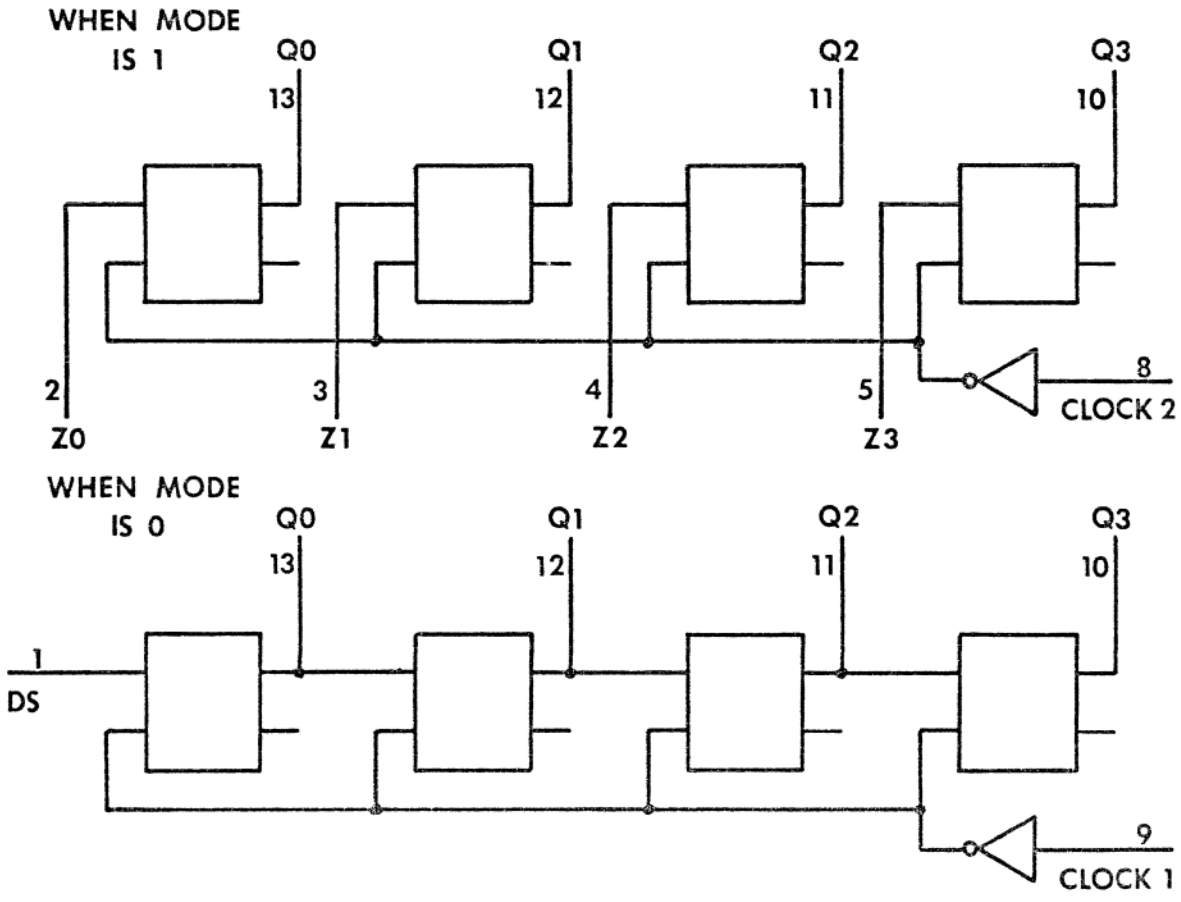
SET AND RESET ARE NOT USED

4 BIT SHIFT REGISTER

7495



EQUIVALENT CIRCUIT



LOGIC SCHEMATICS

In the following pages, the complete logic schematics for the computer are given. Each page is given a "local" two digit reference number which is used to cross reference the sheets. Some signals are so widely used that they are not cross referenced. These include

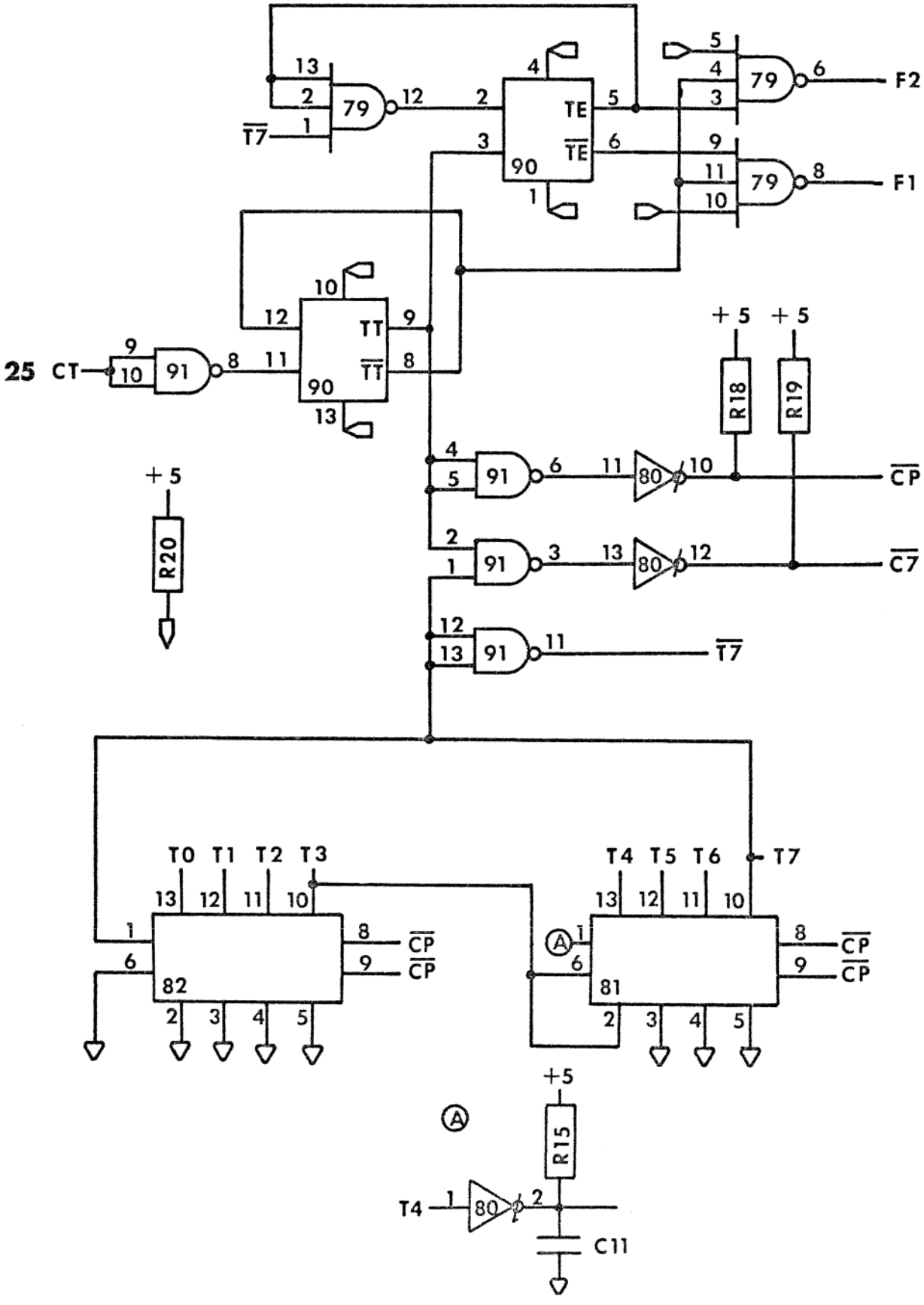
$\overline{CP}$, $\overline{C7}$, T0, T1, T2, T3, T4, T5, T6, T7 (sheet 01)

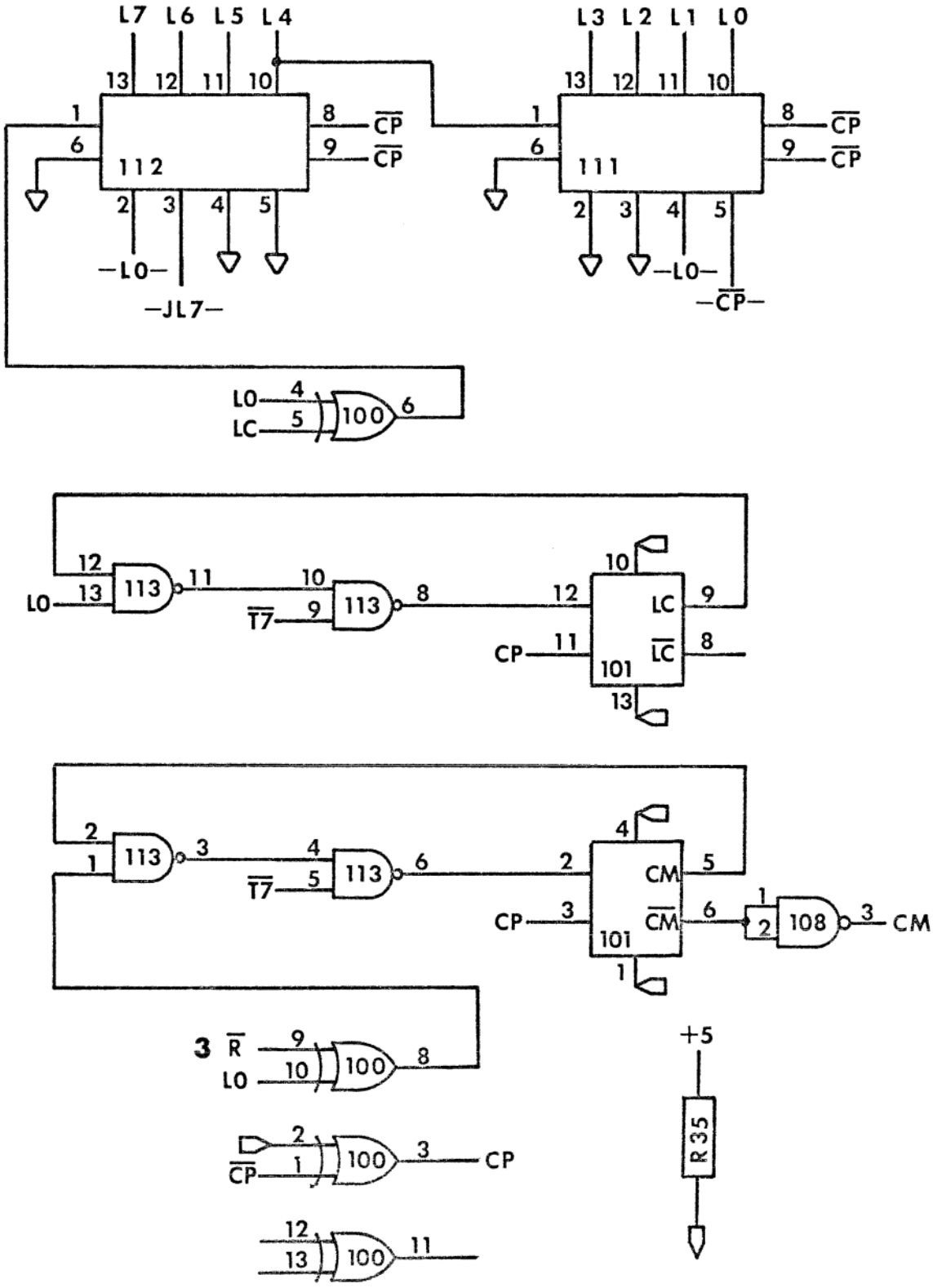
Original state signals, $\overline{Si}$ and $\overline{Qi}$ (sheet 09)

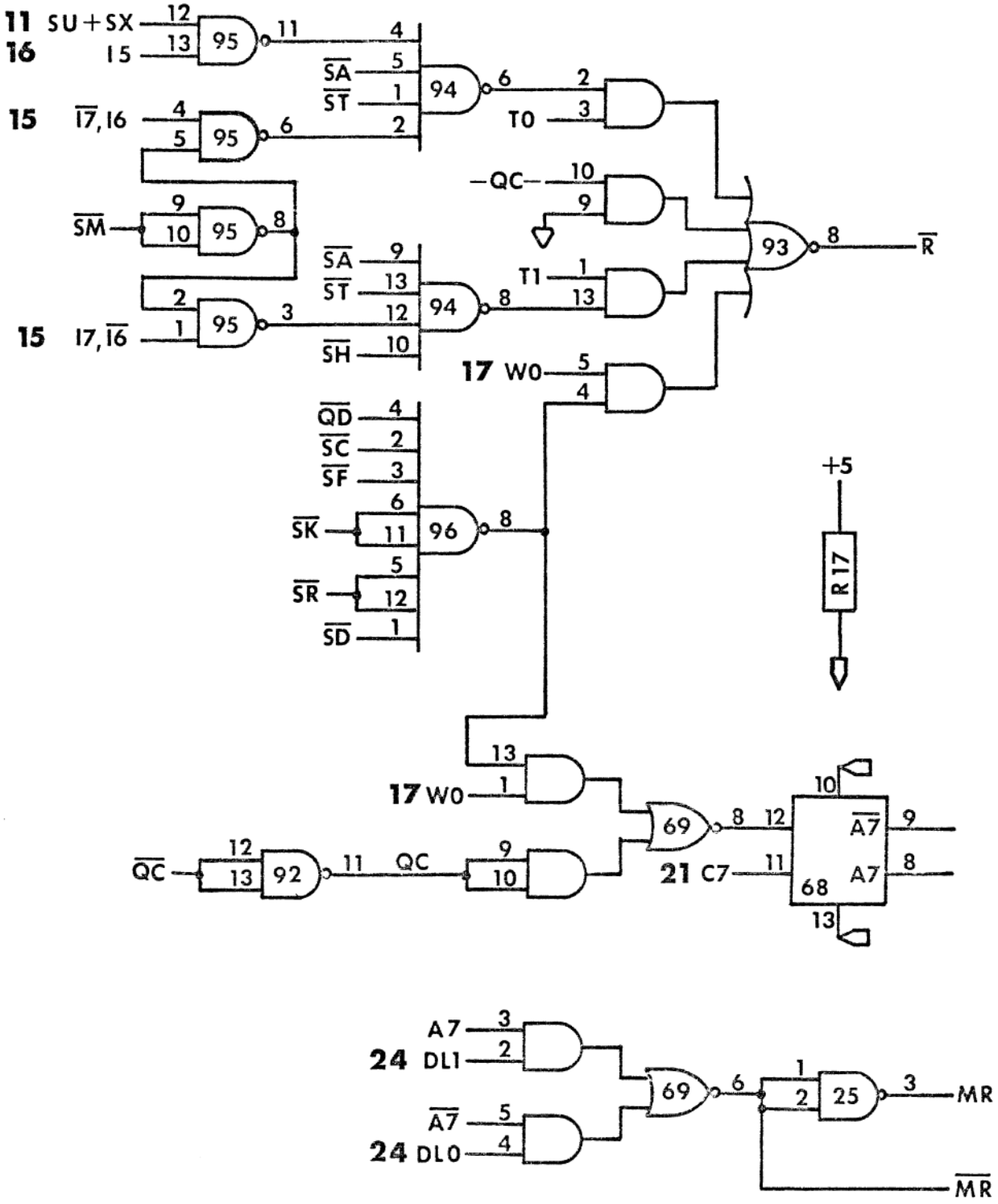
Original I7, I6, I5, I4, I3, I2, I1, I0 (sheet 14)

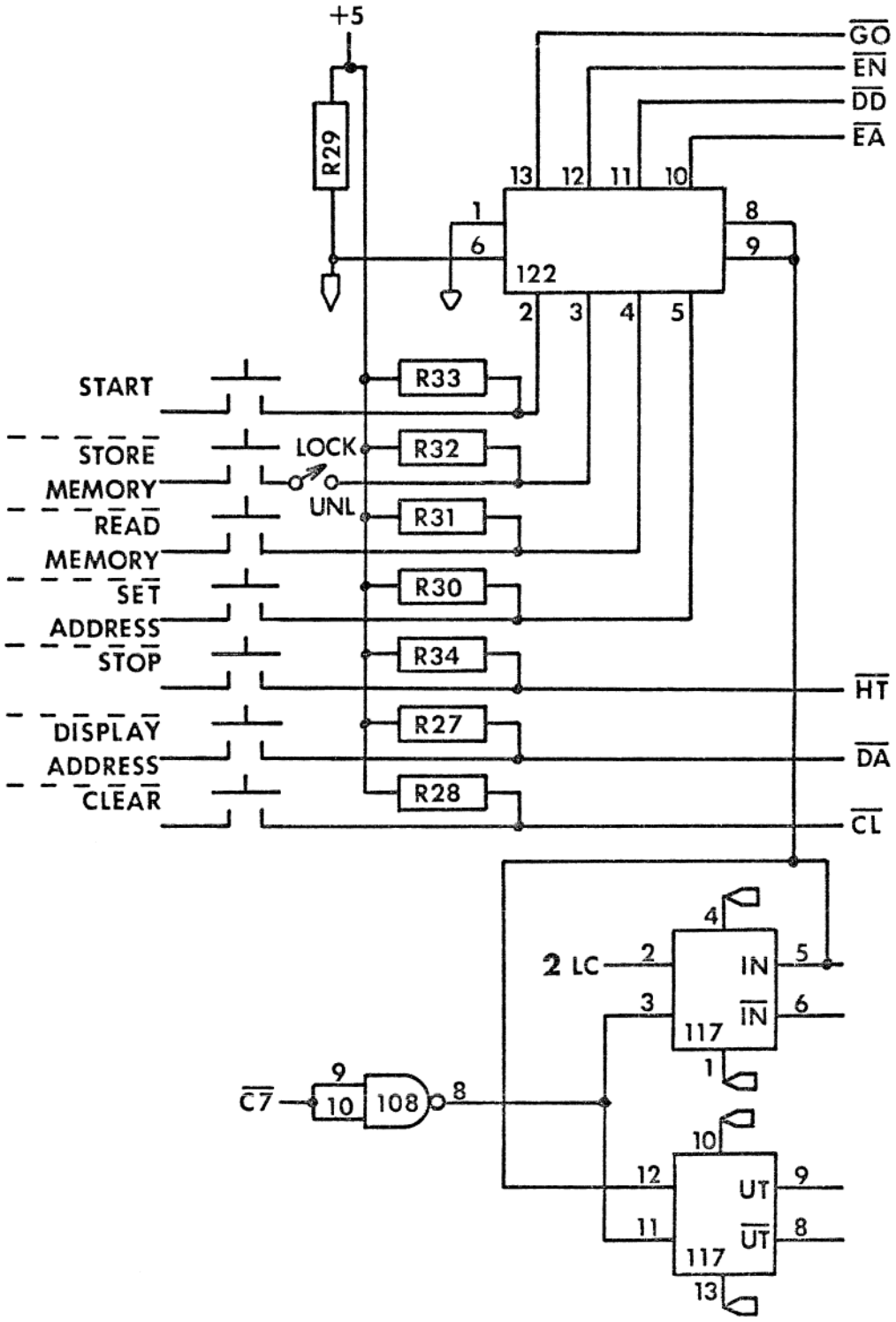
In addition, signals which originate and which are used on the same sheet are not cross-referenced. Cross referencing is only backwards, from use to source.

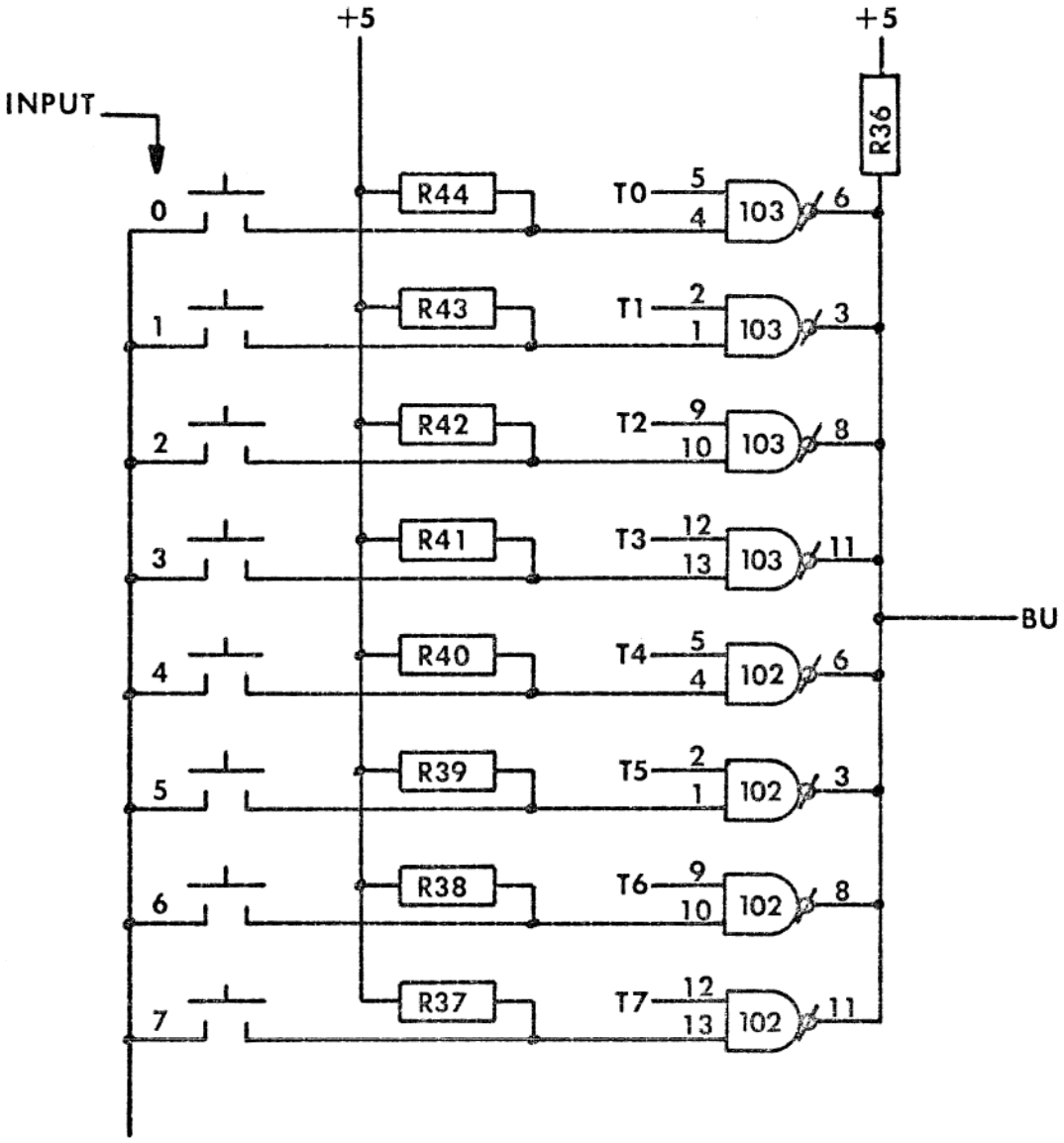
Each logic element shown in the schematics contains a one, two, or three digit number which is the number of the integrated circuit package. This identity number is etched on the printed circuit board.

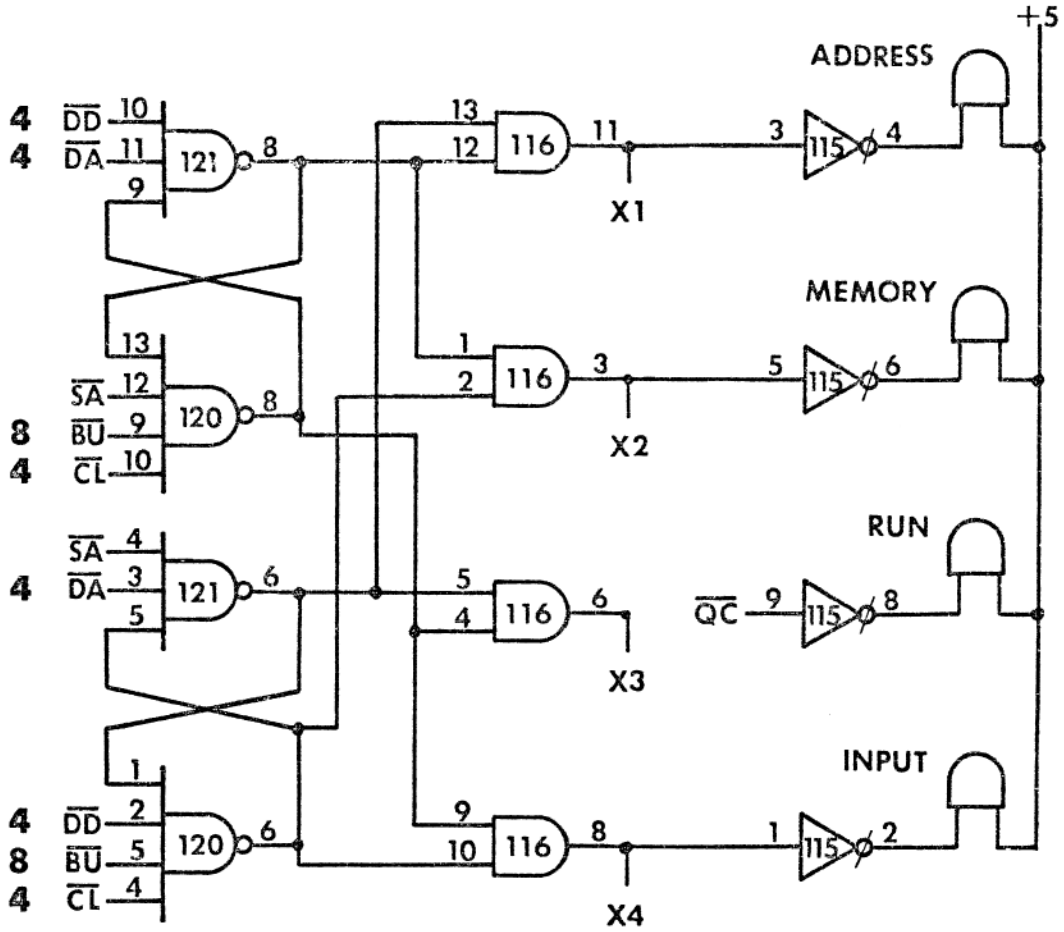

25
CT
79
91
80
TE
TT
F2
F1
+5
R18
R19
R20
R15
C11
CP
C7
T7
T0 T1 T2 T3
T4 T5 T6 T7
82
81
A
T4

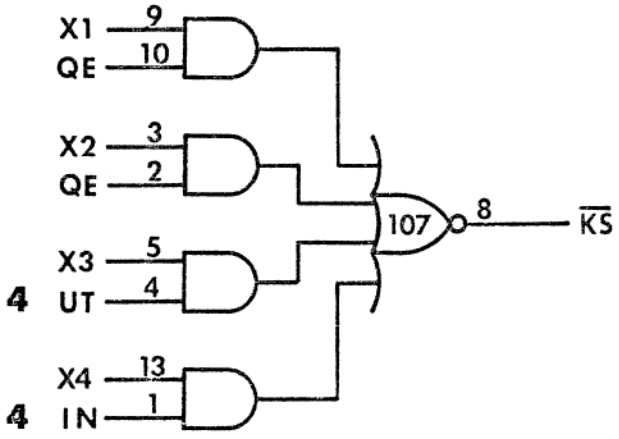

L7 L6 L5 L4
L3 L2 L1 L0
112
111
CP
−L0−
−JL7−
−CP−
LO
LC
100
113
T7
LC
CM
108
R
+5
R35
CP

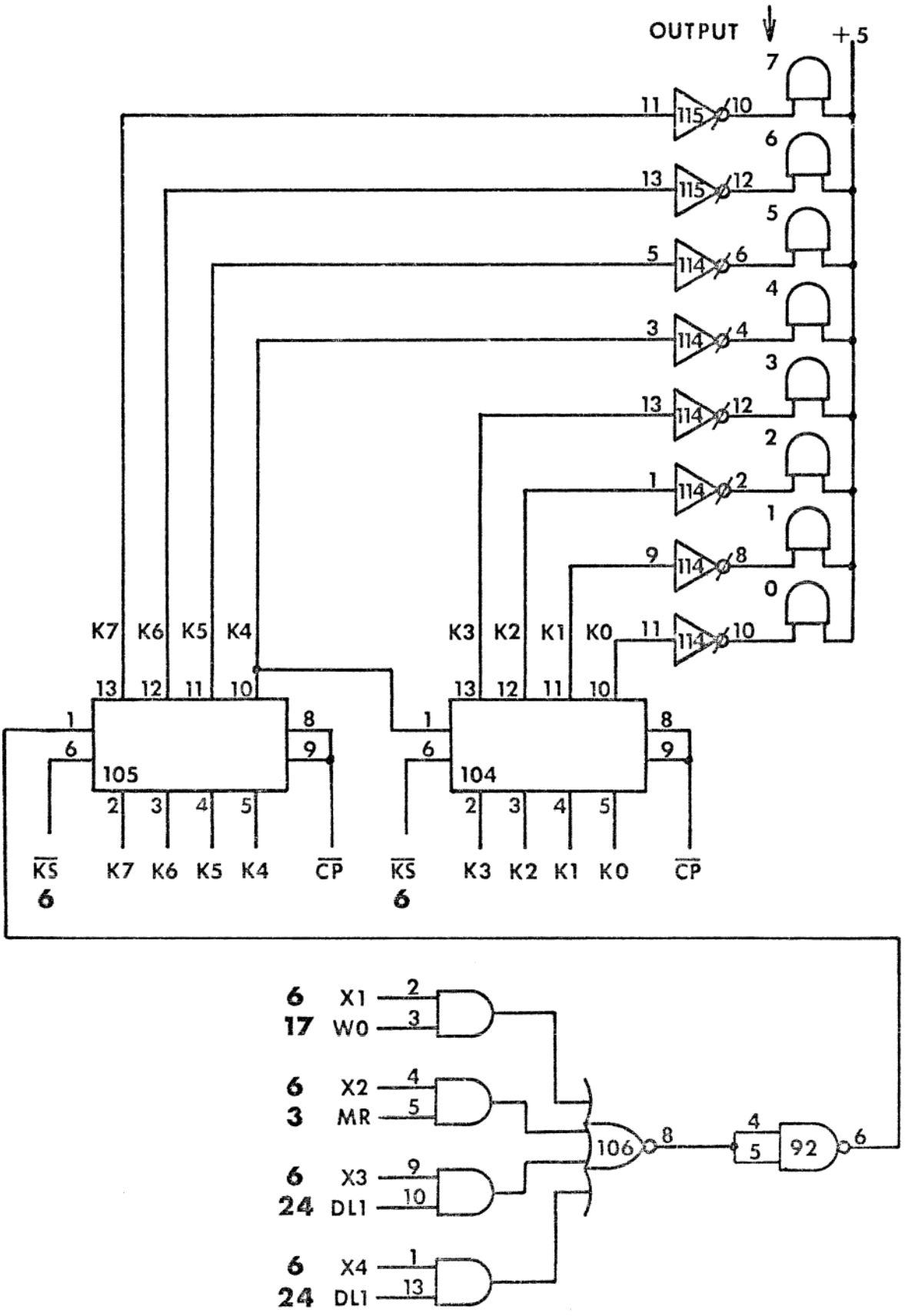

11 SU+SX
16 I5
15 I7,I6
SM
15 I7,I6
SA
ST
SH
T0
T1
-QC-
17 W0
93
R
QD
SC
SF
SK
SR
SD
96
+5
R17
17 W0
QC
92
69
21 C7
A7
68
24 DL1
24 DL0
25
MR

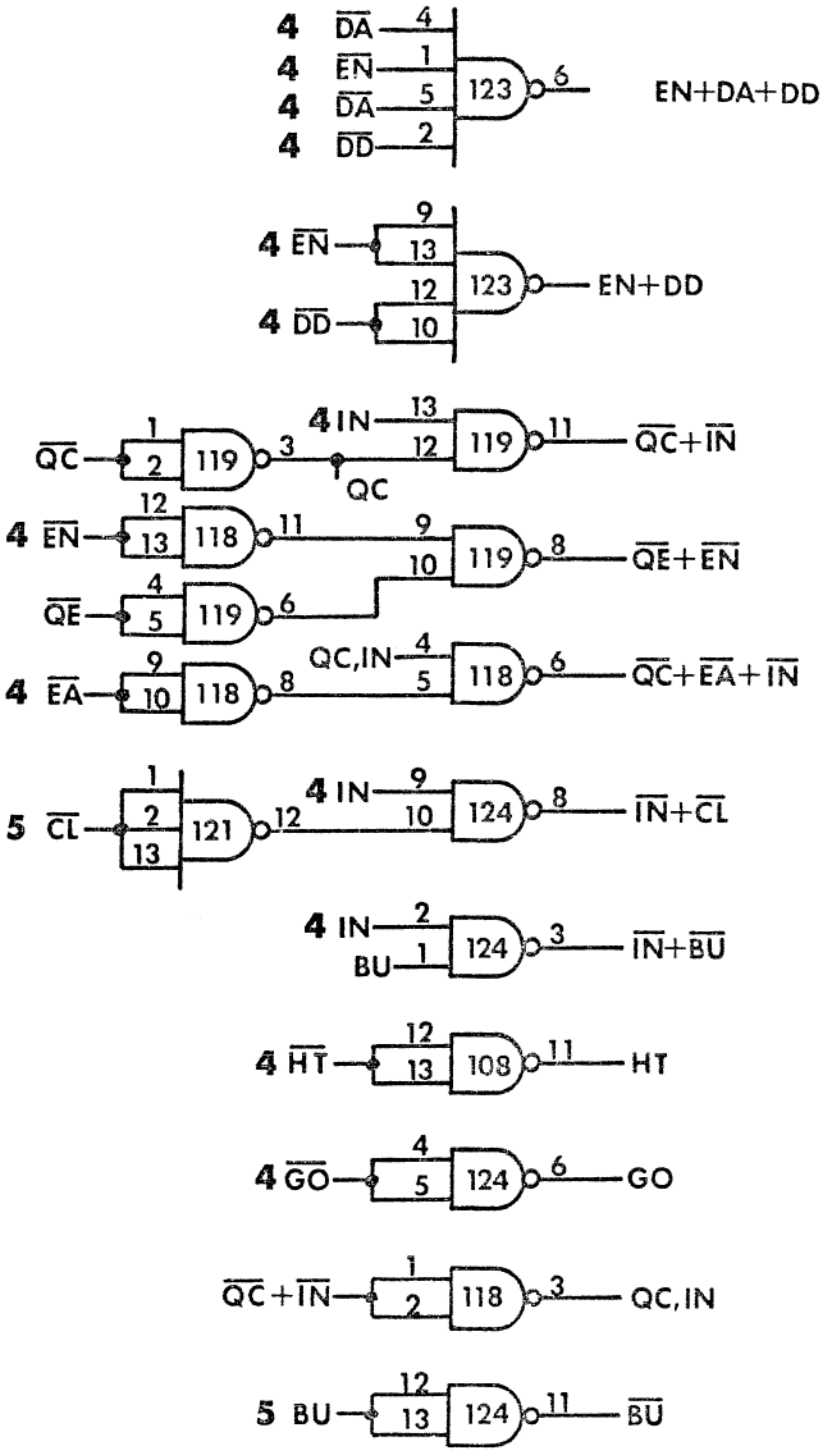

+5
R29
GO
EN
DD
EA
13
12
11
10
1
6
122
8
9
2
3
4
5
START
STORE
MEMORY
READ
MEMORY
SET
ADDRESS
STOP
DISPLAY
ADDRESS
CLEAR
LOCK
UNL
R33
R32
R31
R30
R34
R27
R28
HT
DA
CL
2 LC
IN
IN
117
C7
10
108
UT
UT
117

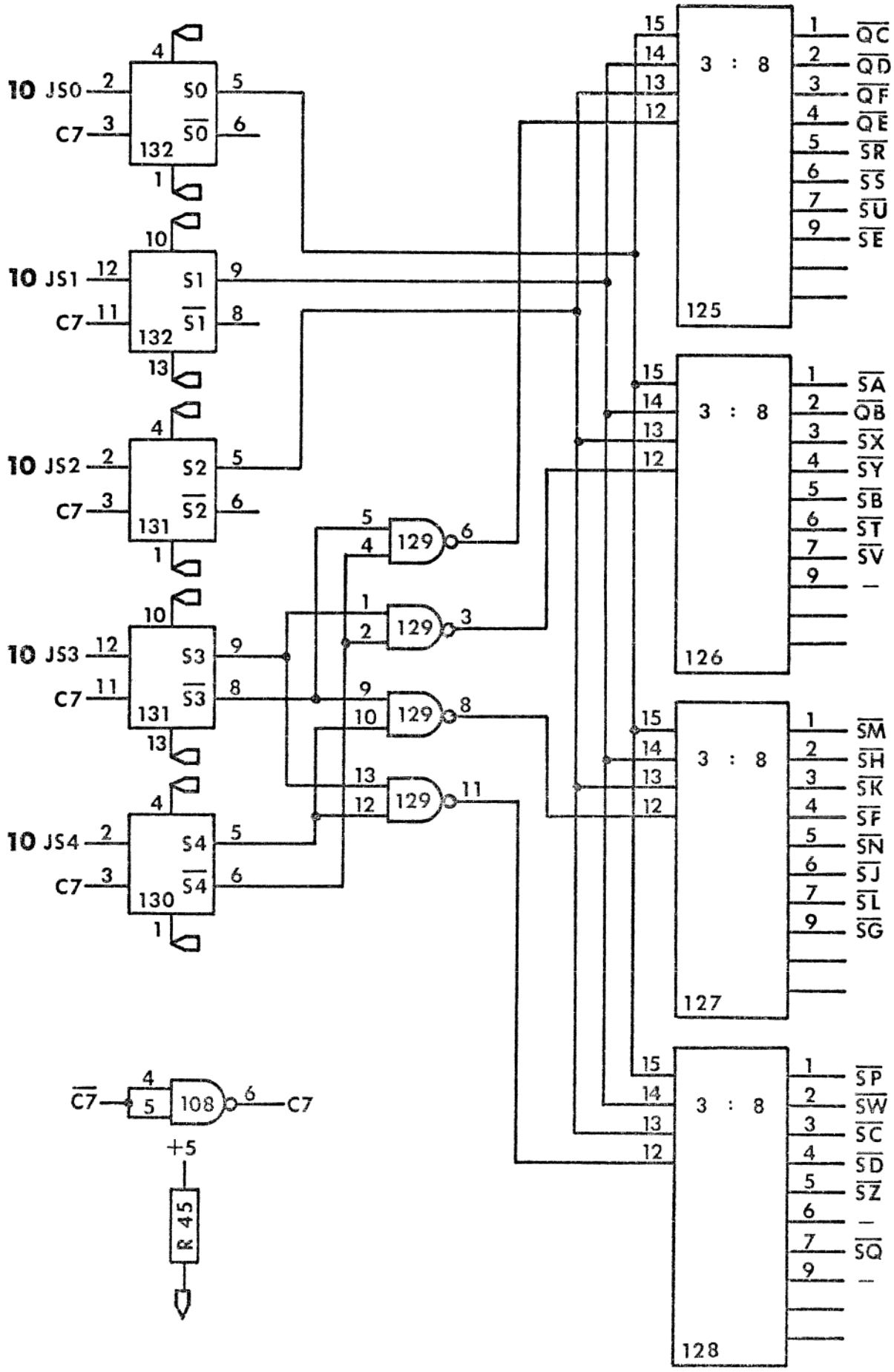

+5
+5
INPUT
R36
0
R44
T0
5
4
103
6
1
R43
T1
2
1
103
3
2
R42
T2
9
10
103
8
3
R41
T3
12
13
103
11
BU
4
R40
T4
5
4
102
6
5
R39
T5
2
1
102
3
6
R38
T6
9
10
102
8
7
R37
T7
12
13
102
11

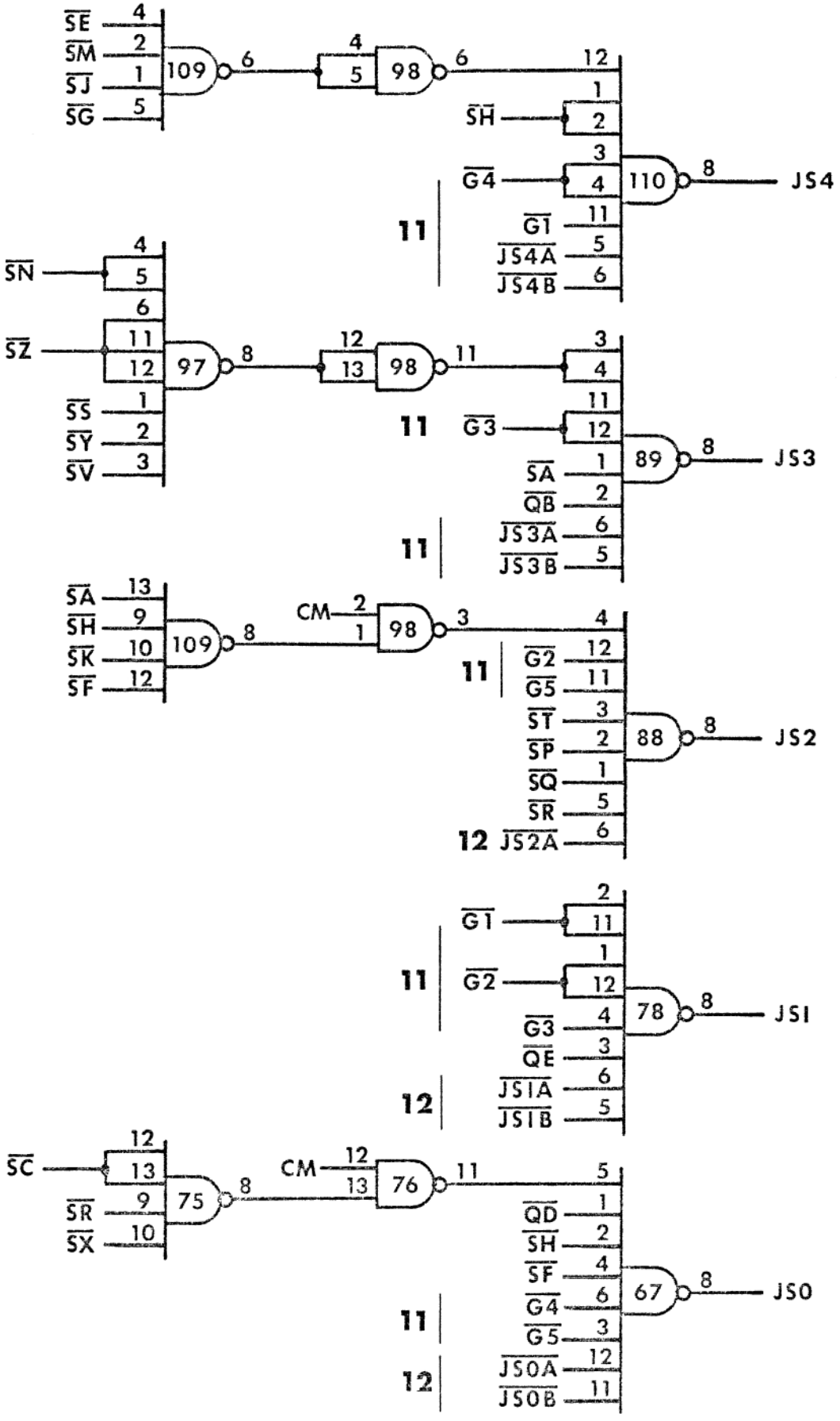

+5
ADDRESS
MEMORY
RUN
INPUT
4 DD 10
4 DA 11
9
121
8
13
12
116
11
X1
3
115
4
13
SA 12
8 BU 9
4 CL 10
120
8
1
2
116
3
X2
5
115
6
4 SA 4
4 DA 3
5
121
6
5
4
116
6
X3
QC 9
115
8
1
4 DD 2
8 BU 5
4 CL 4
120
6
9
10
116
8
X4
1
115
2

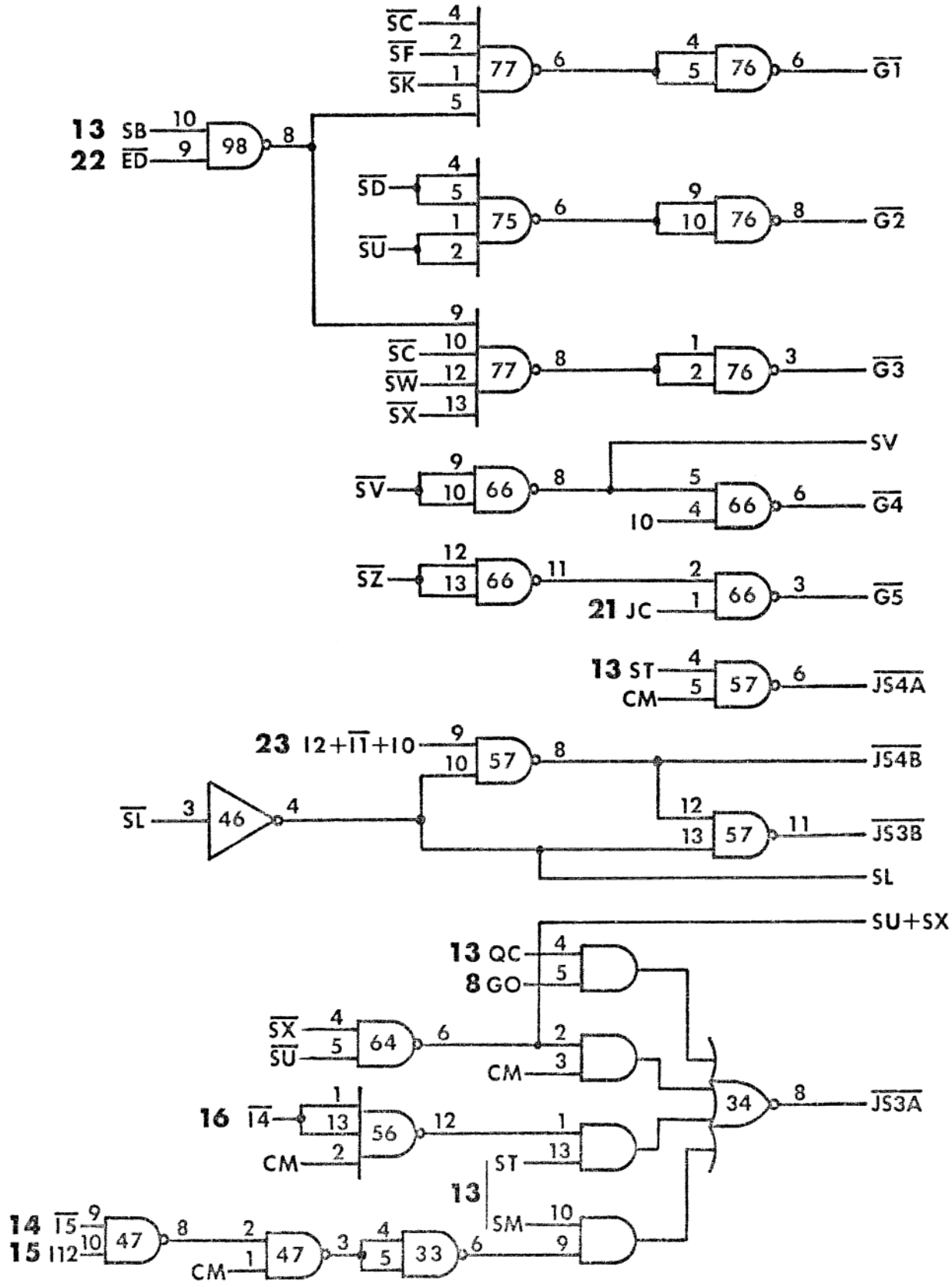

X1 9
QE 10
X2 3
QE 2
X3 5
4 UT 4
X4 13
4 IN 1
107
8
KS

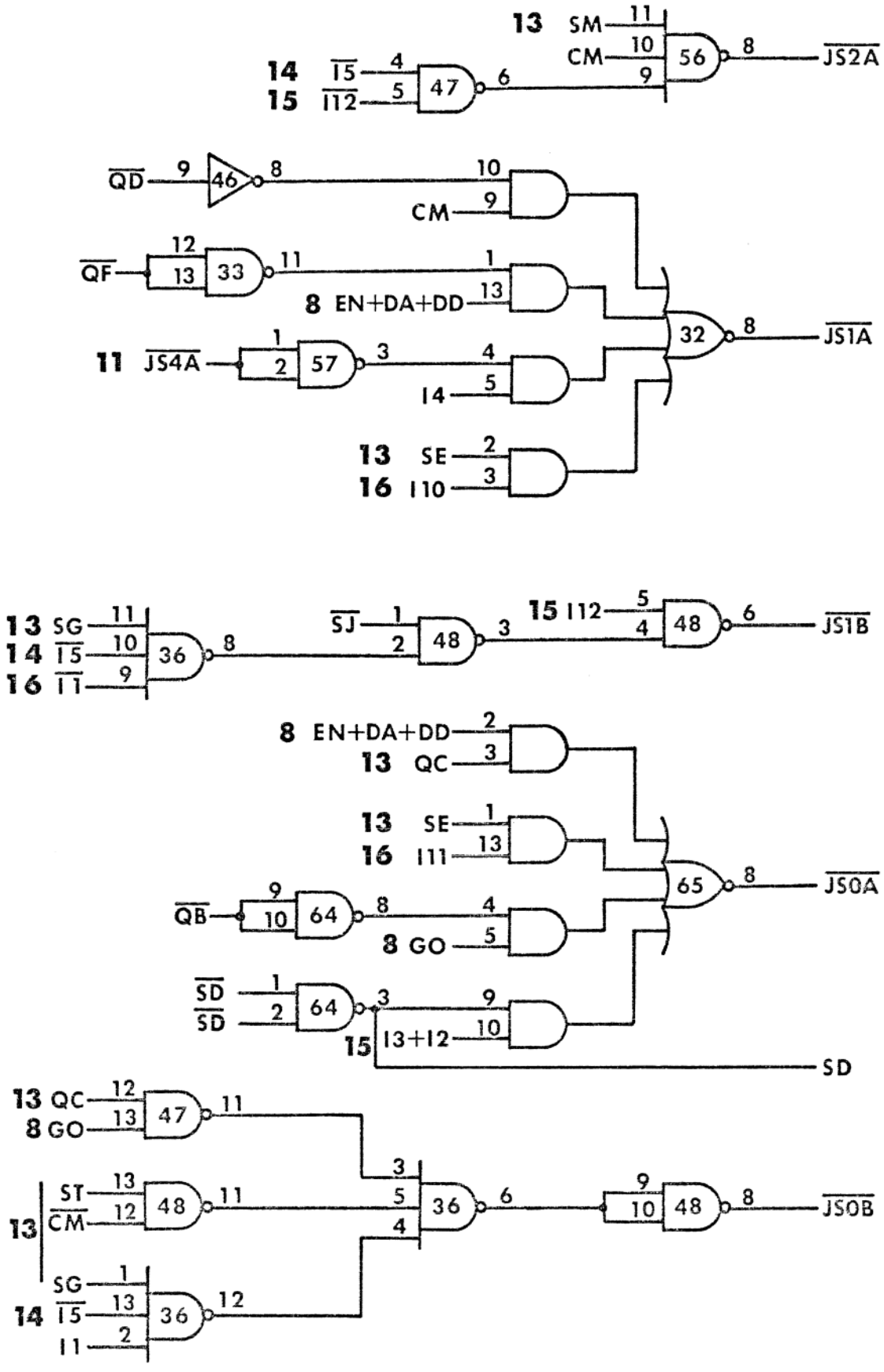

OUTPUT
+5
7
6
5
4
3
2
1
0
115
114
K7 K6 K5 K4
K3 K2 K1 K0
105
104
KS
CP
6
X1
W0
X2
MR
X3
DL1
X4
DL1
17
3
24
106
92

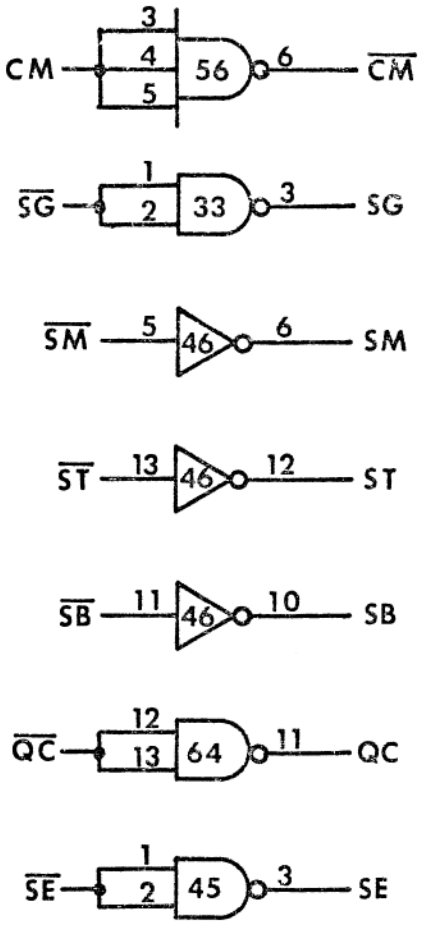

4 DA 4
4 EN 1
4 DA 5
4 DD 2
123 6
EN+DA+DD
4 EN 9
13
12
4 DD 10
123
EN+DD
QC 1 2 119 3
4 IN 13 12 119 11
QC+IN
4 EN 12 13 118 11
9 10 119 8
QE+EN
QE 4 5 119 6
4 EA 9 10 118 8
QC,IN 4 5 118 6
QC+EA+IN
5 CL 1 2 13 121 12
4 IN 9 10 124 8
IN+CL
4 IN 2
BU 1
124 3
IN+BU
4 HT 12 13 108 11
HT
4 GO 4 5 124 6
GO
QC+IN 1 2 118 3
QC,IN
5 BU 12 13 124 11
BU

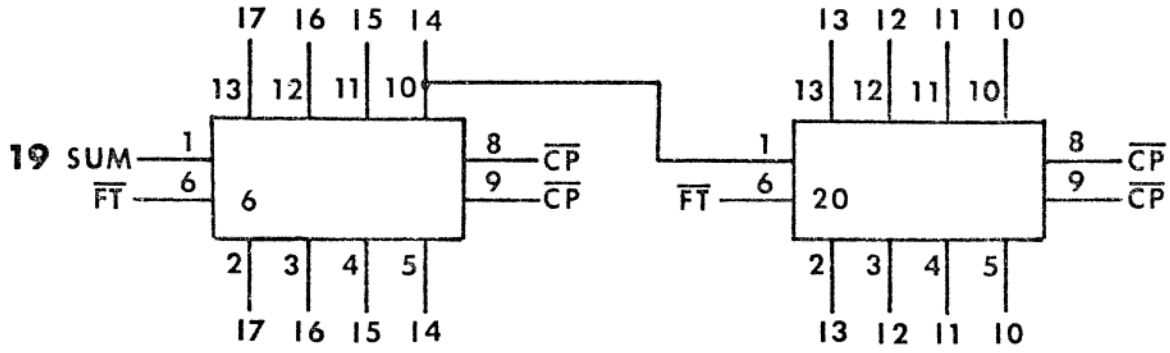

10 JS0
C7
S0
S0
132
10 JS1
C7
S1
S1
132
10 JS2
C7
S2
S2
131
10 JS3
C7
S3
S3
131
10 JS4
C7
S4
S4
130
129
129
129
129
3 : 8
125
3 : 8
126
3 : 8
127
3 : 8
128
QC
QD
QF
QE
SR
SS
SU
SE
SA
QB
SX
SY
SB
ST
SV
SM
SH
SK
SF
SN
SJ
SL
SG
SP
SW
SC
SD
SZ
SQ
C7
108
C7
+5
R 45

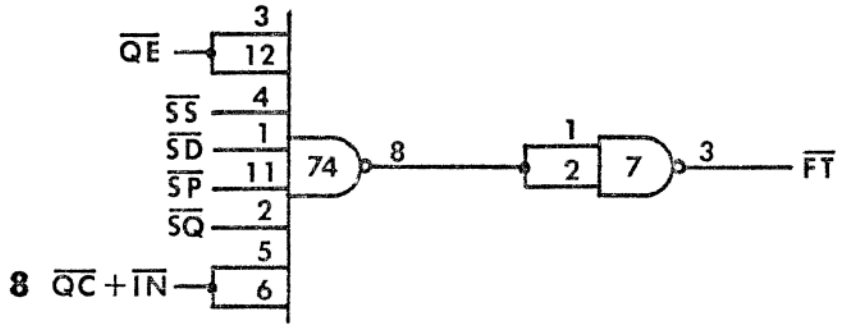

SE
SM
SJ
SG
109
98
SH
G4
110
JS4
11
G1
JS4A
JS4B
SN
SZ
97
98
SS
SY
SV
11
G3
89
JS3
SA
QB
JS3A
JS3B
SA
SH
SK
SF
109
CM
98
11
G2
G5
ST
SP
SQ
SR
12
JS2A
88
JS2
G1
11
G2
78
JS1
G3
QE
JS1A
JS1B
SC
SR
SX
75
CM
76
QD
SH
SF
G4
G5
JS0A
JS0B
67
JS0

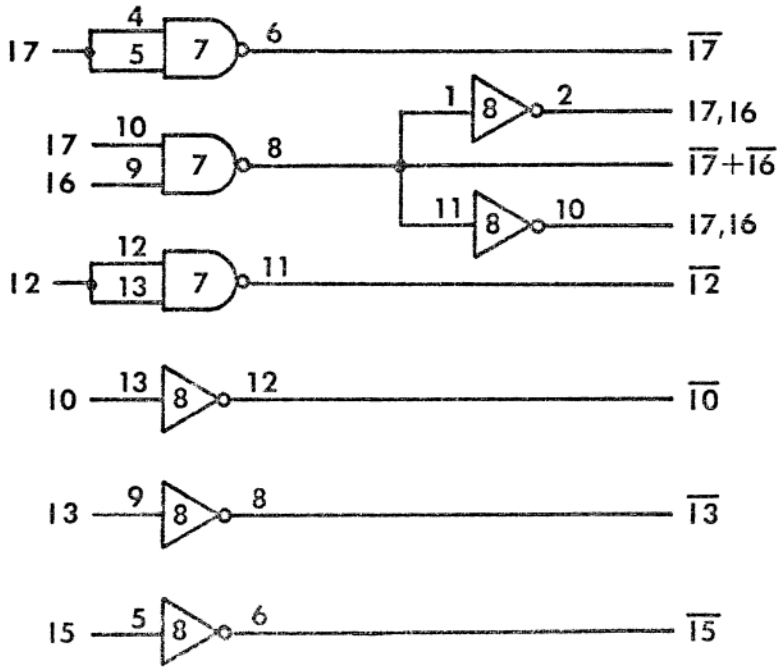

13 SB
22 ED
98
77
76
G1
75
G2
G3
SV
66
G4
SZ
G5
21 JC
13 ST
CM
57
JS4A
23 I2+I1+I0
JS4B
SL
46
JS3B
SU+SX
13 QC
8 GO
64
34
JS3A
16 I4
56
14 I5
15 I12
47
33

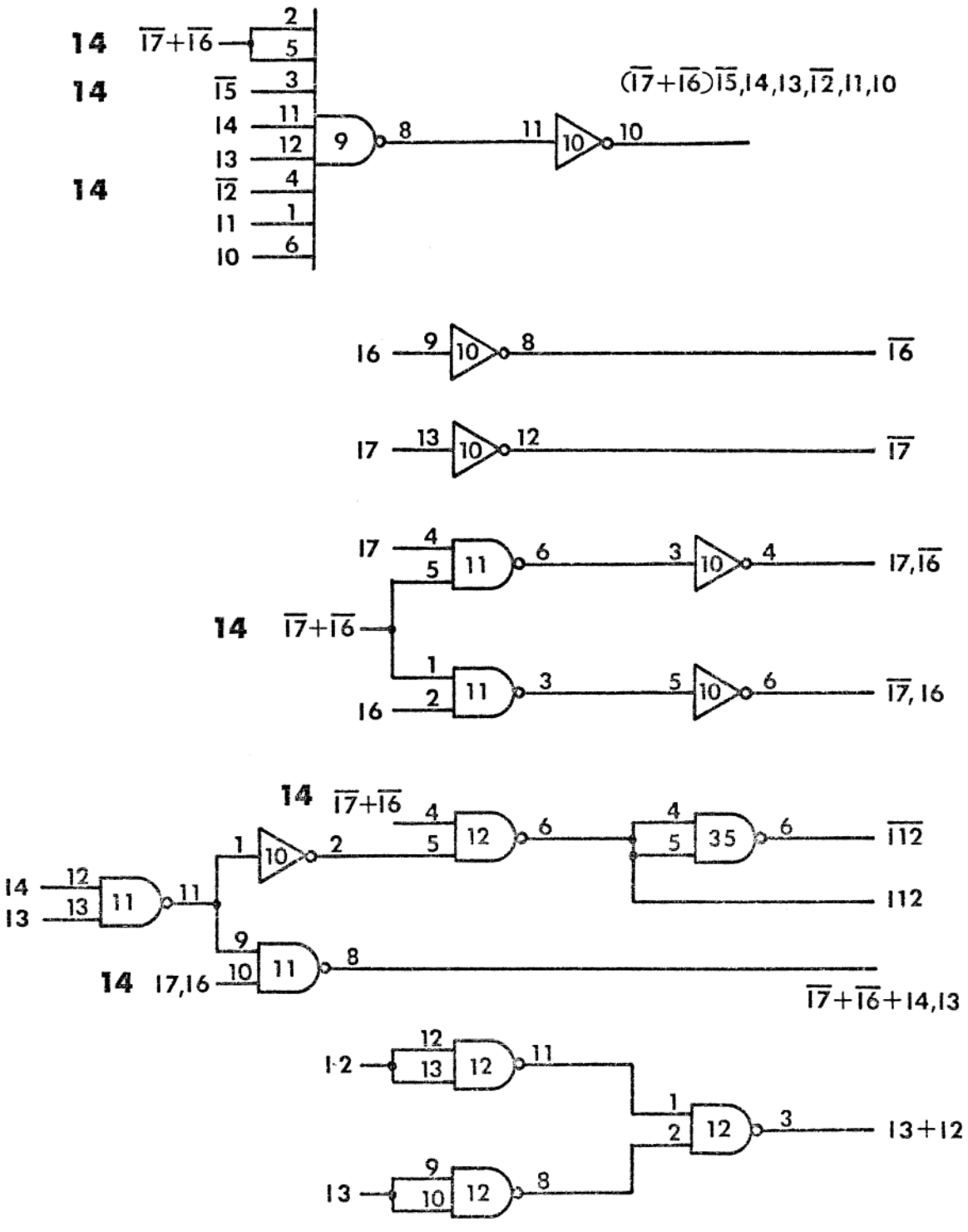

13 SM
CM
56
JS2A
14 I5
15 I12
47
QD
46
CM
QF
33
8 EN+DA+DD
32
JS1A
11 JS4A
57
14
13 SE
16 I10
13 SG
14 I5
16 I1
36
SJ
48
15 I12
48
JS1B
8 EN+DA+DD
13 QC
13 SE
16 I11
65
JS0A
QB
64
8 GO
SD
SD
64
15 I3+I2
SD
13 QC
8 GO
47
13
ST
CM
48
36
48
JS0B
14 I5
I1
36

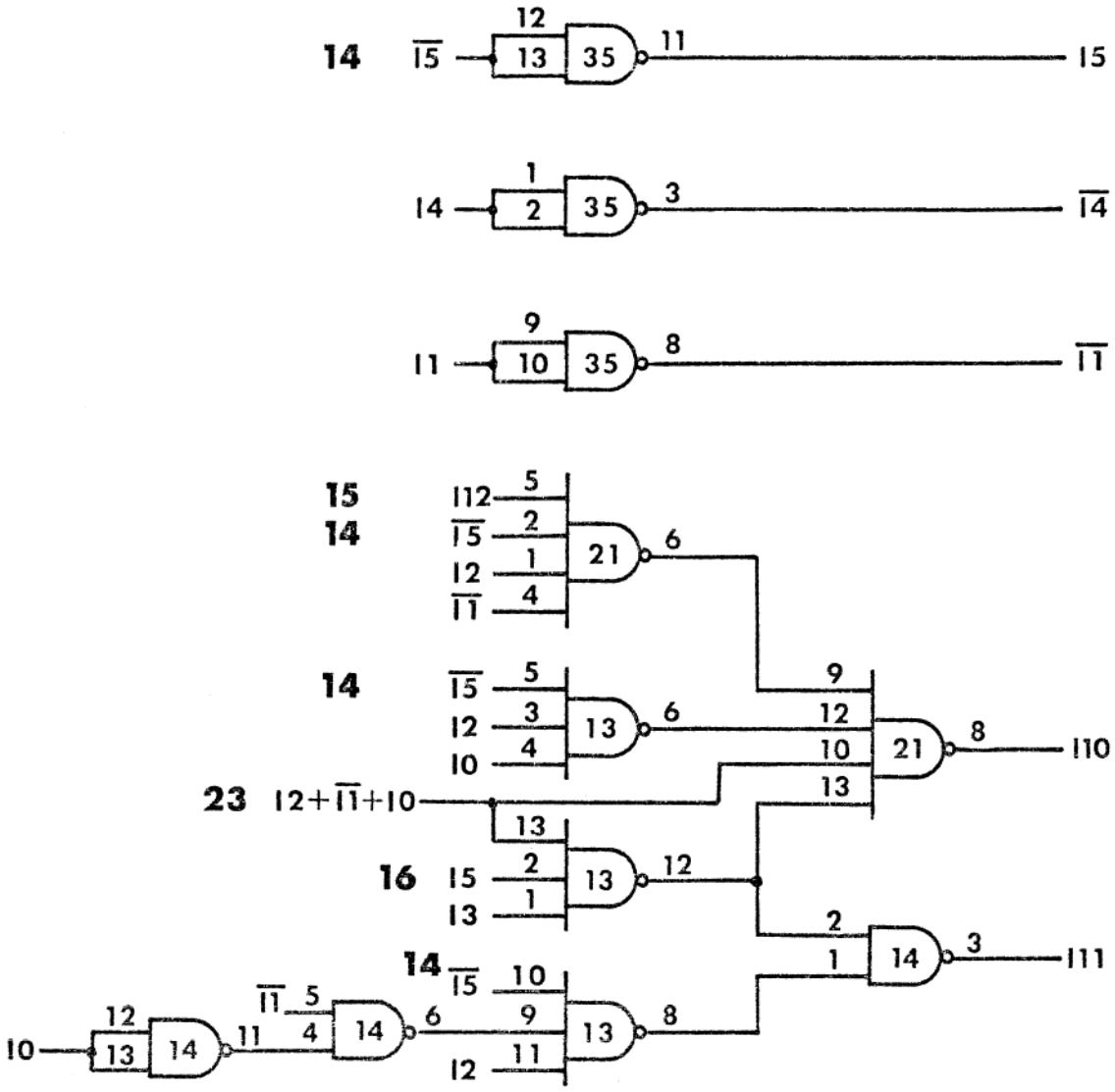

3
CM
4
56
6
$\overline{CM}$
5
1
$\overline{SG}$
2
33
3
SG
$\overline{SM}$
5
46
6
SM
$\overline{ST}$
13
46
12
ST
$\overline{SB}$
11
46
10
SB
12
$\overline{QC}$
13
64
11
QC
1
$\overline{SE}$
2
45
3
SE

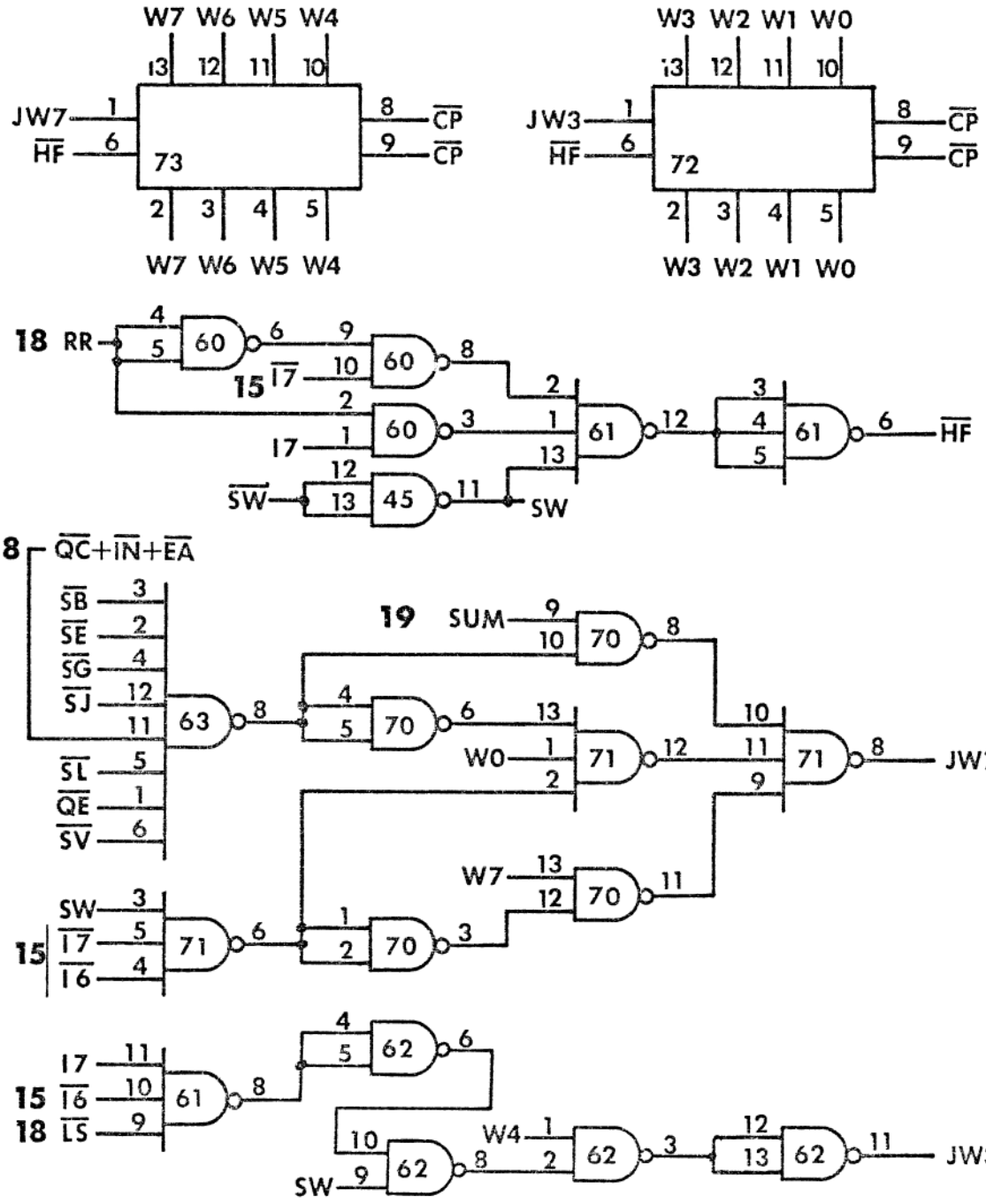

19 SUM
FT
CP
CP
6
20
17 16 15 14
13 12 11 10
2 3 4 5

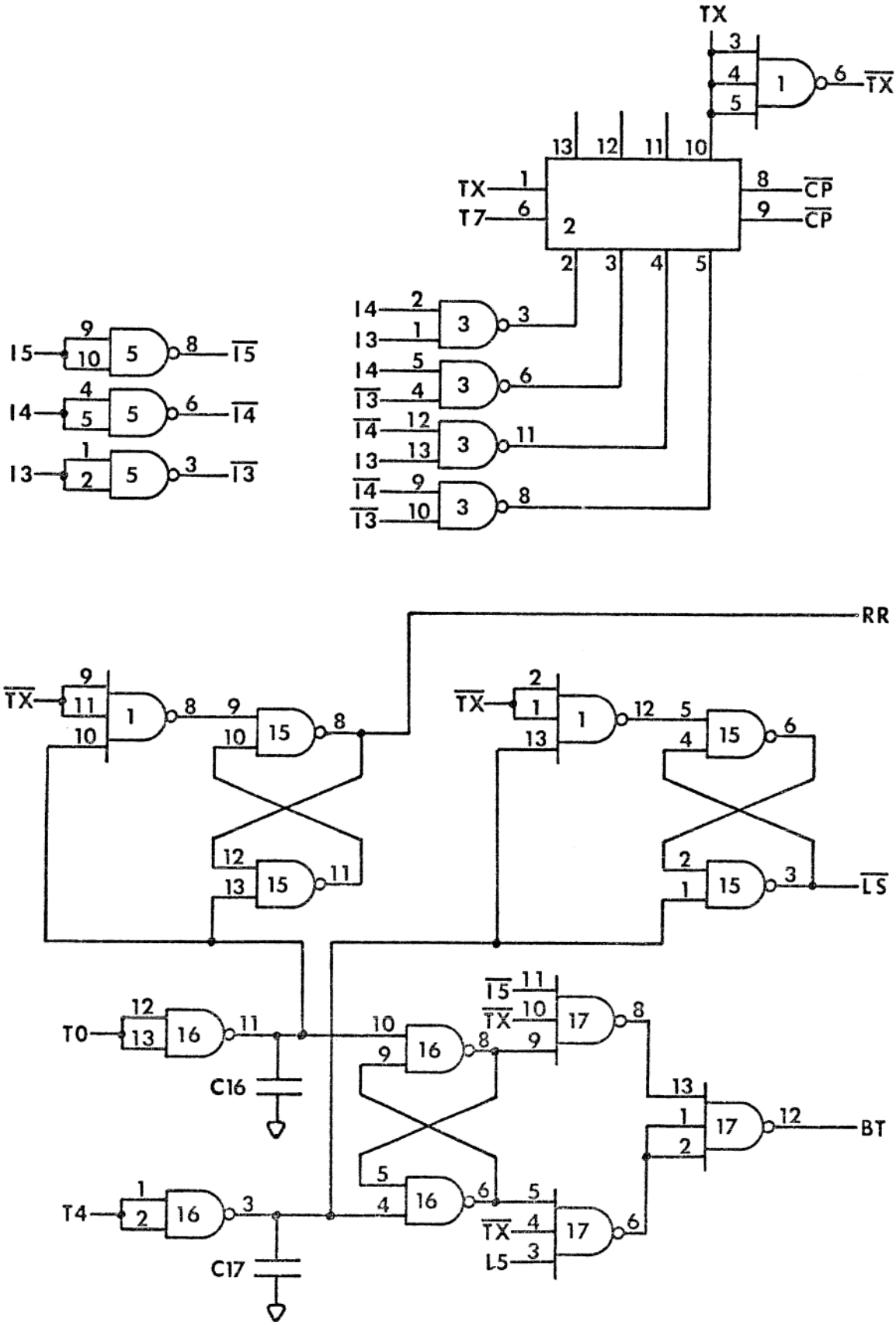

QE
SS
SD
SP
SQ
8 QC + IN
74
FT

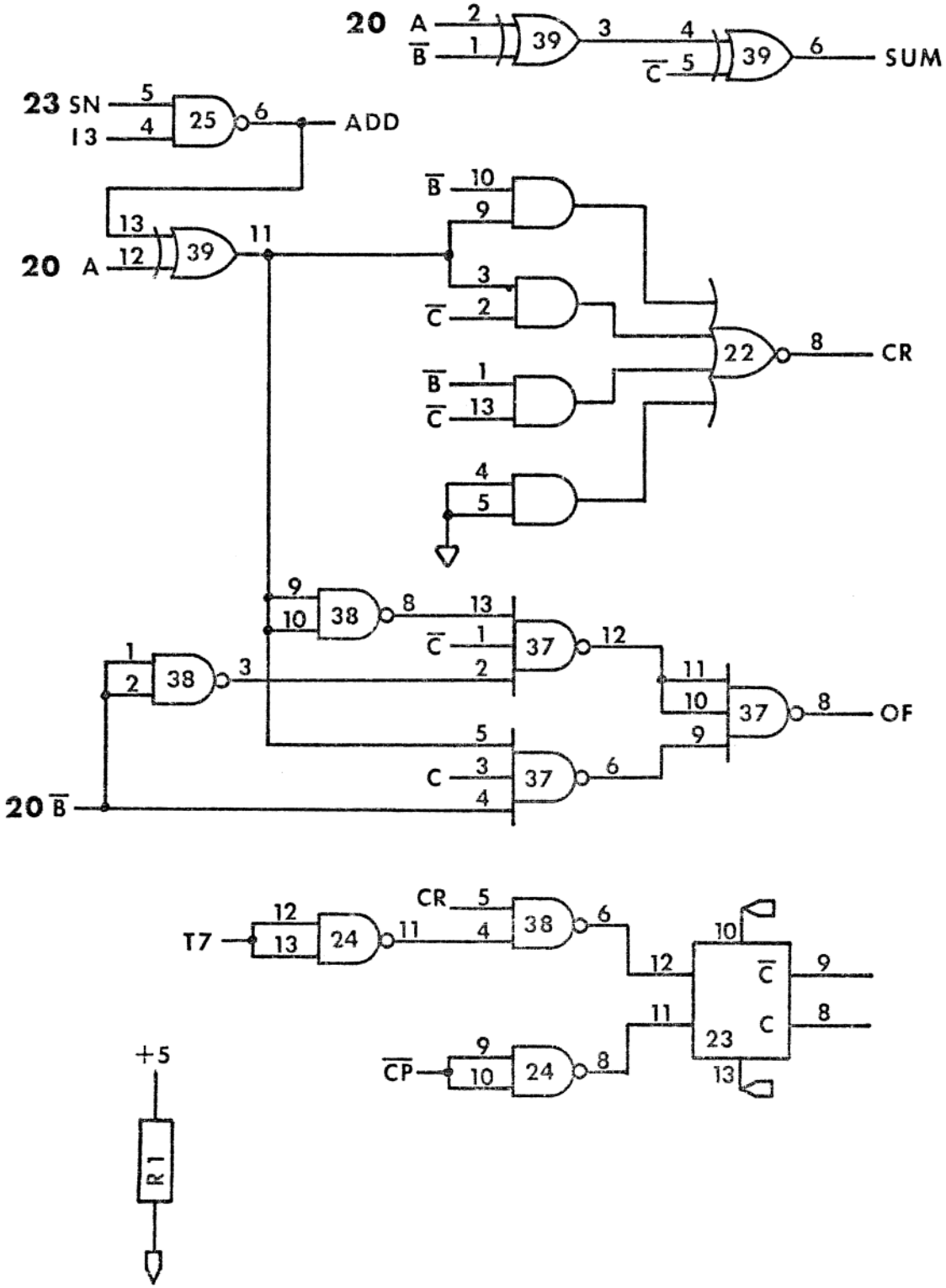

17
17
17,16
17 + 16
17,16
16
12
12
10
10
13
13
15
15

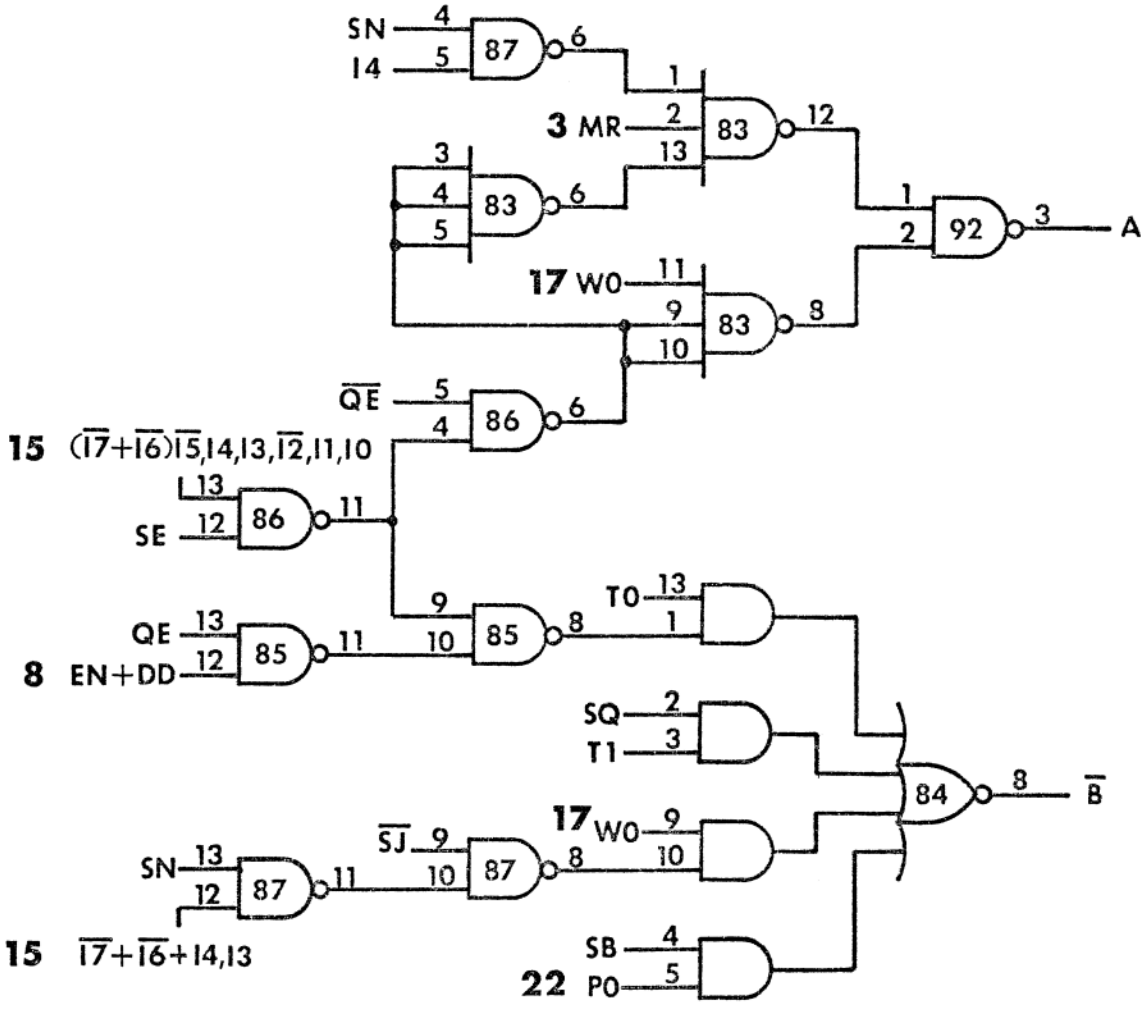

14 $\overline{17}+\overline{16}$
14 $\overline{15}$
14 $\overline{12}$
14 17,16
$(\overline{17}+\overline{16})\overline{15},14,13,\overline{12},11,10$
16
17
$\overline{16}$
$\overline{17}$
$17,\overline{16}$
$\overline{17},16$
$\overline{112}$
112
$\overline{17}+\overline{16}+14,13$
13+12

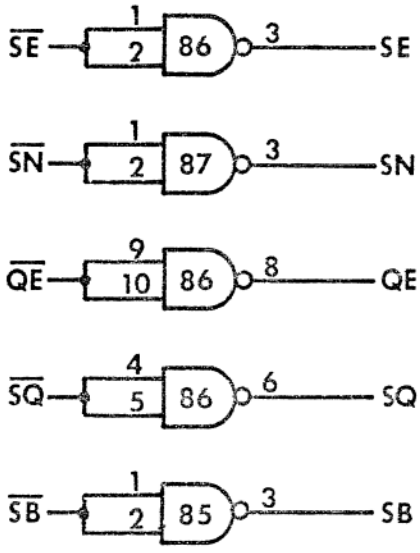

14
15
12
13
35
11
15
14
1
2
35
3
14
11
9
10
35
8
11
15
14
112
5
15
2
12
1
11
4
21
6
14
15
5
12
3
10
4
13
6
9
12
10
13
21
8
110
23
12+11+10
13
16
15
2
13
1
13
12
2
1
14
3
111
14
15
10
11
5
12
13
14
11
4
14
6
9
13
8
10
12
11
12
13
14
11
4
14
6
9
13
8
10
12
11

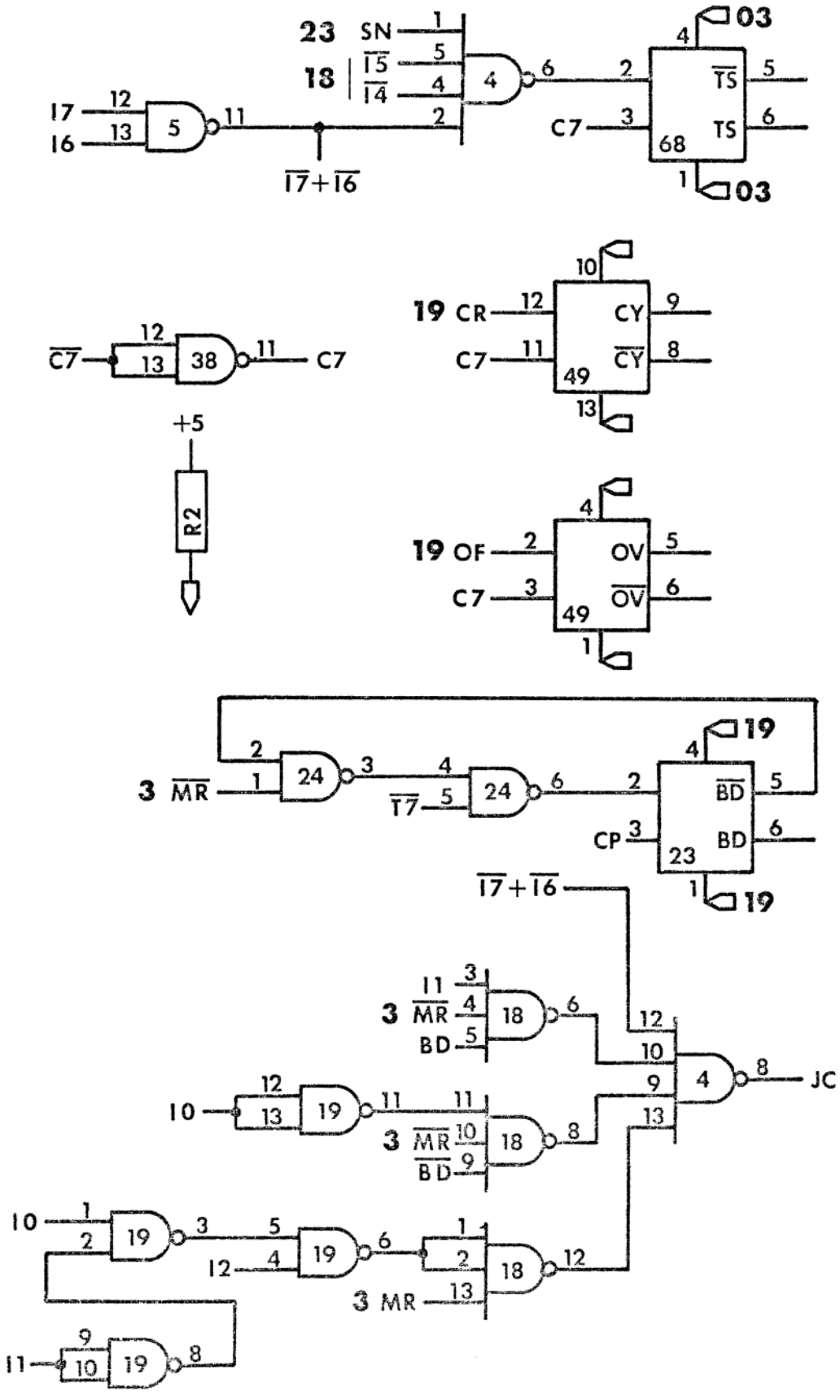

W7 W6 W5 W4
JW7
HF
73
CP
CP
W7 W6 W5 W4
W3 W2 W1 W0
JW3
HF
72
CP
CP
W3 W2 W1 W0
18 RR
15 I7
I7
SW
SW
HF
8 QC+IN+EA
SB
SE
SG
SJ
SL
QE
SV
63
19 SUM
70
W0
71
JW7
W7
70
15 SW
I7
I6
71
70
15 I7
I6
18 LS
61
62
W4
62
SW
62
JW3

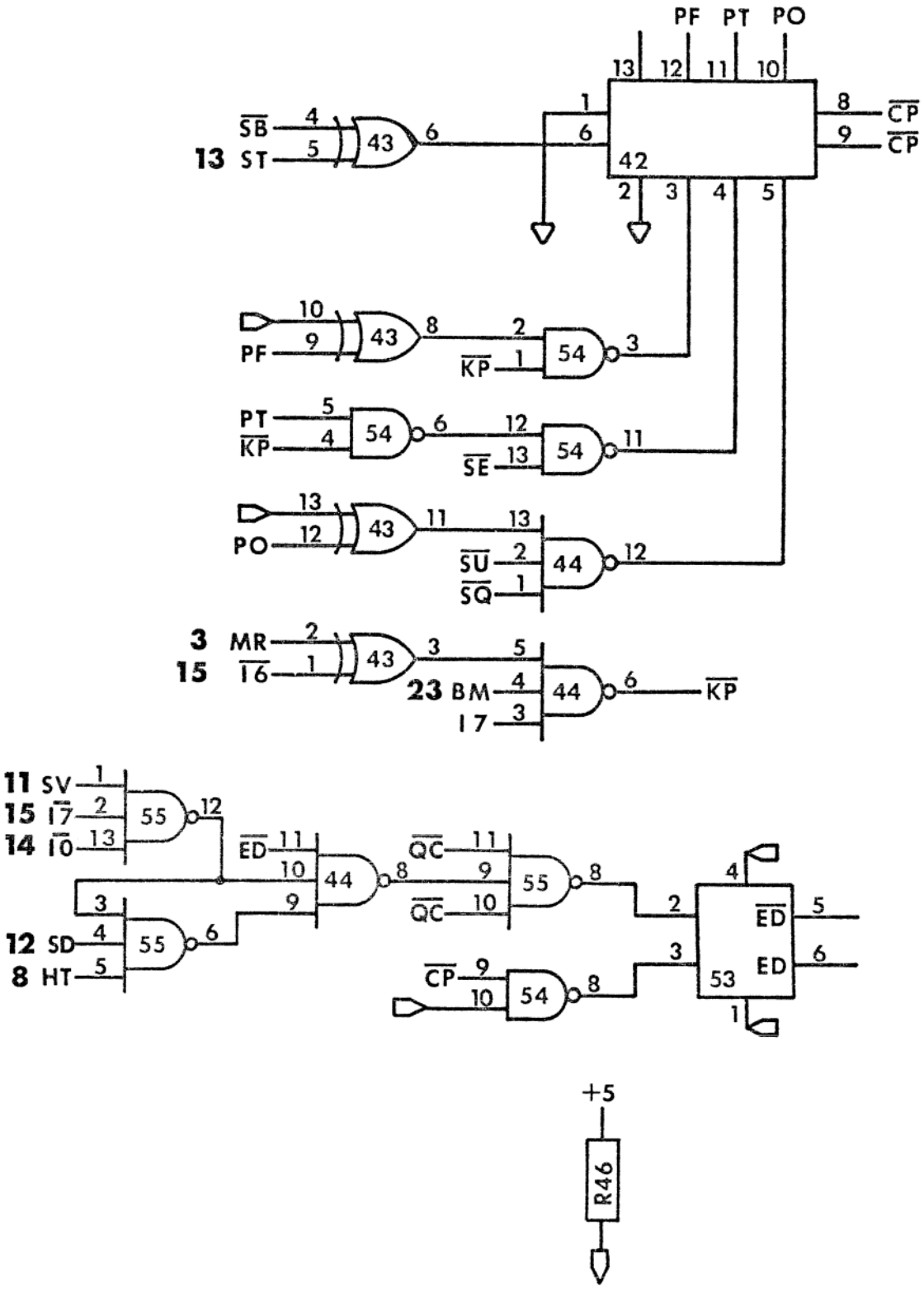

TX
TX
T7
CP
CP
I5
I4
I3
I5
I4
I3
RR
TX
TX
LS
T0
C16
T4
C17
BT
L5

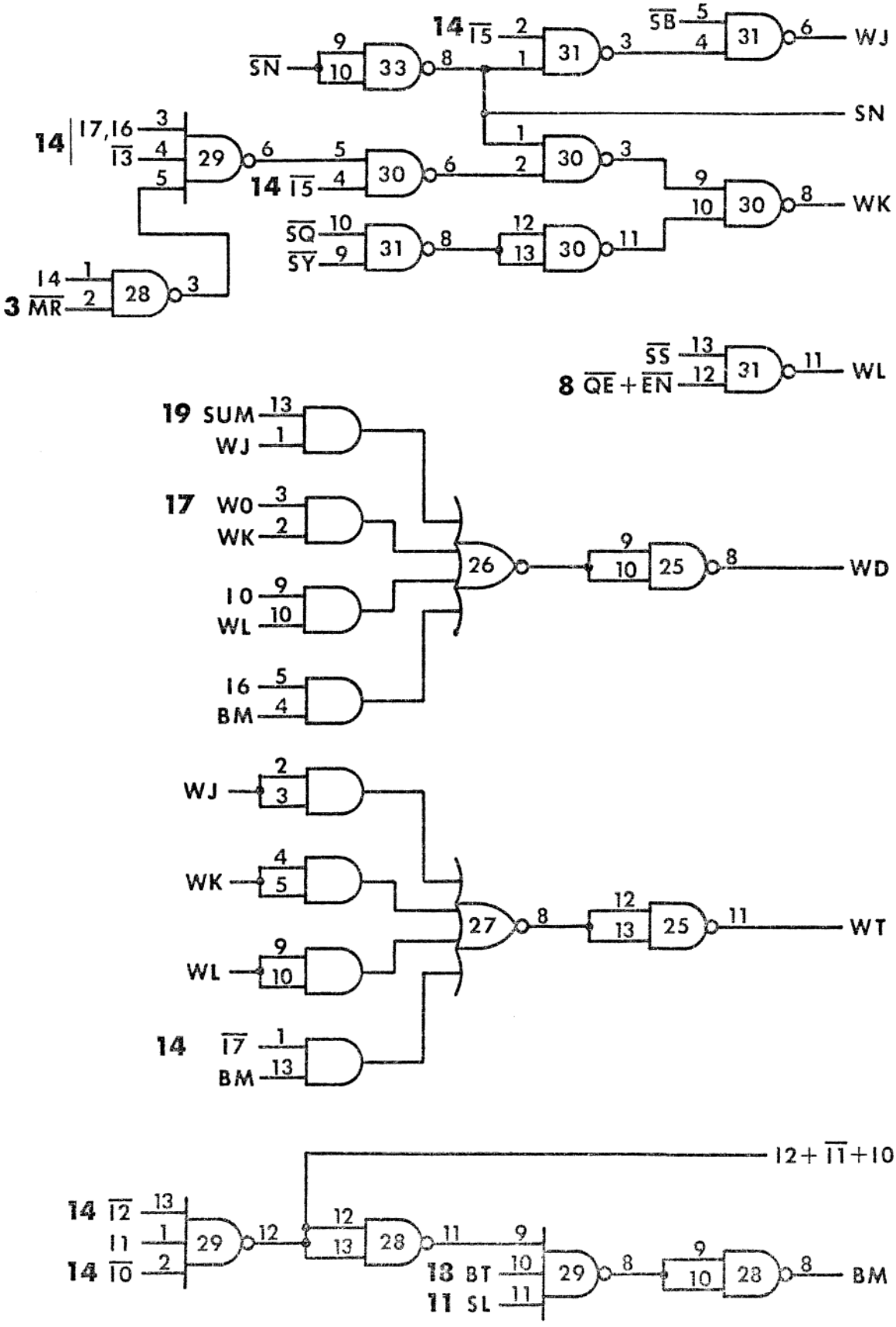

20
A
B
2
1
39
3
4
C
5
39
6
SUM
23
SN
13
5
4
25
6
ADD
20
A
13
12
39
11
B
10
9
3
C
2
B
1
C
13
4
5
22
8
CR
9
10
38
8
13
C
1
2
37
12
1
2
38
3
11
10
37
8
OF
5
C
3
4
37
6
9
20
B
CR
5
T7
12
13
24
11
4
38
6
10
12
C
9
11
C
8
23
9
CP
10
24
8
13
+5
R1

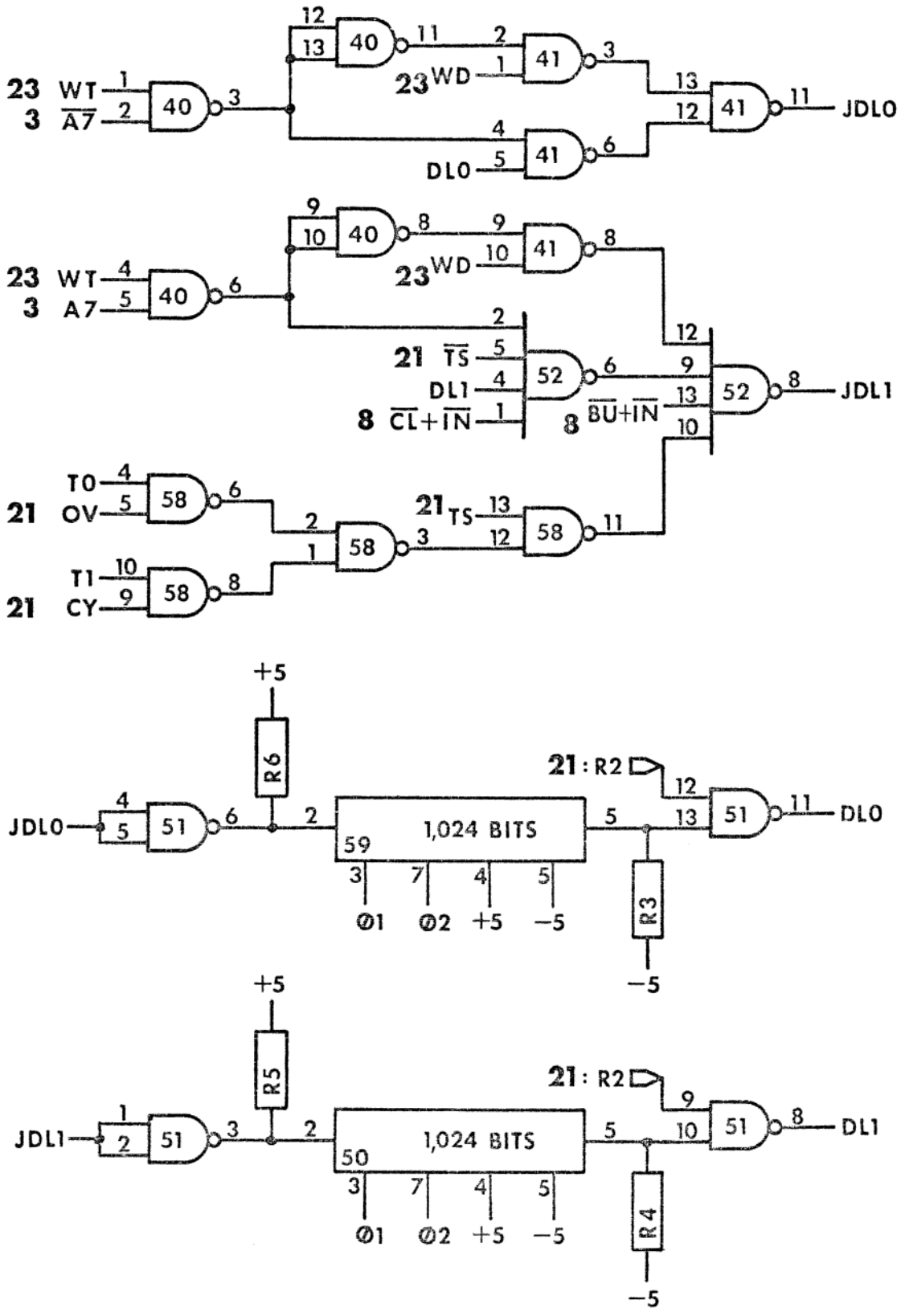

SN
14
87
3 MR
83
83
17 WO
83
92
A
QE
86
15 (17+16)15,14,13,12,11,10
SE
86
QE
EN+DD
85
85
T0
SQ
T1
84
B
17 WO
SJ
SN
87
87
15 17+16+14,13
SB
22 PO

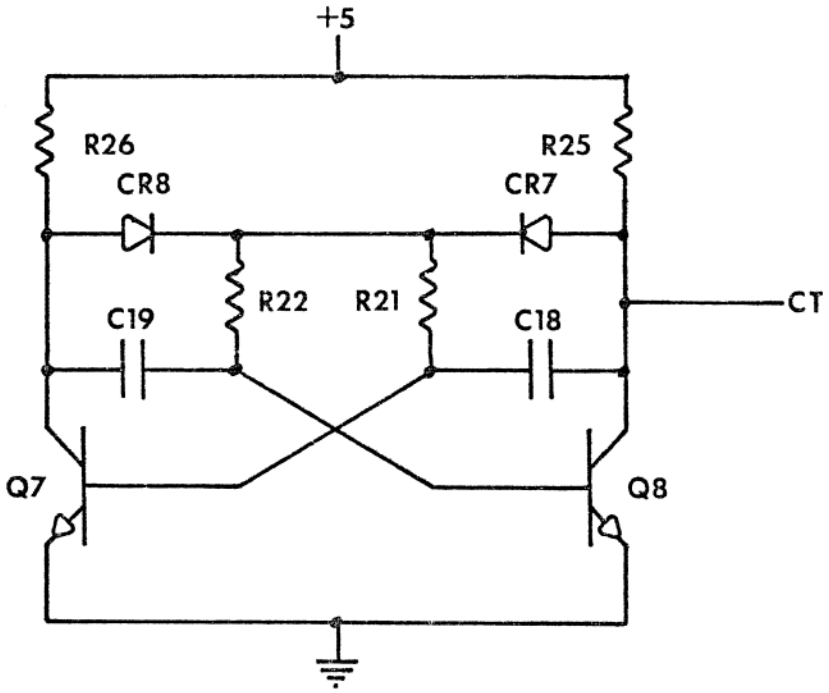

SE
86
SE
SN
87
SN
QE
86
QE
SQ
86
SQ
SB
85
SB

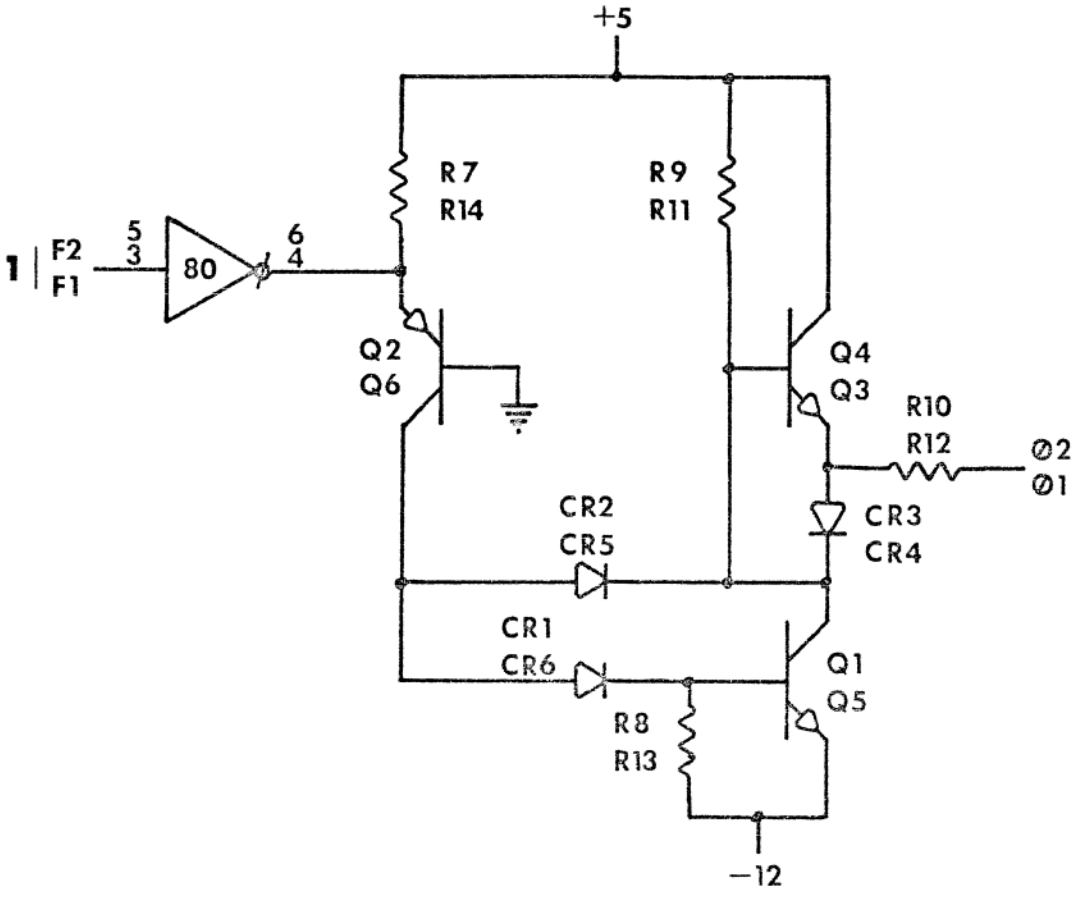

23
SN
18
I5
I4
17
16
I7+I16
03
TS
C7
68
C7
38
C7
19
CR
CY
C7
CY
49
+5
R2
19
OF
OV
C7
OV
49
19
3
MR
24
I7
24
BD
CP
23
BD
19
I7+I16
3
MR
BD
18
JC
10
19
3
MR
BD
18
10
19
19
12
MR
18
11
19
10

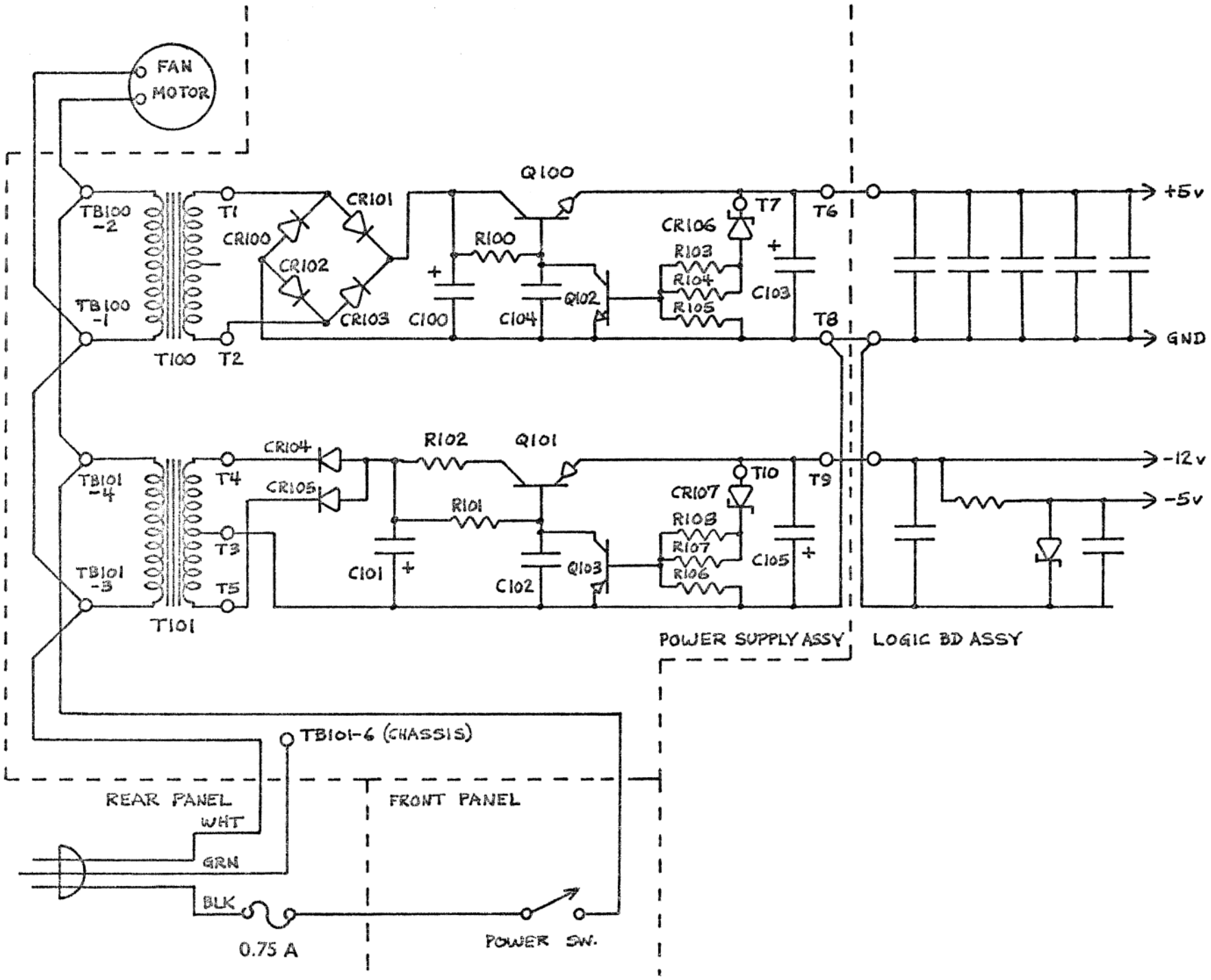

PF PT PO
13 12 11 10
1
6
42
2 3 4 5
8 $\overline{CP}$
9 $\overline{CP}$
$\overline{SB}$ 4
13 ST 5
43
6
10
PF 9
43
8
2
$\overline{KP}$ 1
54
3
PT 5
$\overline{KP}$ 4
54
6
12
$\overline{SE}$ 13
54
11
13
PO 12
43
11
13
$\overline{SU}$ 2
$\overline{SQ}$ 1
44
12
3 MR 2
15 $\overline{I6}$ 1
43
3
5
23 BM 4
I7 3
44
6
$\overline{KP}$
11 SV 1
15 $\overline{I7}$ 2
14 $\overline{I0}$ 13
55
12
$\overline{ED}$ 11
10
44
8
$\overline{QC}$ 11
9
55
8
$\overline{QC}$ 10
3
12 SD 4
8 HT 5
55
6
9
4
2
$\overline{ED}$ 5
3
53
ED 6
1
$\overline{CP}$ 9
10
54
8
+5
R46


14 I5
SB
WJ
SN
SN
14 I7,I6
I3
14 I5
WK
SQ
SY
3 MR
8 QE + EN
SS
EN
WL
19 SUM
WJ
17 W0
WK
I0
WL
I6
BM
WD
WJ
WK
WL
14 I7
BM
WT
12 + I1 + I0
14 I2
I1
14 I0
13 BT
11 SL
BM


23 WT
3 A7
40
41
23 WD
DL0
JDL0
DL1
JDL1
21 TS
8 CL+IN
8 BU+IN
52
T0
T1
21 OV
21 CY
58
21 TS
+5
R6
R5
R3
R4
21 : R2
JDL0
JDL1
51
1,024 BITS
59
50
Ø1
Ø2
−5
DLO
DL1


+5
R26
CR8
R25
CR7
R22
R21
CT
C19
C18
Q7
Q8


+5
R7
R14
R9
R11
1
F2
F1
5
3
80
6
4
Q2
Q6
Q4
Q3
R10
R12
Ø2
Ø1
CR2
CR5
CR3
CR4
CR1
CR6
Q1
Q5
R8
R13
−12


FAN
MOTOR
TB100
-2
TB100
-1
T100
T1
T2
CR100
CR101
CR102
CR103
C100
R100
Q100
C104
Q102
CR106
R103
R104
R105
T7
C103
T6
T8
+5v
GND
TB101
-4
TB101
-3
T101
T4
T3
T5
CR104
CR105
R102
Q101
R101
C101
C102
Q103
T10
CR107
R108
R107
R106
C105
T9
-12v
-5v
POWER SUPPLY ASSY
LOGIC BD ASSY
TB101-6 (CHASSIS)
REAR PANEL
FRONT PANEL
WHT
GRN
BLK
0.75 A
POWER SW.