

IBM

Customer Engineering
Manual of Instruction

1620

Data Processing System
E Level

Issued to: _____

Branch Office: _____

Department: _____

Address: _____

If this manual is misplaced, it should
be returned to the above address.

IBM Customer Engineering Manual of Instruction
1620 Data Processing System
E Level

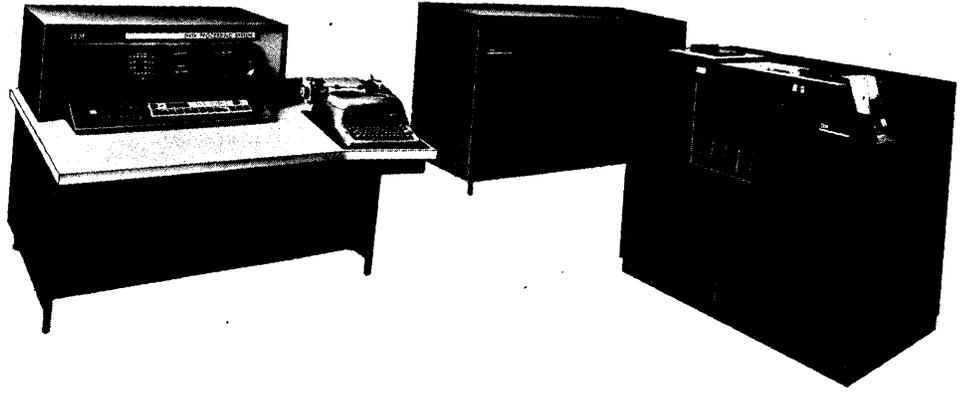
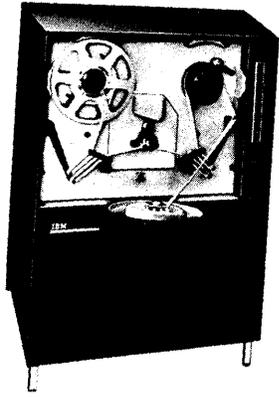
This manual describes the IBM 1620 Data Processing System (serial No. 10701 and up) and associated Input/Output equipment.
For machines prior to the E-level suffix (below serial No. 10700), order Form 227-5507-1.

© 1962 by International Business Machines Corporation

International Business Machines Corp., Product Publications Dept., San Jose, Calif.

Contents

	<i>Page</i>		<i>Page</i>
Section 1—General Information	5	Section 10—Arithmetic Operations	85
Description of System	5	Add (Code 21 - A)	85
Machine Language	6	Add Immediate (Code 11 - AM)	92
Stored Program Concept	7	Subtract (Code 22 - S)	92
Section 2—Memory Addressing and Data Flow	11	Subtract Immediate (Code 12 - SM)	97
Introduction	11	Compare (Code 24 - C)	97
Memory	11	Compare Immediate (Code 14 - CM)	102
Sense Amplifiers and Memory Buffer Register (MBR)	11	Multiply (Code 23 - M)	102
Memory Data Register (MDR)	11	Multiply Immediate (Code 13 - MM)	107
Memory Address Register Storage (MARS)	13	Load Dividend (Code 28 - LD)	109
Increment/Decrement Switch	14	Load Dividend Immediate (Code 18 - LDM)	111
Section 3—Programming	17	Divide (Code 29 - D)	111
Development of a Program	17	Divide Immediate (Code 19 - DM)	120
Insertion of a Program into Memory (Typewriter Keyboard)	17	Section 11—Branch Operations	122
Execution of a Program	17	Branch (Code 49 - B)	122
Automatic Program Repetition (Program Loop)	19	Branch and Transmit (Code 27 - BT)	122
Insertion of a Program into Memory (Paper Tape Reader)	19	Branch and Transmit Immediate (Code 17 - BTM)	124
Section 4—Components and Power Supply	21	Branch Back (Code 42 - BB)	124
Magnetic Cores	21	Branch On Digit (Code 43 - BD)	124
Component Circuits	22	Branch No Flag (Code 44 - BNF)	125
Power Supplies	35	Branch No Record Mark (Code 45 - BNR)	126
Section 5—Functional Units	40	Branch Indicator (Code 46 - BI)	127
Clock and Timing Chart	40	Branch No Indicator (Code 47 - BNI)	130
Memory	44	Section 12—Miscellaneous Operations	131
Memory Buffer Register (MBR)	49	Set Flag (Code 32 - SF)	131
Memory Data Register (MDR)	49	Clear Flag (Code 33 - CF)	131
Memory Address Register Storage (MARS)	49	Halt (Code 48 - H)	132
Memory Address Register (MAR)	53	No Operation (Code 41 - NOP)	132
Increment/Decrement Switch	53	Section 13—Input-Output Devices	133
Operation Register and Decoder	55	Typewriter	133
Digit/Branch Register and Decoder	55	IBM 1621 Paper Tape Reader	133
True/Complement (T/C) Switch	55	IBM 1624 Paper Tape Punch	143
Multiplier/Quotient Register	56	Section 14—Input-Output Operations	151
Doubler	57	General	151
C Bit Correctors	57	Typewriter	151
Section 6—Console and CE Panel	60	Paper Tape	152
Console	60	IBM 1621 Paper Tape Reader	153
Customer Engineering Panel	69	IBM 1624 Paper Tape Punch	153
Section 7—Instruction Cycle	71	Read Numerically (Code 36 - RN)	153
Objectives	71	Read Alphamerically (Code 37 - RA)	158
Functions	71	Write Numerically (Code 38 - WN)	159
Section 8—Execution Cycle	76	Dump Numerically (Code 35 - DN)	166
Objective	76	Write Alphamerically (Code 39 - WA)	169
Functions	76	Control (Code 34 - K)	170
E-Timer Triggers.....	76	Section 15—Appendices	173
Auxiliary Triggers	76	A. Abbreviations and Terms	173
Read Y Point	79	B. Character Code Chart	174
Block Memory Sense Amplifiers	80	C. Numbered Triggers	175
Significance of P & Q Parts of Instructions	80	D. Named Triggers	175
Section 9—Internal Transmission Operations	82	E. Register Triggers	176
Transmit Digit (Code 25 - TD)	82	F. E-Timer Trigger Chart	177
Transmit Digit Immediate (Code 15 - TDM)	82	G. Auxiliary Trigger Chart.....	178
Transmit Field (Code 26 - TF)	82	Glossary	179
Transmit Field Immediate (Code 16 - TFM)	83	Index	181
Transmit Record (Code 31 - Mnemonic TR)	84		



IBM 1620 Data Processing System

Description of System

The IBM 1620 Data Processing System is a solid-state electronic computer system, designed specifically for technological applications. The system is composed of separate units: the 1620 Computer Unit, the 1621 Paper Tape Reader which provides space for the 1624 Paper Tape Punch, the 1622 Card Read Punch, and the 1623 Core Storage unit. The 1621, 1622, and the 1624 are input-output units available for various customer applications. The 1623 is available if additional core storage is desired. Information on the 1622 and 1623 units is included in the *IBM Customer Engineering Manual of Instruction, 1620 Data Processing System, Additional Features* (Form 227-5513). The computer unit contains the logic circuitry, the 20,000 position core storage unit, the console panel, and a typewriter.

Data and instructions entered into the system are placed in memory as decimal digits. Each of the 20,000 positions of core storage (memory) can be addressed individually by a 5-digit address and can store one digit of information. Memory addresses extend from 00000 to 19999. The addressing system provides for the selection of any digit or group of digits within memory. The 1620 Computer processes numerical, alphabetic and special characters.

The computer is capable of performing more than thirty different operations. Each operation is specified by a 12-digit instruction which contains a two-digit operation code and two 5-digit addresses. Instructions comprising a program are normally stored in consecutive locations in memory and executed sequentially. However, the sequence of operations may be altered at any point in the program by conditional branch instructions. Conditional branch instructions make logical decisions by performing tests on indicators or switches set by the computer or the operator.

Arithmetic operations are accomplished by a table method. Three hundred twenty (320) positions of memory are assigned for use in arithmetic operations. Twenty (20) positions, 00080 through 00099, are used to store products or quotients. Two hundred (200) positions, 00100 through 00299, are assigned for the storage of a Multiply Table. The Multiply Table contains all possible two-digit products. One hundred (100) positions, 00300 through 00399, are assigned for storage of an Add Table used in all arithmetic operations. The Add Table contains all possible two-digit

sums, with carries indicated. The memory positions containing the table data are addressable.

The 1620 is a variable field length computer. The shortest admissible field is two digits; the longest can be any number of digits within the capacity of memory. Accuracy of data is ensured by the parity check which is made when the data enters, exits, or is manipulated inside the system.

The console of the 1620 consists of control keys, switches, an indicator panel, and a typewriter. The control keys and switches are used to control operation of the system. The console panel provides a visual indication of the contents of various registers and the status of control circuitry within the computer. The typewriter keyboard is used for manual entry of data and instructions into memory.

Information is entered into the system by the typewriter, 1621 Paper Tape Reader, and the 1622 Card Read Punch. The 1621 reads an eight-channel tape at a rate of 150 characters per second. The 1622 reads 80-column IBM cards at a rate of 250 cards per minute. Output data is recorded by the typewriter, 1622 Card Read Punch, and the 1624 Paper Tape Punch. The typewriter prints at the rate of 10 characters per second. The 1622 punches data into 80-column IBM cards at the rate of 125 cards per minute. The 1624 punches data into eight-channel paper tape at the rate of 15 characters per second.

When the computer is reading from or writing on an input-output device and is in the numerical mode, each character received from or sent to the input-output device is represented in memory as a single digit and occupies one memory position. In the numerical mode, data must consist of numerical characters only. When the computer is reading from or writing on an input-output device and is in the alphameric mode, each character received from or sent to the input-output device is represented in memory as two digits occupying two adjacent memory positions. In the alphameric mode, data may consist of numerical, alphabetic, and special characters. The digits for numerical characters, in the alphameric mode, consist of an arbitrarily assigned digit (zone digit) and a second digit (numerical digit) representing the true decimal value of the numerical character. Alphabetic and special characters are represented by two arbitrarily assigned digits.

Figures 1-1 and 1-2 illustrate all characters and their assigned digital values for both numerical and alphabetic modes.

Machine Language

Character Code

All data stored, transferred, and processed within the computer is represented as decimal digits in binary coded decimal (BCD) form. A digit is represented by a particular combination of bits. The bit positions of each digit consist of four numerical bits (8, 4, 2, and 1), one flag (F) bit, and one check (C) bit (see Figure 1-3). The value of a significant digit is the sum represented by the bits present in the 8, 4, 2, and 1 numerical bit positions. A zero is represented by a C bit alone. Only bit combinations whose sum is nine or less are used. Considering only the numerical bit positions, the digit 6 is represented by a 4 bit and a 2 bit; the digit 7 by a 4 bit, a 2 bit, and a 1 bit; and the digit 8 by an 8 bit alone (see Figure 1-4).

The flag (F) bit is used in three ways. The presence or absence of the flag bit in the units position of a numerical field determines the sign of the field. Absence of the flag bit is interpreted as a field signed plus. The presence of the flag bit is interpreted as a field signed minus. The flag bit is also used as a field mark defining the high-order position of a numerical field. A flag bit with a digit of the add table indicates a carry in arithmetic operations.

The C bit is used for parity checking purposes. Each digit within the computer must consist of an odd total number of bits, or a parity error will be indicated. The C bit will be present in a digit when the number of bits present in the numerical bit positions and the F bit position consists of an even number of bits.

A special combination consisting of the C bit, 8 bit, and 2 bit is used as a record mark.

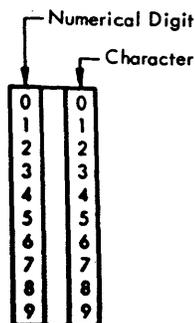


Figure 1-1. Numerical Code

Instruction Format

The IBM 1620 uses a 12-digit instruction divided into three parts; a two-digit operation code, a five-digit "P part", and a five digit "Q part" (see Figure 1-5).

The operation codes consist of two digits (00 through 99) which specify the operation to be performed. Figure 1-6 is a chart of operation codes and their associated mnemonics.

Zone Digit	Numerical Digit	Character
00		b
03		.
04)
10		+
13		\$
14		*
20		-
21		/
23		,
24		(
33		=
34		@
41		A
42		B
43		C
44		D
45		E
46		F
47		G
48		H
49		I
50		0
51		J
52		K
53		L
54		M
55		N
56		O
57		P
58		Q
59		R
62		S
63		T
64		U
65		V
66		W
67		X
68		Y
69		Z
70		0
71		1
72		2
73		3
74		4
75		5
76		6
77		7
78		8
79		9

Figure 1-2. Alphameric and Special Character Codes

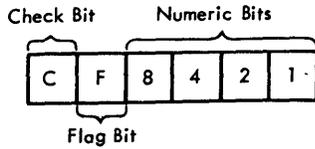


Figure 1-3. Bit Positions

The functions of the P and Q parts of an instruction are dependent upon the particular operation to be performed. The P part of an instruction can represent the address in memory of a digit, a field, a record, or another instruction. The Q part can represent the address in memory of a digit, field, or record; a data field itself; the "address" of an input-output device; the control function to be performed by an input-output device together with the "address" of the device; or the code for a switch or indicator. The specific use of the P and Q parts will be discussed with each operation.

Data Format

Data is stored in memory to form fields or records that can be of any length within the capacity of memory. Data can be classified as digits, fields or records depending upon how they are addressed and the limiting factor of the data.

A digit occupies one memory position and is addressed individually.

Fields in memory consist of a number of consecutive digits and are composed of data related to arithmetic operations and internal field transmission. A field is addressed by its rightmost (low-order) position which occupies the highest numbered memory position of the field. Fields are processed from right to left into suc-

	C	F	8	4	2	1
0	X					
1						X
2					X	
3	X				X	X
4				X		
5	X			X		X
6	X			X	X	
7				X	X	X
8			X			
9	X		X			X

Figure 1-4. Bit Configuration—Decimal Digits 0-9

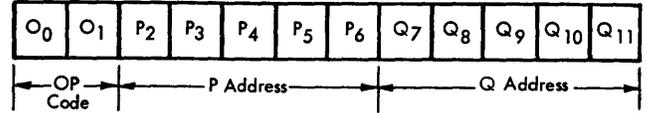


Figure 1-5. Instruction Format

cessively lower memory positions until a digit with a flag bit is sensed. The shortest admissible field consists of two digits, the addressed digit and the adjacent digit containing the field definition flag bit. Numerical fields are signed minus by a flag bit present in the units position. The absence of a flag bit in the units position is interpreted as a field signed plus.

Record in memory consists of a field or fields of data related to input-output operations and internal record transmission. A record in memory is addressed at the leftmost (high-order) position, which occupies the lowest numbered memory position of the record. Records are processed serially from left to right into successively higher memory positions. Output and internal record transmission operations are terminated whenever a record mark is sensed. For memory to receive a record, data is entered starting at the addressed position and continuing from left to right into successively higher memory locations until terminated by an end-of-record signal from the input unit.

Stored Program Concept

To solve a problem or to process data, a programmer selects from the different operations which the system is capable of performing, those which are required to accomplish the desired results. The series of instructions which designate the operations to be performed is called a program. Because the instructions comprising a program are written into memory from an input device and read from memory for interpretation and execution, the 1620 is called a stored program computer.

For interpretation by the computer, an instruction must be read from memory to registers, starting with the high-order digit and continuing through successively higher memory locations until all twelve digits have been read.

Instructions within a program are normally interpreted and executed sequentially; that is, execution of the first instruction is followed by interpretation and execution of the second instruction, etc. (see Figure 1-7). However, this sequence can be altered by the use of conditional branch instructions that direct the computer to an instruction located at other than the next sequential position.

Arithmetic Instructions:	MNEMONIC	CODE
Add	A	21
Add Immediate	AM	11
Subtract	S	22
Subtract Immediate	SM	12
Compare	C	24
Compare Immediate	CM	14
Multiply	M	23
Multiply Immediate	MM	13
Load Dividend	LD	28
Load Dividend Immediate	LDM	18
Divide	D	29
Divide Immediate	DM	19
Internal Data Transmission Instructions:		
Transmit Digit	TD	25
Transmit Digit Immediate	TDM	15
Transmit Field	TF	26
Transmit Field Immediate	TFM	16
Transmit Record	TR	31
* Transfer Numerical Strip	TNS	72
* Transfer Numerical Fill	TNF	73
Branch Instructions:		
Branch	B	49
Branch No Flag	BNF	44
Branch No Record Mark	BNR	45
Branch On Digit	BD	43
Branch Indicator	BI	46
Branch No Indicator	BNI	47
Branch and Transmit	BT	27
Branch and Transmit Immediate	BTM	17
Branch Back	BB	42
Input-Output Instructions:		
Read Numerically	RN	36
Write Numerically	WN	38
Dump Numerically	DN	35
Read Alphanumerically	RA	37
Write Alphanumerically	WA	39
Control	K	34
Miscellaneous Instructions:		
Set Flag	SF	32
Clear Flag	CF	33
* Move Flag	MF	71
Halt	H	48
No Operation	NOP	41
* Additional Feature		

Figure 1-6. Operation Codes and Mnemonics

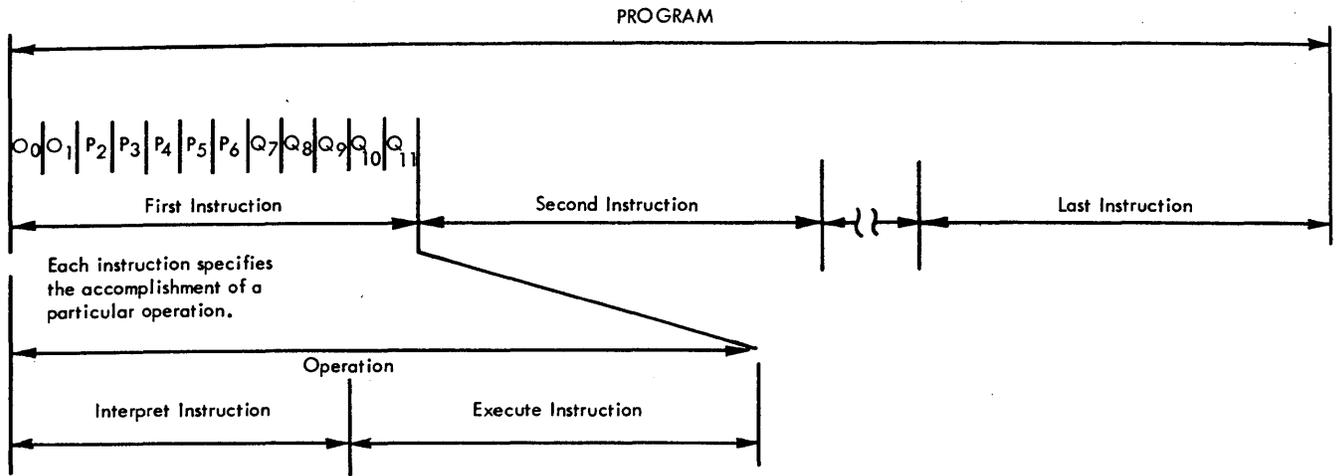


Figure 1-7. Program

Additional instructions and/or data may be inserted into memory from an input device during the solution of a problem.

The only distinction between instructions and data, in memory, is the manner in which they are interpreted by the computer. If for any reason data was placed in memory locations assigned to instructions, the data would be acted upon as instructions. Conversely, the operation code or either address part of

an instruction may be modified by treating the instruction parts as data. Flag bits may be stored with the digits of an instruction to permit operation on the instruction as data. The flag bits are ignored during interpretation of the instruction except when Indirect Addressing, an additional feature, is installed in the 1620. The functions of Indirect Addressing are given in the *IBM Customer Engineering Manual of Instruction, 1620 Data Processing System, Additional Features* (Form 227-5513).

Section 2 Memory Addressing and Data Flow

Introduction

The IBM 1620 Computer normally processes data serially by digit, parallel by bit. However, the two digits which are read from memory for each address, are processed simultaneously under certain conditions.

Data, as such, is confined to memory, memory buffer register (MBR), memory data register (MDR), and input-output areas. Digits of data which are presented to other registers by MBR and/or MDR are used to develop memory addresses or to control program execution.

Each instruction of a program is interpreted and executed by a series of machine cycles. A machine cycle consists essentially of: addressing memory, reading out of a memory location, and writing into a memory location. Digits read from memory, that are to be used for addressing or control purposes, are set into registers from MBR and/or MDR within the cycle. Digits are presented to an output device or received from an input device within the machine cycle.

NOTE: The original design concepts of this system made extensive use of triggers. Therefore the functional nomenclature chosen included the term trigger (trigger 21, first cycle trigger, I-cycle trigger). Subsequent Engineering changes have replaced all triggers with latches except in four cases. The True/Complement, High/Plus, Clock drive, and A/B triggers are still binary connected triggers. The system diagrams still maintain the original "trigger" designation for many of the latches and line names. For this reason the terms trigger and latch are generally used synonymously throughout this manual. These terms are not intended to designate the actual type of component used. The functions of SMS card types used in the 1620 system are given in Section 4 of this manual.

Refer to Figure 2-1 for the following discussion of data flow.

Memory

Memory is addressed by means of two matrix switches which select one "column" of 12 cores (one core in the same position in each plane). Since each digit position in memory consists of six bits (C, F, 8, 4, 2, 1), it follows that in selecting one column of 12 cores, two

digits will be read out each time memory is addressed. Addressing is in the form of five digits (00000 to 19999) and is from a single register, memory address register (MAR). Memory is divided into two sections, even and odd, by the wiring scheme. If memory is addressed per 00000, the digit at 00000 and at 00001 will be read out (see Figure 2-2). If the address is 00001 the digit at 00001 and the digit at 00000 will be read out (see Figure 2-3).

Sense Amplifiers and Memory Buffer Register (MBR)

Digits which are read out of memory are sensed by sense amplifiers which in turn set latches in the memory buffer register (MBR). The MBR is designated even (MBR-even) and odd (MBR-odd) and will receive the corresponding digit from memory (see Figure 2-4).

The read out of cores is destructive and therefore to retain the digits they must be written back into memory. The latches in MBR control inhibit lines which in turn permit the digits to be written back into memory (see Figure 2-5).

Memory Data Register (MDR)

Generally the 1620 System will use only the specific digit addressed. Therefore, a single digit register called memory data register (MDR) is provided to receive and store the digit. Controls are provided to determine which digit in MBR (odd or even) is transferred to MDR. To accomplish this, the output of MBR-odd and even is controlled by an odd/even trigger which in turn is controlled by the units position of MAR. Figure 2-6 shows odd/even trigger control of the data path for the digit 5 from memory to MDR. The digit is available to the data bus from MDR.

There are times when it is necessary to clear a location in memory. At this time the odd/even trigger exercises control, as to which specific location (odd or even) is to be cleared, by blocking the odd or even sense amplifiers (see Figure 2-7). There are functions requiring the clearing of two memory locations both odd and even at the same time. This is done by blocking both the odd and the even sense amplifiers without regard to the status of the odd/even trigger. The

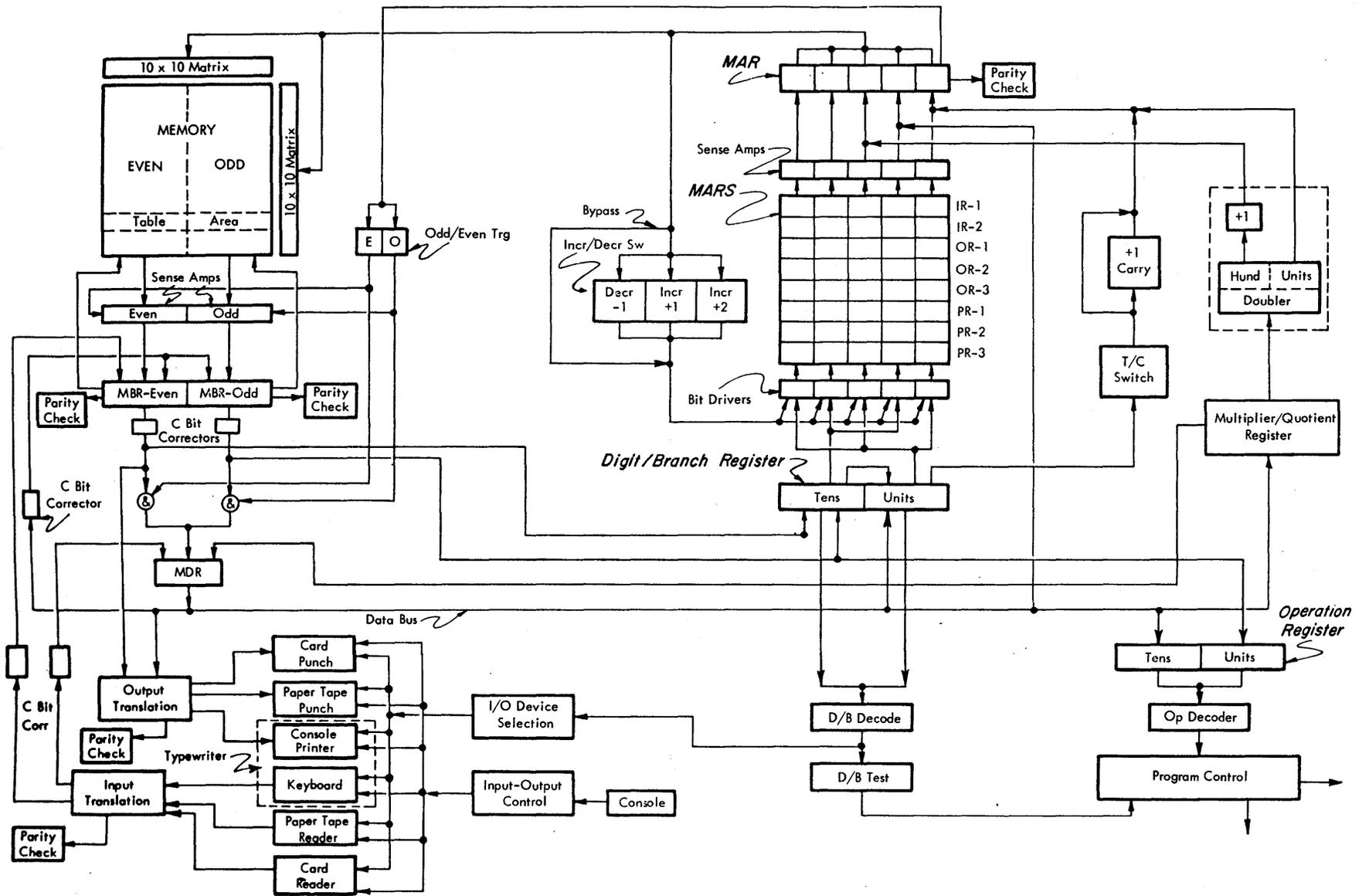


Figure 2-1. Data Flow

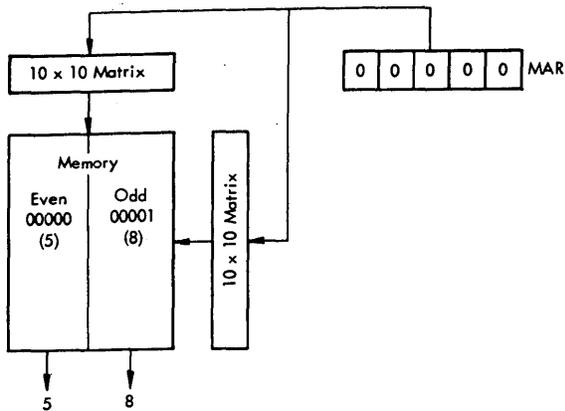


Figure 2-2. Memory Addressed per 00000

objective of clearing a memory location is to reset the cores at that address so that a new digit may be entered. The new digit will be placed in MBR. MBR controls the inhibit lines which permit writing the new digit into memory.

Memory Address Register Storage (MARS)

Memory address register storage (MARS) is a single plane core array in a 24 x 16 matrix. The basic 1620 system makes use of only a part of the available storage, using a 24 x 8 matrix. MARS is divided into eight sections, each section capable of storing a 5 digit address. Each section or register has its own read and write drivers so that selection may be made as to which register is to read into the memory address register (MAR) and which one (or more) will be written into (see Figure 2-8).

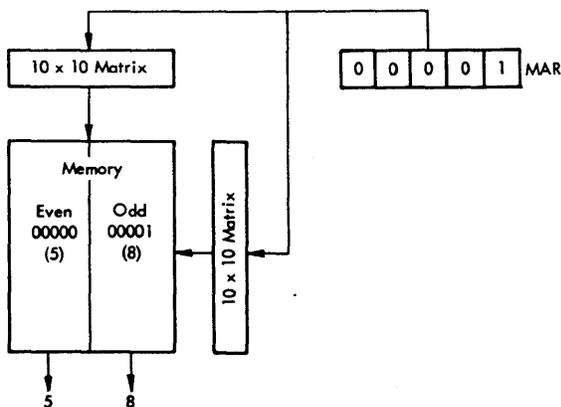


Figure 2-3. Memory Addressed per 00001

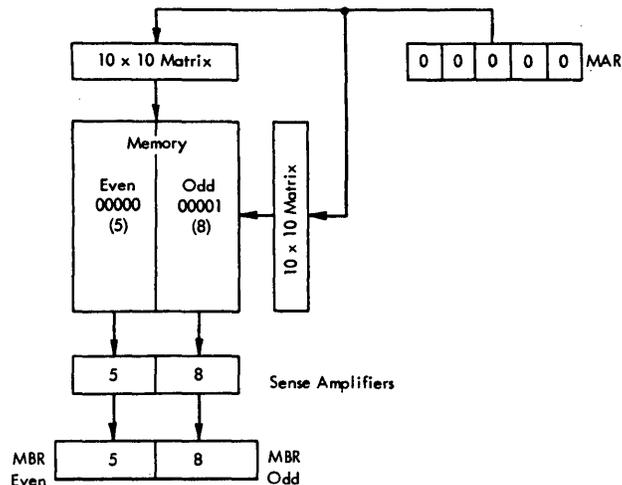


Figure 2-4. Reading Out of Memory

In order to address memory, an address must be placed in MAR. This is done by causing the read drivers of a particular MARS register to be operative. The read drivers cause the address stored in that MARS register to read out, through sense amplifiers, to MAR.

To reset a MARS register the read drivers for that register are made operative at a time when the sense amplifiers are not conditioned to function.

In order to write into a MARS register, coincidence of the MARS write drivers for that register and the bit drivers is required.

The bit drivers can be gated so that a complete 5-digit address is written into MARS at one time. Another gate can cause 1 or 2 digits, as needed, to be written into MARS.

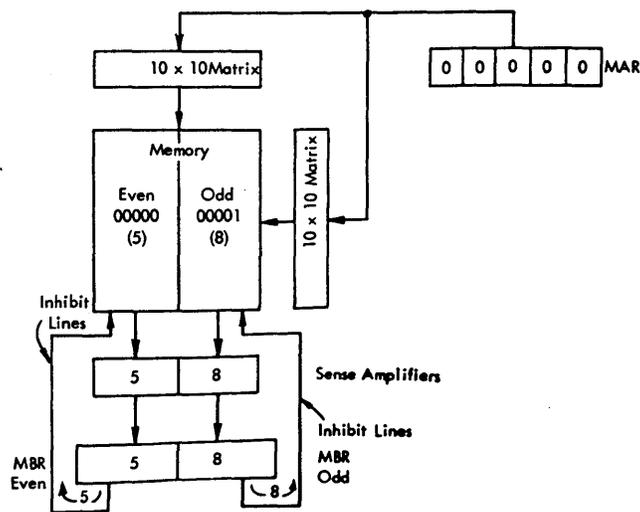


Figure 2-5. Writing Into Memory

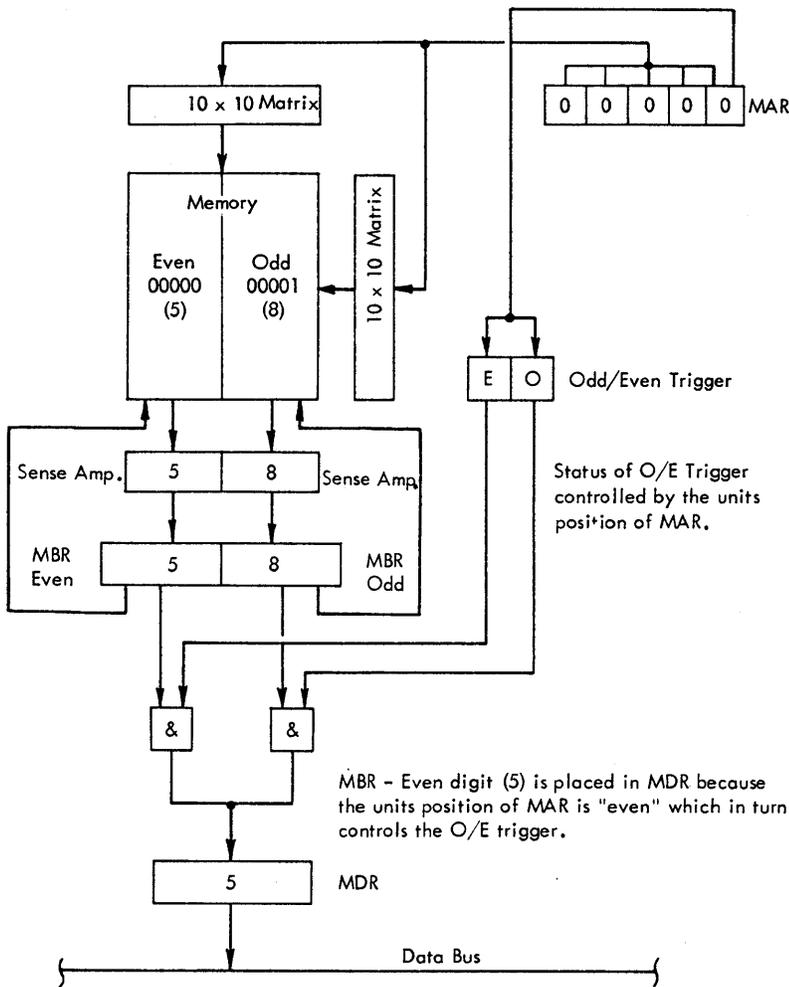


Figure 2-6. MBR to MDR Data Flow with Odd/Even Trigger Control

Increment/Decrement Switch

Each address read into MAR from MARS is the address of a single location in memory. Since a field or record consists of more than one digit, a means is provided to increase the address by one (increment + 1) or decrease the address by one (decrement - 1) depending on the requirements of the operation being performed. There are operations which make use of two digits at a time for which an increment +2 is provided. A decrement of -2 is provided with the Indirect Addressing feature. The increment/decrement switch (incr/decr switch) consisting of a network of AND switches and OR switches performs these functions. The de-

crement trigger exercises control over the incr/decr switch. The status of the decrement trigger is determined by the instruction to be executed. The outputs of all the MAR bit triggers are connected to the inputs of the increment/decrement switch. The outputs of the incr/decr switch are connected to the bit drivers. The coincidence of the bit drivers and the write drivers for a particular MARS register will write the MAR address (as altered by the incr/decr switch) into the MARS register.

There are operations requiring the read out of MARS to MAR where it is desired to write the address back into MARS unchanged. A bypass around the incr/decr switch accomplishes this function (see Figure 2-9).

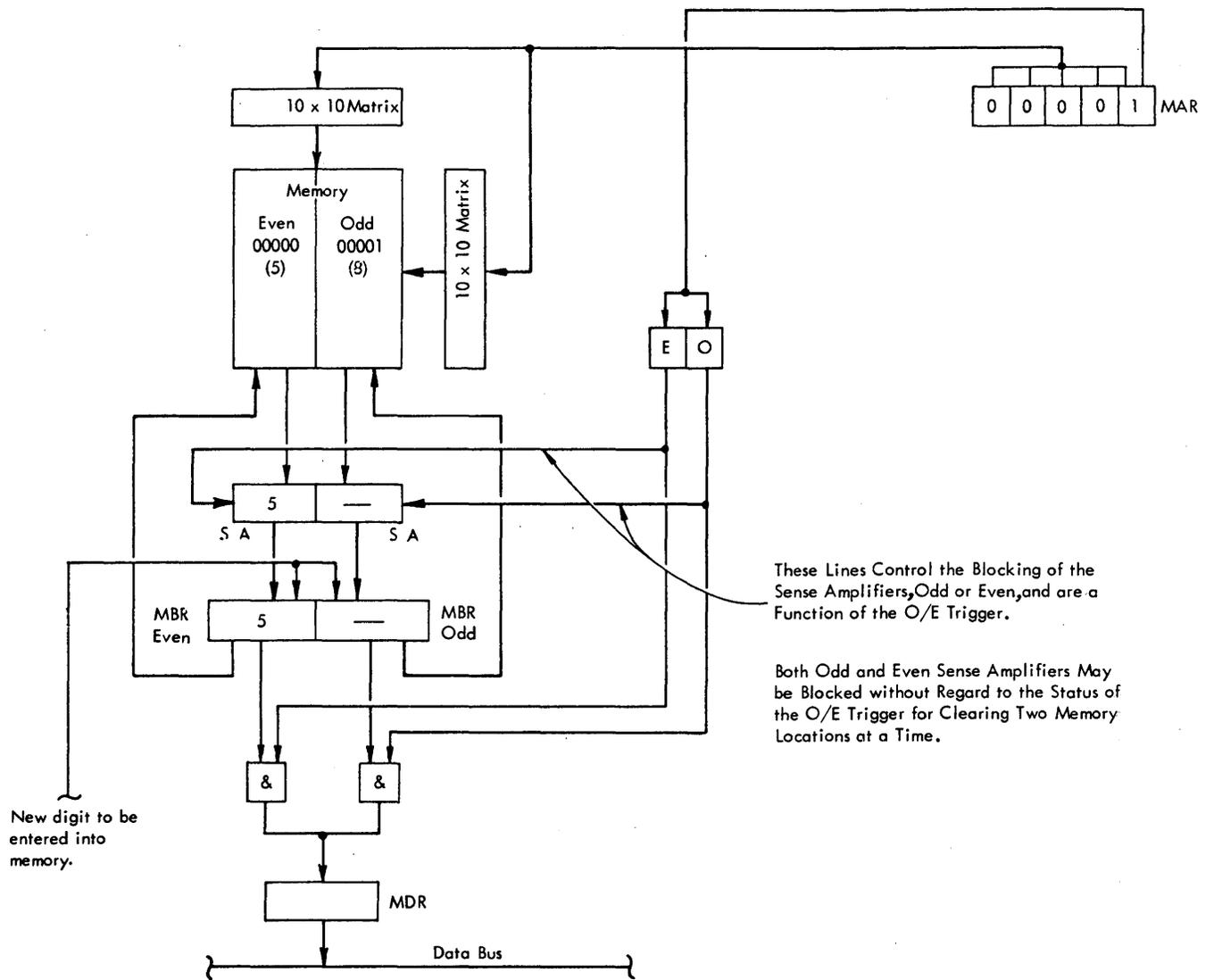


Figure 2-7. Clearing Memory Locations

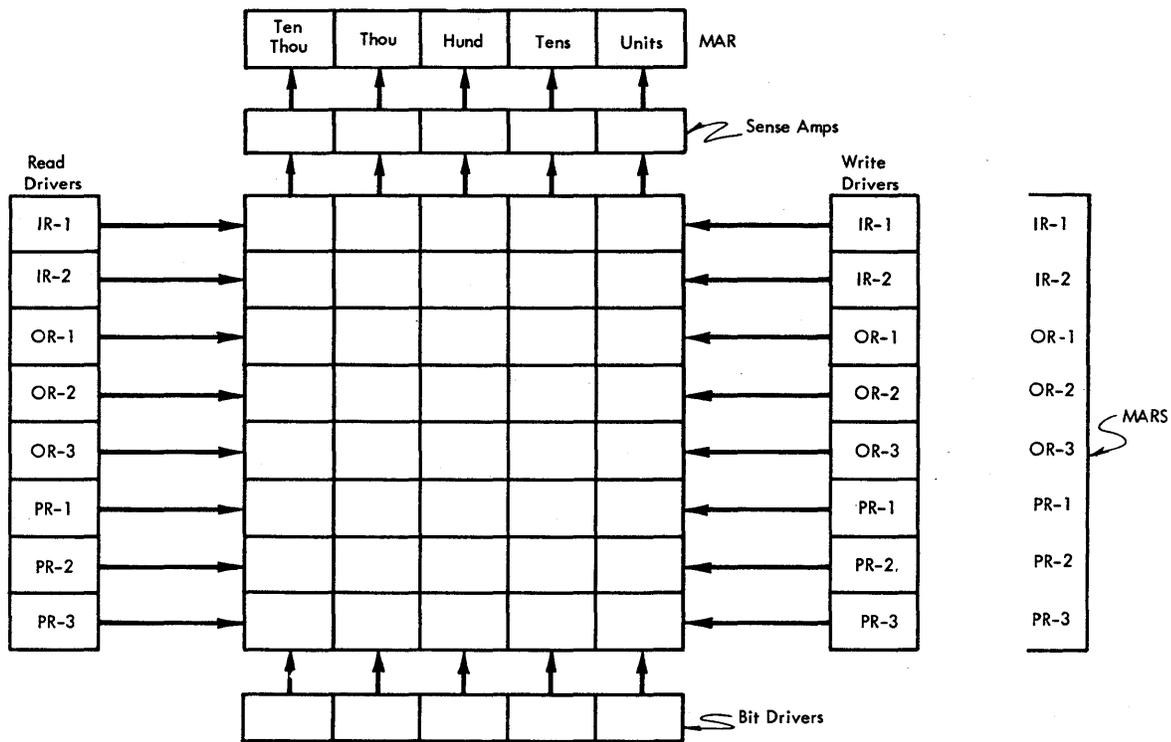


Figure 2-8. Memory Address Register Storage

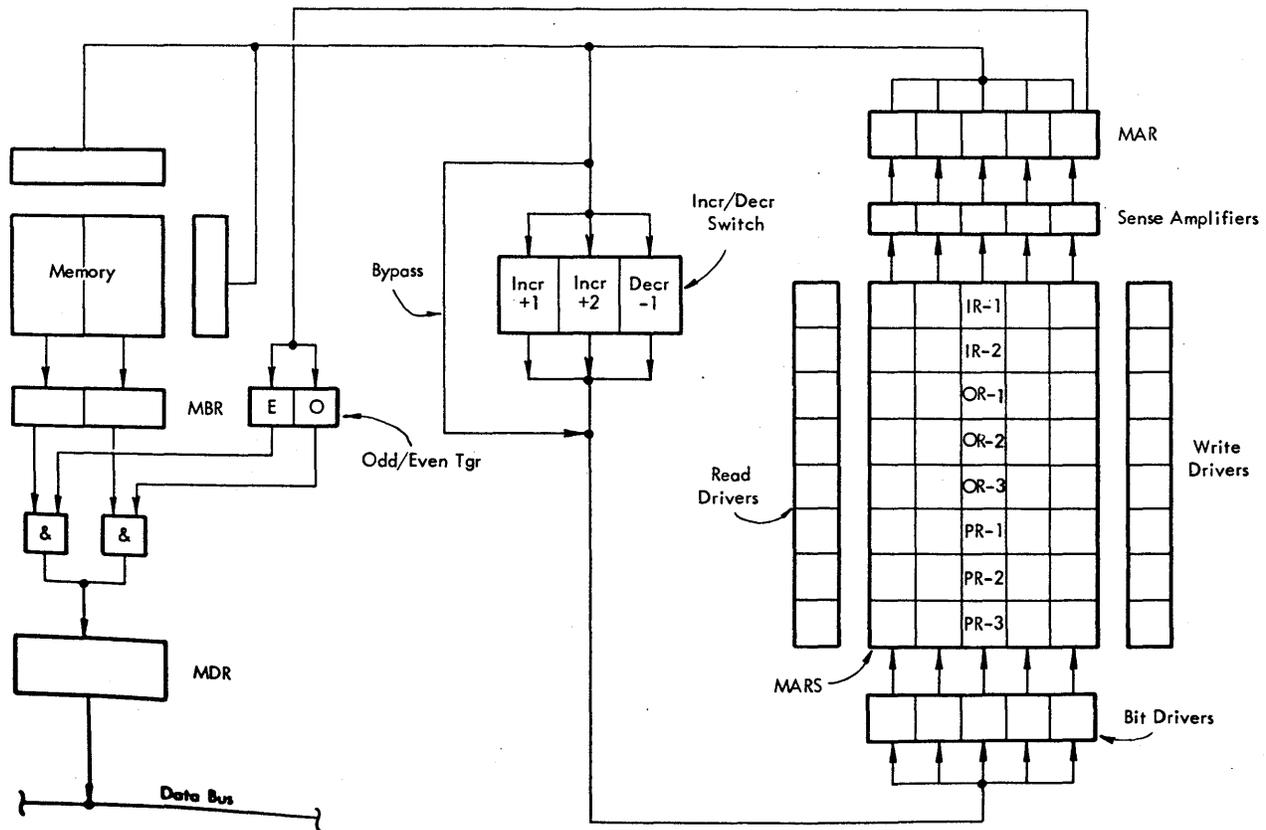


Figure 2-9. Increment/Decrement Switch

Development of a Program

Programming consists of defining the steps required to receive data, process data, and record results in terms of the operations which the computer system is capable of performing. Each step must be written as an instruction to the computer, with the series of instructions pertaining to an entire procedure constituting a program.

A program to accomplish entry of factors, solution of a simple problem, and printing of the result is presented to illustrate 1620 Computer programming (see Figures 3-1 and 3-2).

COMPUTE A + B = C

Specifications:

1. Enter factors A and B into memory by means of the typewriter keyboard.
2. Store the units position of factor A at memory location 00500. (If the assigned field length for factor A is assumed to be five digits, the high-order position will be stored at location 00496.)
3. Store a record mark character at memory location 00501. (This record mark will be used to terminate the write operation demanded by specification 5.)
4. Store the units position of factor B at memory location 00800. (If the assigned field length for factor B is assumed to be four digits, the high-order position will be stored at location 00797.)
5. Record the sum, C, on the paper form in the typewriter. (The developed sum will replace factor A and assume its location in memory.)
6. Stop the computer upon completion of the program.

Insertion of a Program Into Memory (Typewriter Keyboard)

Because the IBM 1620 is a stored program computer, the instructions comprising a program must be written into memory for availability to the computer in the accomplishment of the program objectives.

The program is stored in memory by depressing INSERT on the console and then by typing each digit of the five instructions consecutively.

Depression of INSERT places the typewriter in numerical shift; activates the typewriter keyboard; and pre-

pares the computer to receive the high-order digit of the first 12-digit instruction at memory location 00000. By consecutive depression of the 3, 6, 0, 0, 4, 9, 6, 0, 0, 1, 0, 0, 3, 6, 0, 0, 7, 9, 7, 0, 0, 1, 0, 0, 2, 1, 0, etc. keys on the typewriter keyboard, the program is stored in memory locations 00000 through 00059. (In the program chart where the letters "P" or "Q" appear in the five instructions, any numerical key may be depressed. Those digit positions must be filled, but they are not used in the execution of the particular instruction.)

After entry of the low-order digit of the fifth instruction, depression of RELEASE on the console terminates the insert operation. Depression of START then initiates execution of the stored program. Depression of the R-S key on the typewriter keyboard performs the functions of RELEASE followed by START.

Execution of a Program

Instructions within a program are normally interpreted and executed sequentially: that is, if the first 12-digit instruction is stored in memory locations 00000 through 00011, the execution of that instruction is followed by the interpretation and execution of the instruction stored in locations 00012 through 00023, etc.

Each instruction is read from memory and stored in registers for interpretation, starting with the high-order digit and continuing through successively higher memory locations until all twelve digits have been read.

Since execution of the first instruction, 36 00496 00100, requires that factor A and a record mark character be entered into memory manually from the typewriter keyboard, the $\bar{\quad}$ (flag), 0, 0, 3, 2, 1, and record mark keys are depressed consecutively by the operator. Depression of RELEASE then terminates execution of the first instruction.

Depression of START directs the computer to proceed with the program and the second instruction is interpreted.

Execution of the second instruction, 36 00797 00100, requires that factor B be entered into memory manually from the typewriter keyboard. The $\bar{\quad}$ (flag), 0, 0, 6, and 7 keys are depressed consecutively by the operator. Depression of RELEASE terminates execution of the second instruction.

Instruction Number	Memory Location for Instruction	Instruction			Function to be Executed
		Op Code	P-Part	Q-Part	
One	00000	36	00496	Q01QQ	Enter Factor A (00321) at Memory Location 00496 - 00500. Enter Record Mark Character at 00501
Two	00012	36	00797	Q01QQ	Enter Factor B (0067) at Memory Locations 00797 - 00800.
Three	00024	21	00500	00800	Add A + B and Store Sum, C, at Memory Locations 00496 - 00500.
Four	00036	38	00496	Q01QQ	Type Sum, C, (00388) on Paper Form in Typewriter from Memory Locations 00496 - 00500.
Five	00048	48	PPPPP	QQQQQ	Stop Computer

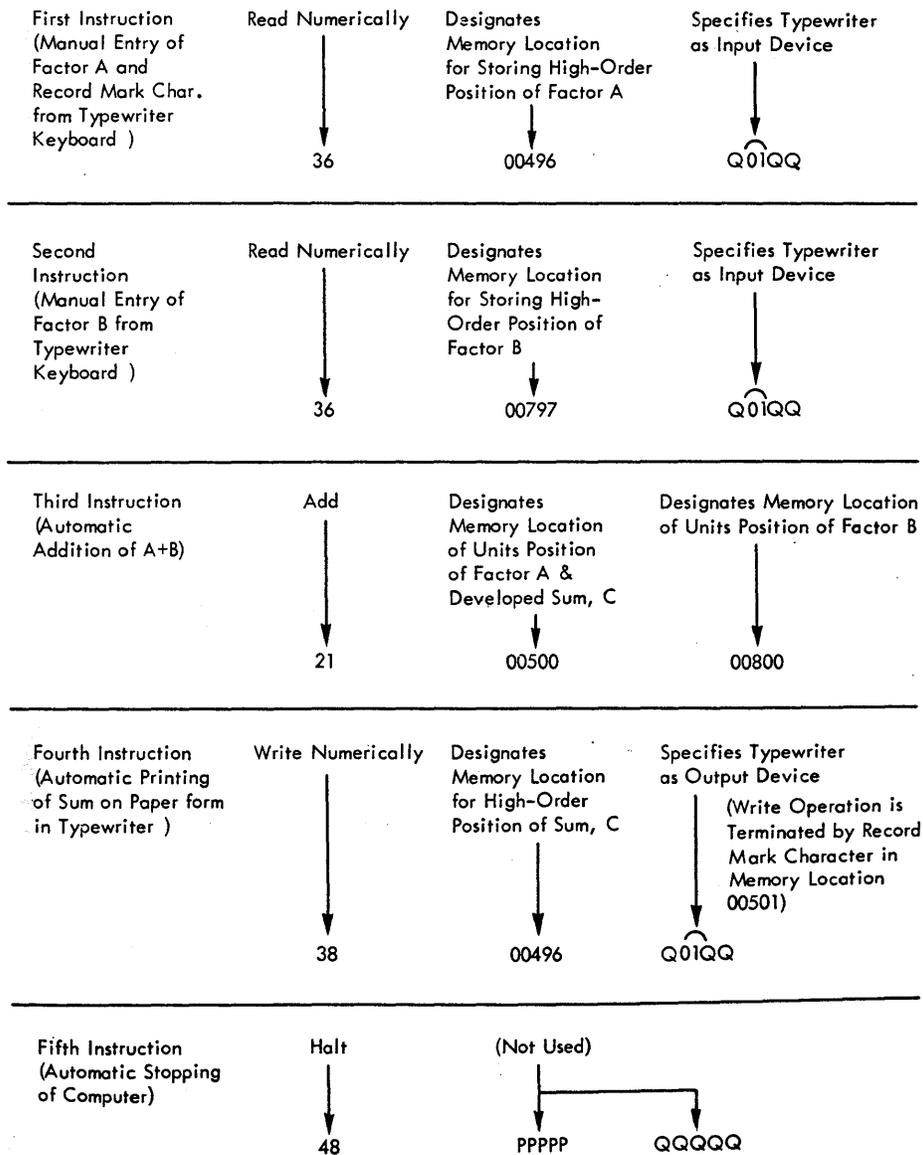


Figure 3-1. Program "A"

Instruction Number	Memory Location for Instruction	Instruction			Function to be Executed
		Op Code	P-Part	Q-Part	
One	15000	36	00496	Q01QQ	Enter Factor A (00321) at Memory Location 00496 - 00500. Enter Record Mark Character at 00501
Two	15012	36	00797	Q01QQ	Enter Factor B (0067) at Memory Locations 00797 - 00800.
Three	15024	21	00500	00800	Add A + B and Store Sum, C, at Memory Locations 00496 - 00500.
Four	15036	38	00496	Q02QQ	Punch Sum, C, into Paper Tape from Memory Locations 00496 - 00500.
Five	15048	48	PPPPP	QQQQQ	Stop Computer

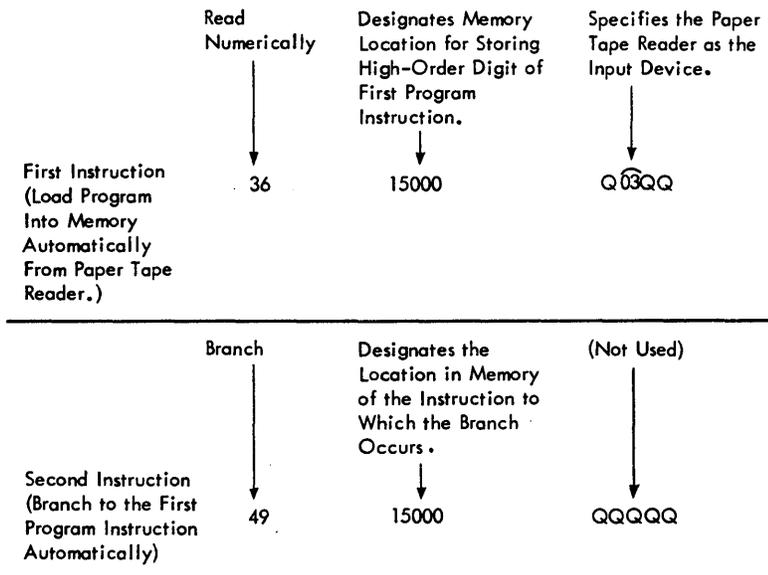


Figure 3-2. Program "B"

Depression of START again directs the computer to proceed with the program and the third instruction is interpreted. No further manual entry of data is required by the program, so the third instruction is executed and the two remaining instructions are interpreted and executed automatically. The computer halts upon completion of the program.

Automatic Program Repetition (Program Loop)

The program shown in Figure 3-1 computes the sum for only one problem, but by returning to the first instruction of the program and repeating the program as a loop, the sums for any number of problems can be computed. A program loop will be initiated if the halt instruction, 48 PPPPP QQQQQ, is replaced by

the branch instruction, 49 00000 QQQQQ. The branch instruction directs the computer to the instruction located at its P address, 00000, which is the address of the first instruction in the program. (A typewriter control instruction, 34 PPPPP Q01Q1 or 34 PPPPP Q01Q2, could be inserted into the program to cause the typewriter to line space and carriage return or simply space between sums.)

Insertion of a Program Into Memory (Paper Tape Reader)

Where it is required that a program be loaded into memory frequently or where a program contains many instruction steps, the program may be punched in paper tape and stored in memory from the paper tape reader.

The procedure required to load a program, Figure 3-2, into memory by means of the paper tape reader is described in the following paragraphs.

The program shown in Figure 3-2, differs from that shown in Figure 3-1 only as follows:

1. The "02" in the Q part of the fourth instruction specifies the IBM 1624 Paper Tape Punch as the output device rather than the typewriter.
2. The program is stored in memory locations 15000 through 15059 rather than 00000 through 00059.

Condition the IBM 1620 Data Processing System for this operation as follows:

1. The digits of the five instructions are punched in consecutive paper tape columns with an "end-of-line" character punched in the following tape column. (The end-of-line character is used to terminate the program loading operation.)

2. The paper tape containing the program is loaded on the paper tape reader with the reader in a condition ready to operate.
3. The paper tape punch is loaded with blank tape and is ready to operate.

Insertion from the typewriter keyboard of the two "starting" instructions, shown in Figure 3-2, into memory at locations 00000 through 00023 is required to cause loading of the program and subsequent branching to the program.

The two "starting" instructions are placed in memory at locations 00000 through 00023 by depressing INSERT and typing consecutively the 24 digits of the instructions. Depression of RELEASE then terminates the insert operation. Upon depression of START, the "starting" instructions are interpreted and executed, and the program is entered automatically at completion of the second "starting" instruction.

Magnetic Cores

Memory Cores

A magnetic core is a small doughnut shaped ring that is uniformly constructed of ferrite particles bonded together by a ceramic material. The ferrite particles have good magnetic properties and the core has a high retentivity of the magnetic flux lines after the magnetizing force is removed. It is this property of retentivity that makes a memory core useful as a storage device.

The operation of a memory core can best be described by reference to the hysteresis curve, Figure 4-1. This curve is a plot of the relationship between a magnetizing current (I_m) and the flux density.

A memory core is capable of maintaining indefinitely one of two stable magnetic states, either at point A or at point D on the hysteresis curve. Because the core has two stable states, it can be used as a binary storage device. At point A the core has a residual flux in a negative direction, and at point D a residual flux in the positive direction. These two directions can be arbitrarily assigned as binary "zero" and binary "one", respectively.

I_m is the amount of current necessary to change the state of the core. Plus I_m is that amount of current flowing in one direction, and minus I_m the same amount of current flowing in the opposite direction.

On the hysteresis curve, it can be observed that a magnetizing current of plus I_m will change the magnetism of the core from point A, a binary zero, to the magnetic saturation value in the positive direction at point C. When the current is removed, the total amount of magnetization drops back to point D (binary one). If, instead of full plus I_m , a current of plus $I_m/2$ were applied, the flux would change only the small amount from point A to point B on the curve, and when the current returned to zero, the flux would return to its original value at point A.

A reverse current, minus I_m , develops flux of opposite polarity and changes the magnetic field of the core from point D to the magnetic saturation value in the negative direction at point F. The total amount of magnetization drops back to point A (binary zero) when the driving current is removed.

When a matrix of memory cores is constructed to store multiple bits of information, a specific core is selected (addressed) by the coincidence of $I_m/2$ flowing through each of two wires threaded through the core, with a total effective current of full I_m . The state of the core is determined by the direction of the current flowing through these wires. A current I_m is passed to store a "1" in the core; this is called "writing" into cores. A current I_m is passed in the opposite direction to change a stored value of "1" to a "0" and can be considered a current of minus I_m ; this is called "reading" a core.

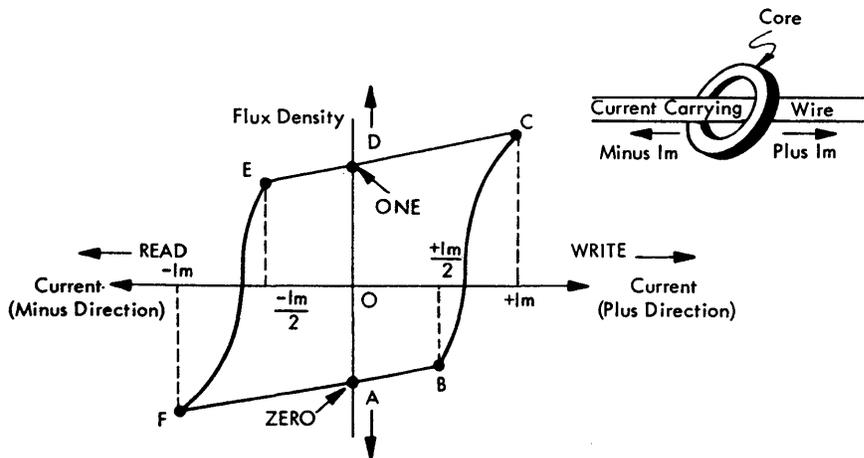


Figure 4-1. Hysteresis Curve

Reading a memory core depends on a system of sensing the state of the flux field within the core. When the core contains a "0" and a current of minus I_m read current is passed through the core, the field changes from point A to point F on the hysteresis curve, which is a very small change in total flux density. If the core contains a "1" and a minus I_m read current is passed through the core, the field changes from point D to point F and a large change in the flux field occurs. A third wire, called the "sense" wire, is threaded through the core to recognize these changes in the magnetic field. Circuits are used to discriminate against low value "0" signals and amplify the large signal that results when a "1" is read from a core.

Switch Cores

The construction of the switch cores is similar to memory cores, but the switch cores are physically larger. Switch cores, however, are not used to store or retain information, but are used as transformers, making use of the induced current pulses when the core is flipped. The induced current pulses are used to read into or write from memory cores.

The memory addressing scheme selects two specific switch cores. The outputs of the two switch cores are used to establish X and Y coordinate lines for addressing a memory location.

Component Circuits

Discussion of electronic or transistor theory is not intended in this manual. Other IBM manuals have been written to provide such information. Samples of the various types of SMS cards are discussed to provide the Customer Engineer with sufficient information to interpret *System Diagrams* and understand machine functions.

The SMS Card Index, Figures 4-2a and 4-2b, lists the card types by card code used presently and formerly in the 1620 Computer system. Circuit schematics, part numbers, and associated logic block symbols for the cards are shown on pages C.00.03.1 through C.00.69.1 of the *1620 System Diagrams*.

The 1620 Computer uses a kind of circuitry known as **TRL**, Transistor Resistor Logic. This type of circuit uses resistors in voltage dividing networks to control the functions of transistors. **TRL** is also called **NOR**, Negative OR.

There are two types of transistors used in the 1620: alloy and drift. The drift type is also called saturating drift. The term **SDTRL**, used with an SMS card, means Saturating Drift Transistor Resistor Logic and indicates that the transistor or transistors used in that

card are of the drift type. The difference between alloy and drift transistors is in transition times. The drift is faster than the alloy and is used in areas where speeds are critical.

The term **CTRL** means Complementary Transistor Resistor Logic which indicates that inputs and outputs can be of a different level. For example a $\pm S$ level input can result in a $\pm R$ level output. **R** levels are $+R = +12v$ and $-R = 0v$. The relay thyatron drivers, **DP** cards, are an example, having $+48v$ on the output. **CTRL** cards are not necessarily used with the complementary function.

Figure 4-3 depicts a typical "block," as shown in *1620 Automated System Diagrams*, with symbols defined.

AHK, CAB, CD, DAX, DAW, MX, VE, and VF Cards

These cards consist of **NOR** circuits which are similar in function and operation. The following description of one of the three circuits on a type **VF** 3-way inverter card (Figure 4-4) will serve to explain the function and operation for the group.

As an Inverter, this block has a $+S$ output level if any one of its inputs is at a $-S$ level.

As an **AND** switch, this block has a $-S$ output level if all inputs are at $+S$ level.

As an **OR** switch, this block has a $+S$ output level if any one or more of its inputs is at $-S$ level.

The transistor is conducting when any one or more of the inputs is at $-S$ level.

EXTENDERS

When additional inputs are necessary to perform a particular logical function, **NOR** blocks are connected in parallel. One and only one of the logic blocks has a collector load resistor. This configuration merely "extends" the logical function (Figure 4-5). All input lines to extender logic blocks as well as the basic logic block must be at $+S$ level to obtain a $-S$ output level. One or more $-S$ input levels to an extender block as well as to the basic logic block will give a $+S$ output level.

NOR LATCH

A common method of storing a bit of information in **NOR** logic is by means of a "latch" (Figure 4-6).

A shift to the $-S$ level at point F or point G places the **OFF** side transistor into conduction, which puts point CL at $+S$ level. If points A and B are at $+S$ level, the **ON** side transistor is cut **OFF**, which drops point E to the $-S$ level. A $-S$ level at point E reinforces the $-S$ level introduced at point F or point G and holds point CL at a $+S$ level even if the negative signal at point F or point G is removed. This latches the circuit in an **OFF** state until a negative ($-S$) signal occurs at either point A or point B.

Card Code	Symbol	Systems Diagram	Description
ACY	IS	C.00.03.1	Constant Current Source – used in 1622 buffer (04.30.06.1)
AFB YJA YJB	R	C.00.05.1	Load Resistor Cards (01.30.80.1)
AFR	LD or ID	C.00.05.1	Lamp Driver – a -S input level causes a +S output level (01.06.10.1)
AHK	A, O, or I	C.00.06.1	Two way alloy type extender for AND/OR, or OR latches (01.06.10.1)
AHS	DCM	C.00.07.1	Clutch Magnet Driver (04.15.06.1)
AHT	DPM	C.00.07.1	Punch Magnet Driver – requires a CB input to the L pin (04.82.03.1)
AHU	∇	C.00.08.1	Exclusive OR – like inputs produce a +S out and unlike inputs produce -S out (01.40.20.1).
CAB	E	C.00.10.1	Three way alloy extender card – collectors not internally connected to load resistor (01.06.11.1).
CD	A or O	C.00.10.1	Three way alloy AND/OR card (01.06.10.1).
CAU	AL	C.00.12.1	AND latch – alloy type (01.06.11.1)
CE	DE	C.00.15.1	Emitter follower – alloy (01.06.20.1).
DAR	DP	C.00.15.1	Emitter follower – drift (04.83.08.1)
DAT	DP	C.00.17.1	Heavy Power Inverter – used for 1622 reset lines (04.05.07.1).
DAW	E	C.00.20.1	Three way drift extender card – similar to CAB (01.40.03.1).
DAX	E	C.00.20.1	Two way drift extender card – similar to AHK (01.41.07.1).
DBU	IDL	C.00.22.1	Inverting Data Line Driver – used with 1623. A +S input level produces a -C output level (01.41.03.1).
DED	DL	C.00.23.1	Non-Inverting Data Line Driver – used with 1622. A +S input level produces a +C output level (01.80.20.1).
DEE	DT	C.00.23.1	Inverting data line terminator – a +C input level produces a -S output level (01.80.20.1).
DFA	L	C.00.24.1	Latch Inverter – used in 1622, similar to TCZ (04.35.16.1).
DFD	AL	C.00.24.1	And latch – drift type (01.06.10.1)
FL WF	SWD	C.00.25.1	Decode switch – four -S inputs are required to cause conduction (01.30.26.1).
FM WG	D	C.00.25.1	Current driver – used in conjunction with the FL card to provide matrix switch input currents (01.30.10.1).
FP	R	C.00.30.1	Load resistors – MARS Rd/Wr driver (01.56.45.1) and MARS Bit driver (01.55.62.1).
FQ ALY	DP	C.00.30.1	Constant Current Source – Matrix switches and buffer (01.30.10.1).
MH	I	C.00.35.1	Power Inverter (01.06.20.1).
MX	A, O, or I	C.00.35.1	Two way alloy AND/OR or Inverter card (01.06.10.1).

Figure 4-2a. SMS Card Index

Card Code	Symbol	Systems Diagram	Description
NN	AM	C.00.40.1	Sense Amplifier – used in 1622 buffer and MARS (01.57.01.1).
NP	DP	C.00.40.1	Relay Driver – for typewriter relays, punch magnets, and console lights (01.06.12.1).
PF	D	C.00.41.1	Driver – relay, light, or inverter.
PG	SW	C.00.42.1	1621 Brake and Clutch switch card (02.83.60.1).
PH CBW	AM	C.00.42.1	1621 Photocell Amplifier (02.83.50.1)
PJ	D	C.00.43.1	1621 Brake and Clutch Driver (02.83.60.1).
PK	AM	C.00.43.1	1621 Sync Disk Amplifier (02.83.50.1).
RE	THER	C.00.45.1	Thermoswitch Circuit (01.90.42.1).
VE TAG	A, O, or I	C.00.45.1	Two way drift type inverter (01.06.12.1).
TAB	DL	C.00.46.1	Non-Inverting Data line driver – a +S input level produces a +C output level (01.41.16.1).
TCK	R	C.00.46.1	Load Resistors for line terminators (01.41.07.1).
TDA	TL	C.00.47.1	Inverting Line terminator circuit – a +C input level produces a -S output level (01.41.07.1).
TBD	SW	C.00.48.1	1622 Decode Switch – used in conjunction with the ACY card (04.30.06.1).
TCZ	L	C.00.48.1	Latch Inverter – similar to DFA except that R pin is a common reset to both circuits (01.10.07.1).
VF TAH	A or O	C.00.50.1	Three way drift type AND/OR card (01.06.10.1).
VG TAF	OSC	C.00.50.1	Oscillator circuit card provides one megacycle clock timing for direct drive to voltage mode circuits and AC drive to SDTRL triggers (01.10.05.1).
VJ TAJ	TB	C.00.55.1	Binary Trigger (01.10.05.1)
VL TAL	DSP	C.00.55.1	Sample Pulse generator (01.10.12.1)
VM TAK TFC	I	C.00.60.1	Power inverter – drift type (01.06.10.1).
WM	AMP	C.00.60.1	Single Shot sense amplifier – receives signal from a memory pre-sense amplifier. Provides 1 usec pulse to MBR latches (01.30.50.1)
WJ	D	C.00.65.1	Current driver for core storage (01.30.80.1).
WY	AMP	C.00.65.1	Pre-sense Amplifier – used to amplify a nominal 40 mv signal (01.30.50.1).
YAR	DIODE	C.00.69.1	Diode card for back circuit elimination (04.30.05.1).

Figure 4-2b. SMS Card Index

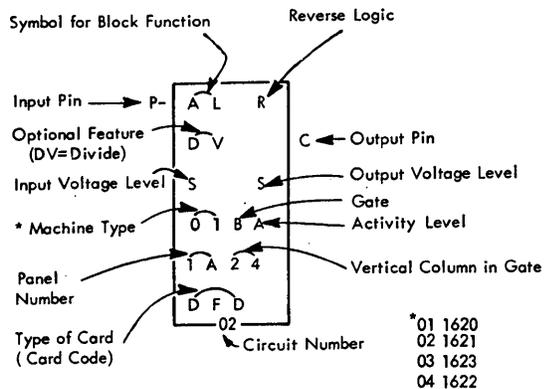


Figure 4-3. 1620 System Diagram "Block"

TAJ Card

There are two AC set inputs per state (OFF or ON) with each input single gated. There is also a provision for DC set and reset (Figure 4-7).

OPERATION (Figure 4-8)

Assume T3 is OFF and T1 is ON. Then the output of T3 is about -10v (driving no external loads) and the output of T1 is ground (+S). The anode of D23 is at -12v because of R22 and the anode of D4 is at ground because of R2.

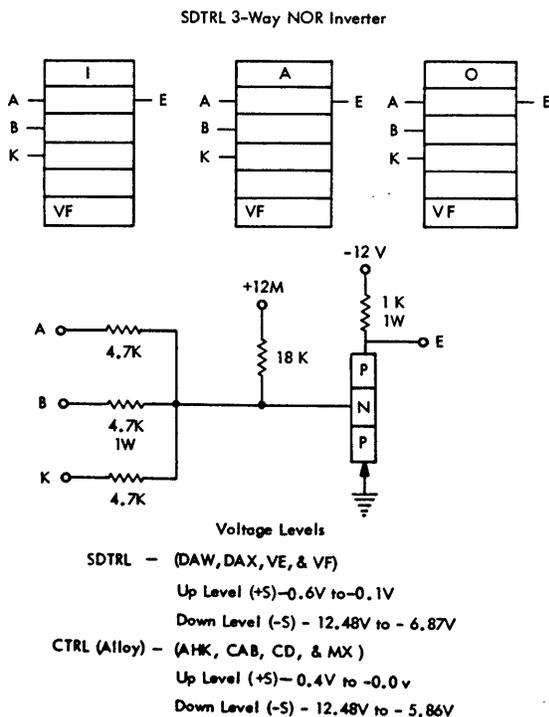


Figure 4-4. SDTRL 3-Way NOR Inverter

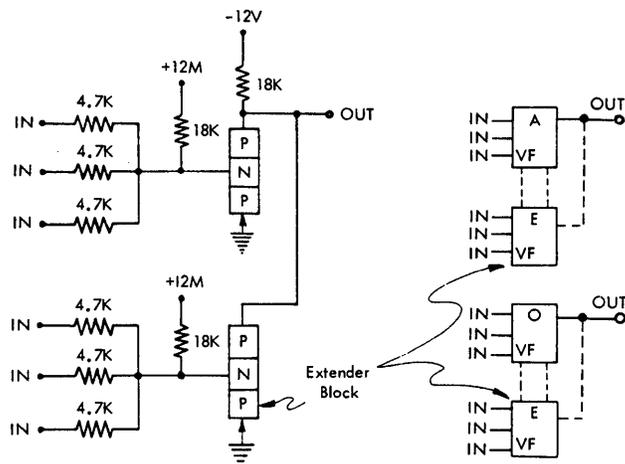


Figure 4-5. Extenders

T3 is held OFF by R32 and T1 is held ON by R30 and R28. A 6-v positive shift at the input is passed through C24 and C3 causing the anode of both D23 and D4 to shift positive. Since D23 is back biased by 12v, the 6-v shift does not get through to the base of T-3. However, since D4 was zero biased the 6-v shift causes the base of T1 to go positive, turning T1 OFF. When T1 goes OFF, R14 and R10 cause T3 to go ON, which in turn holds T1 OFF by nature of the voltage divider formed by R7 and R28. The voltage conditions at the anode of D23 and D4 are now reversed and the trigger is ready for the next positive shift.

By applying a negative level to either R33 or R26 (DC set or reset), the corresponding side of the trigger may be turned ON.

The additional AC set and gates are provided to increase the logical flexibility of the trigger.

A +S level applied to pin P will turn the trigger ON by "collector pull on."

A +S level applied to pin B will turn the trigger OFF by "collector pull off."

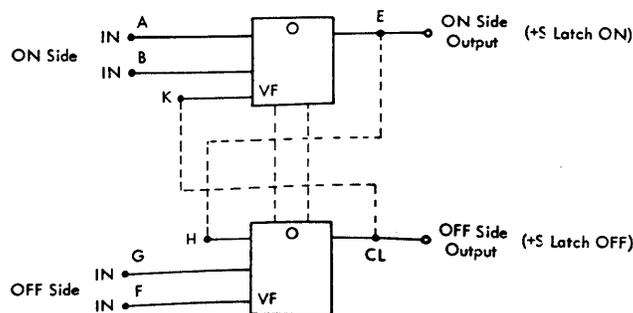


Figure 4-6. NOR Latch

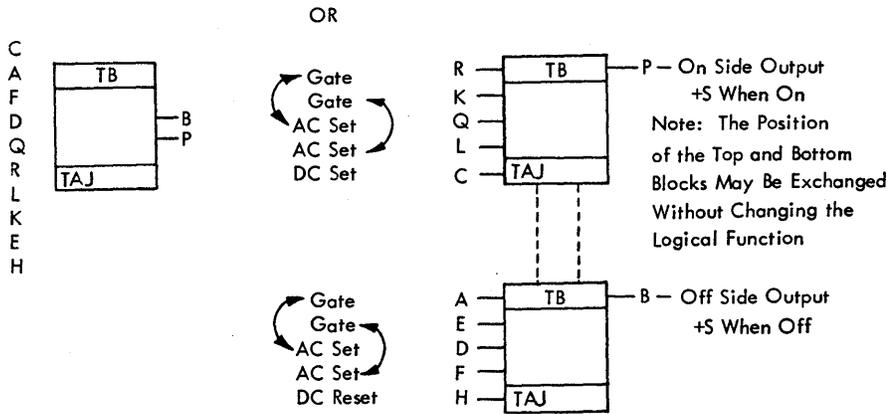
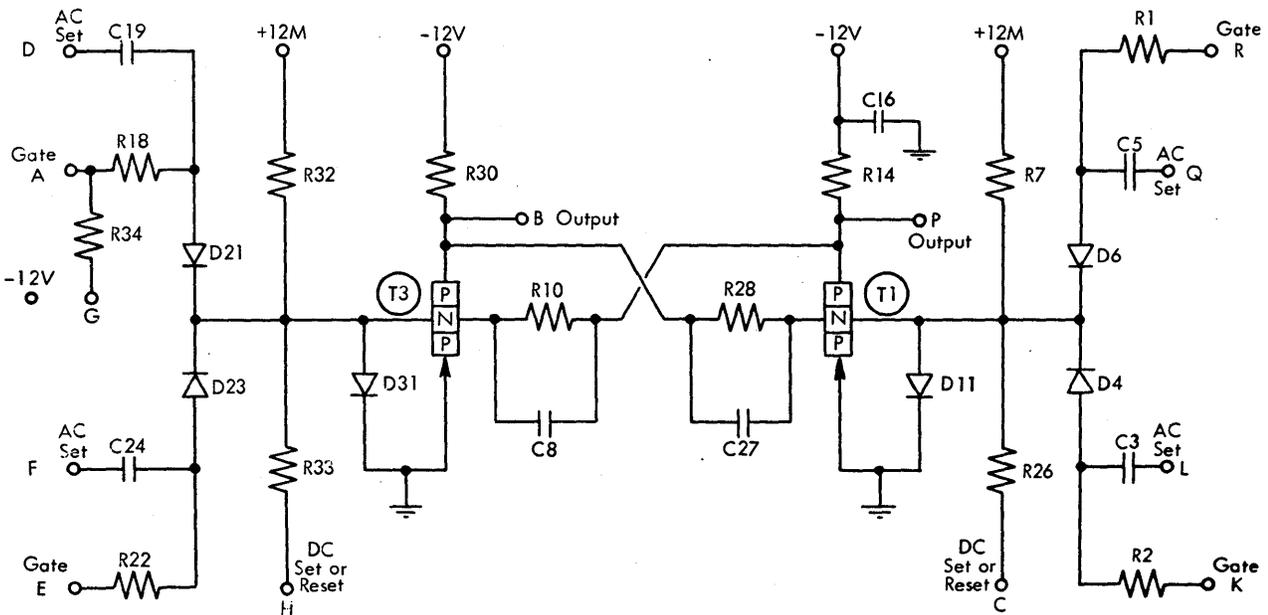


Figure 4-7. SDTRL Trigger, Binary (Block)



VOLTAGE LEVELS

AC Set:	Min. 4.5V Shift (Positive) Max. 5V Shift (Positive)	DC Set or Reset	Up Level: Max. 0V Min. -0.65V
DC Gate	Up Level: Max. 0V. Min. -0.6V		Down Level: Max. 6.87V Min. -12.48V
	Down Level: Max. -6.87V Min. -12.48V	Output Levels DC	Up Level: Max. -0.1V Min. -0.6V
			Down Level: Max. Depends on Load Min. -9.96V

Figure 4-8. SDTRL Trigger, Binary (Schematic)

AFR Card

This block provides a current source for the 12v indicator lamps on the console (Figure 4-9).

A - S level applied to pin A causes T2 to conduct, supplying a low resistance path to ground for the current through the lamp. When T2 is cut OFF the 1.8K resistor to ground does not allow enough current to pass, to light the lamp.

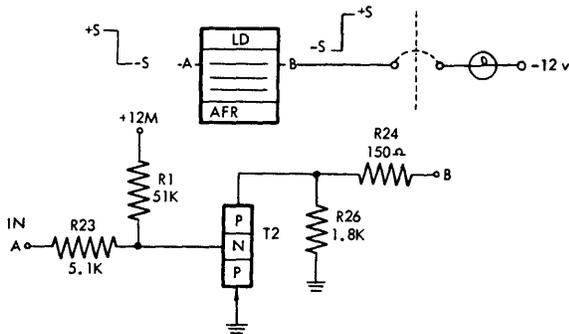


Figure 4-9. CTRL Light Driver

CAU and DFD Cards

Each latch circuit contains three transistors. One transistor is used as a two-way or three-way AND circuit which feeds an input to the other two transistors. The second two transistors are connected as an OR latch. The AND latch is turned on with a + S input to all legs of the AND portion of the circuit (+ S on E and F pins for circuit 1 or + S on P, Q, and R pins for circuit 2) as shown in Figure 4-10. The latch may also be turned on with a - S input to pin H for circuit 1 or to pin K for circuit 2. Output pin G for circuit 1 or pin C for circuit 2 is + S when the latch is on and is - S when the latch

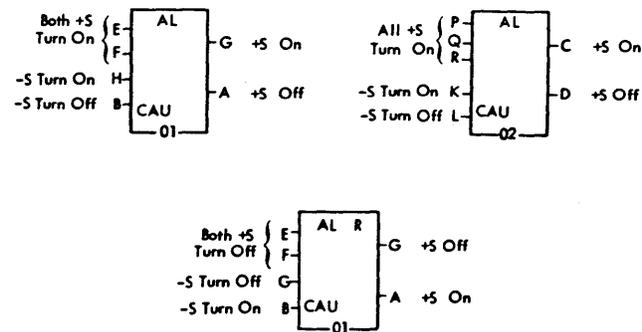


Figure 4-10. AND Latch

is off. Output pin A for circuit 1 or pin D for circuit 2 is - S when the latch is on and is + S when the latch is off. The input and output functions of the logic block are reversed if an "R" is in the upper right corner of the logic block. This reversed logic is used when an AND condition is needed to turn off a latch.

NP Card

This block provides the control and current source for operation of 48v relays, magnets, and signal lights (Figure 4-11).

A + S potential applied to pins A and B (see note) will cause T4 to go into conduction when a 48v potential is applied to pin C. Once conduction is started, the current drawn from the base of T4 to the emitter will cause T4 to remain in conduction regardless of the potential at pins A and B. Conduction is cut off only by removing the 48v supply to pin C. The diagram shows the usual configuration for the output of this circuit. When the CB opens conduction ceases.

NOTE: As generally used in the 1620, one input pin is not connected to any potential (Floating). In this case a + S level to the other pin will cause conduction as described above.

WY Card

This card provides a bipolar amplifier to detect and amplify a signal on a sense winding of a memory core plane (Figure 4-12).

Due to the physical position of memory cores, a core flipping to the "zero" state or "one" state can induce a positive going pulse or a negative going pulse in the sense winding. The bipolar pre-sense amplifier detects either type of pulse. The memory sense amp

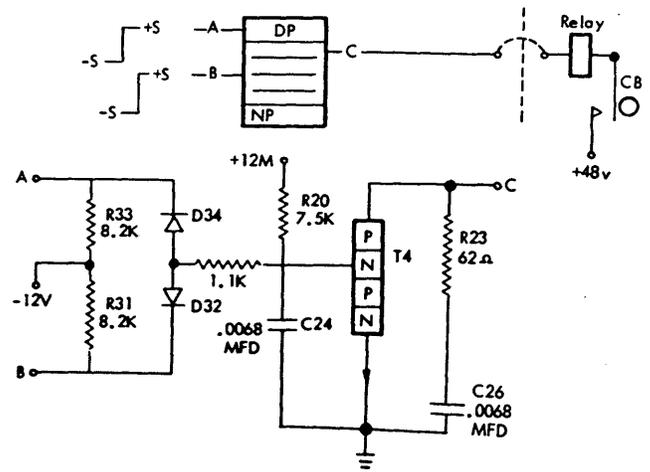


Figure 4-11. CTRL-Driver, Relay, Thyatron

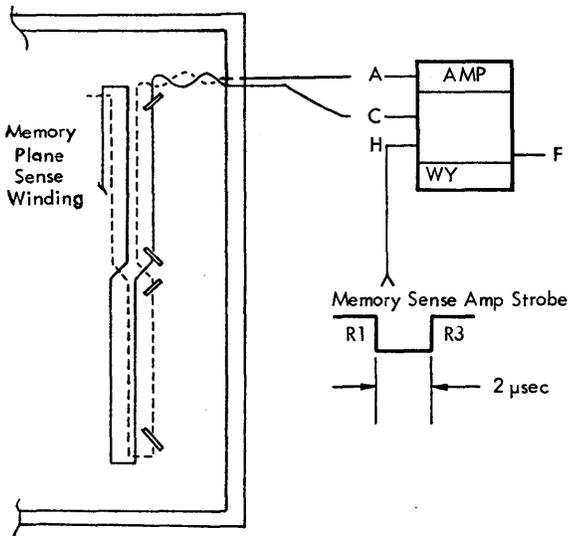


Figure 4-12. Pre-Sense Amplifier, Alloy

strobe discriminates between a read and a write pulse and is timed to select the read pulse only.

TYPICAL VOLTAGE LEVELS AND WAVEFORMS
(Figures 4-13, 4-14, and 4-15)

Waveforms and potential for pulse generated when a core is flipped, i.e., the selected core is in the "one" state and flips to the "zero" state (Figure 4-13).

Waveform and potential for pulse generated when a core is selected but does not flip, i.e., the selected core is in the "zero" state and the currents applied are in a direction such as to try to flip the core to the "zero" state (Figure 4-14).

NOTE: These waveforms and potentials apply at "write" time also. However, the memory sense amp strobe is not present at write time and there is no output from the pre-sense amplifier (Figure 4-15).

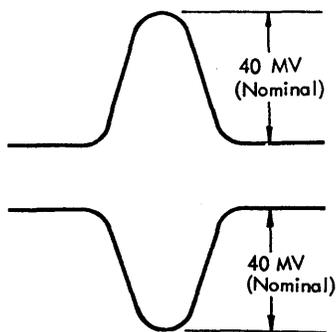


Figure 4-13. Memory Core Output

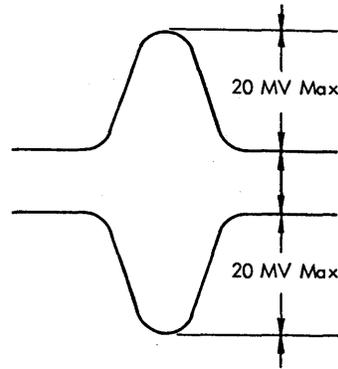


Figure 4-14. Memory Core Output

WM Card

These cards take the output of the pre-sense amplifiers, reinforce it, and provide a 1 μsec pulse output to turn ON latches in the memory buffer registers (Figure 4-16).

TYPICAL VOLTAGE LEVELS AND WAVEFORMS

(Figures 4-17 and 4-18)

For input waveform and potentials, see pre-sense amplifier output.

Output waveform and potential of sense amplifier single shot as illustrated in Figure 4-17.

NOTE: When the output of this single shot is connected so as to cause collector pull on of a latch, the waveform as seen on a scope will show the output of the trigger.

VL Card

These sample pulse generator cards provide a positive going pulse of a definite duration suitable for driving several SDTRL binary triggers (Figure 4-19).

The sample pulse generator requires a +S gate applied to pin C and a positive shift (AC set) applied to pin BA or BE to obtain a +S pulse output at pin D. Pin C can be tied to ground, thereby supplying a continuous gate. The +S gate must be applied a minimum of 0.7 μsec prior to the AC set or shift pulse. The output waveform is shown in Figure 4-20.

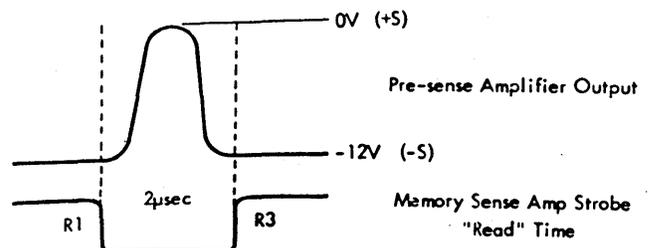


Figure 4-15. "Read" Time

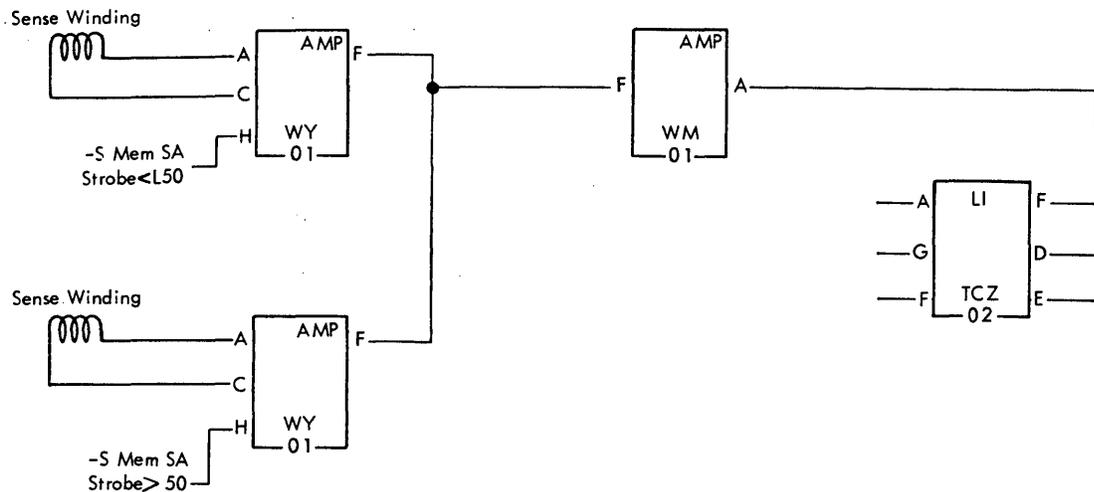


Figure 4-16. Sense Amplifier, Alloy

RE Card

This card is used in the SMS gates to check temperature. The thermal switch opens at 122°F, causing the power supply to sequence OFF (Figure 4-21).

PG, PH, PJ, and PK Cards

These cards are used in the IBM 1621 Paper Tape Reader. A description of their functions will be found in Section 13.

NN Card

This card provides a bipolar amplifier to detect and amplify a signal on a sense winding of the MARS core plane (Figure 4-22). The NN card wired in this manner functions as an extender to the TCZ card and turns on the latch by collector pullover.

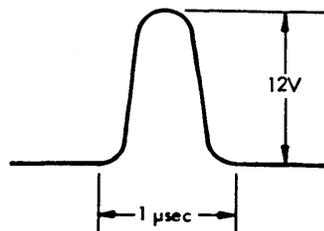


Figure 4-17. Sense Amplifier Output

Because of the physical position of the cores with respect to the "X" and "Y" wires, a unipolar pulse will be generated in the sense winding when a core is flipped. A "MARS Sense Amp Sample" (Strobe) is used to discriminate between read and write pulses, being timed so that only a read pulse is detected. The output waveforms and voltage levels are shown in Figures 4-23, 4-24, and 4-25.

NOTE: When the output of the sense amplifier is connected to the collector of a latch which is turned ON by collector pullover, the waveform as seen by a scope at pin A or AF will therefore be the output of the latch (Figure 4-26).

FL, FM, and ALY Cards

These cards are associated with the addressing of memory and will be discussed in terms of that application (Figure 4-27).

The ALY card provides a constant current source for

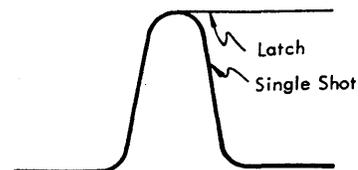


Figure 4-18. Sense Amplifier Output Connected to Trigger (Collector Pullover)

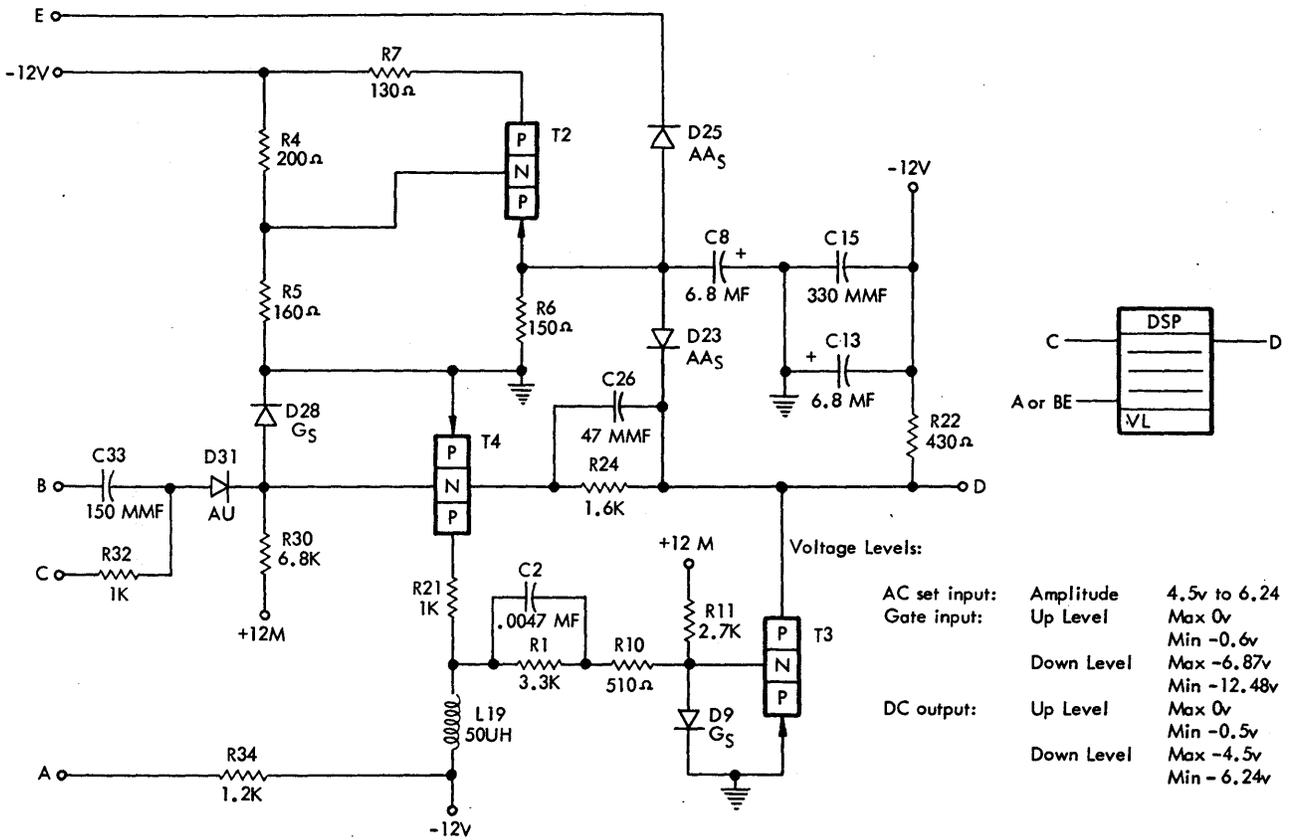


Figure 4-19. SDTRL, Sample Pulse Generator

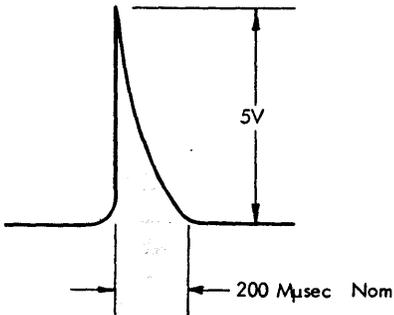


Figure 4-20. Sample Pulse Generator Output Waveform

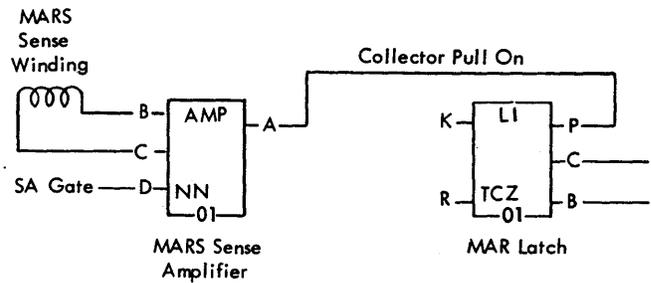
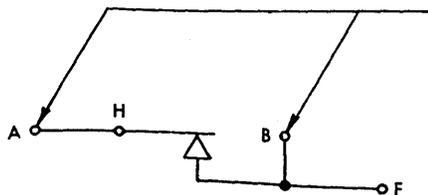


Figure 4-22. CTDL-Sense Amplifier Load Resistor



Pins A and B are not Inputs or Outputs
They Insure Positive Identification
When This Card is Accidentally
Substituted for 371696, 371697 and
371698

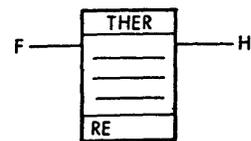


Figure 4-21. Thermal Switch

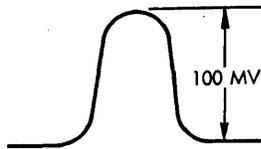


Figure 4-23. Input Waveform

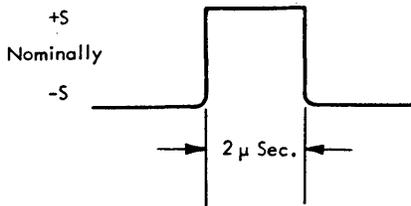


Figure 4-24. Output Waveform

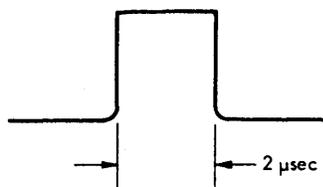


Figure 4-25. Strobe, MARS Sense Amp Sample

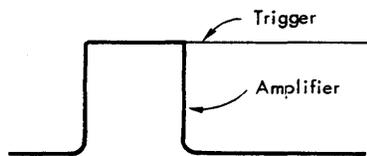


Figure 4-26. Amplifier Output Connected to Trigger (Collector Pullover)

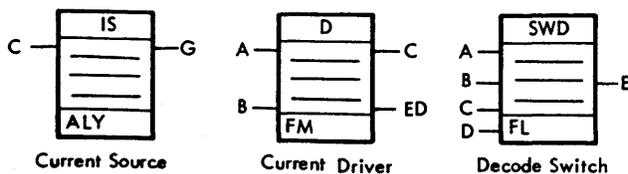


Figure 4-27. SMS Cards Associated with Matrix Switches

use with the switch cores which are packaged one circuit per card.

The FM card provides: (1) A path for the current source during the time when a switch core is not selected and the control gate is not present. This path maintains current flow through the bias winding.

(2) A timed control (gate) causing the current path to be directed through the axes of switch cores to a selected decode switch (FL card). The current through the bias winding is maintained, during the time a switch core is selected, by a path, through the selected core, to the decode switch. Packaged 1 circuit per card.

The FL card (decode switch) provides a means for selecting a particular switch core coordinate according to the addressing scheme. The decode switch provides a path to ground, through the switch core, for the current source when the gate to the FM card blocks the path through the FM card circuits. Packaged two circuits per card. Circuit 1 shown.

The bias winding is wound with four turns around every switch core, in series, of matrix switches A and D (refer to Figures 4-28 and 4-29). This winding carries a current in such a direction and of such a magnitude that it tends to retain all switch cores in a "zero" state. The bias winding current (1 amp) supplies current for the Y'Y" and X'X" axes of the switch cores. Note that the bias winding current is divided through four circuits; units and tens (Y'Y") of matrix switch A and hundreds and thousands (X'X") of matrix switch D. One ampere in the bias winding times 4 turns equals 4 ampere turns of magneto-motive force. The Y'Y" and X'X" axes are wired with 12 turns around their respective switch cores. These 12 turns are wound in such a manner that their MMF opposes the bias winding MMF. Since the bias current divides equally through the Y'Y" and X'X" wiring, there will be 250 MA flowing through each coordinate wire. The 250 MA's times 12 turns equals 3 ampere turns of magneto-motive force. It requires 2 ampere turns to flip a switch core. When one coordinate of a matrix switch is carrying current (3 ampere turns) in opposition to the bias current (4 ampere turns), the effect of the bias current is nearly cancelled. When both coordinates are carrying current (3 + 3 = 6 ampere turns), the effect of bias current is cancelled and the core will flip to its "one" state. If the current through one coordinate is removed, the core cannot flip back to its "zero" state since 3 ampere turns from the remaining coordinate are opposing the 4 ampere turns of the bias winding. Four minus three equals one turn which is not enough to flip the switch core to "zero". When the second coordinate current is removed, the 4 ampere turns of MMF in the bias winding will flip the core to its "zero" state.

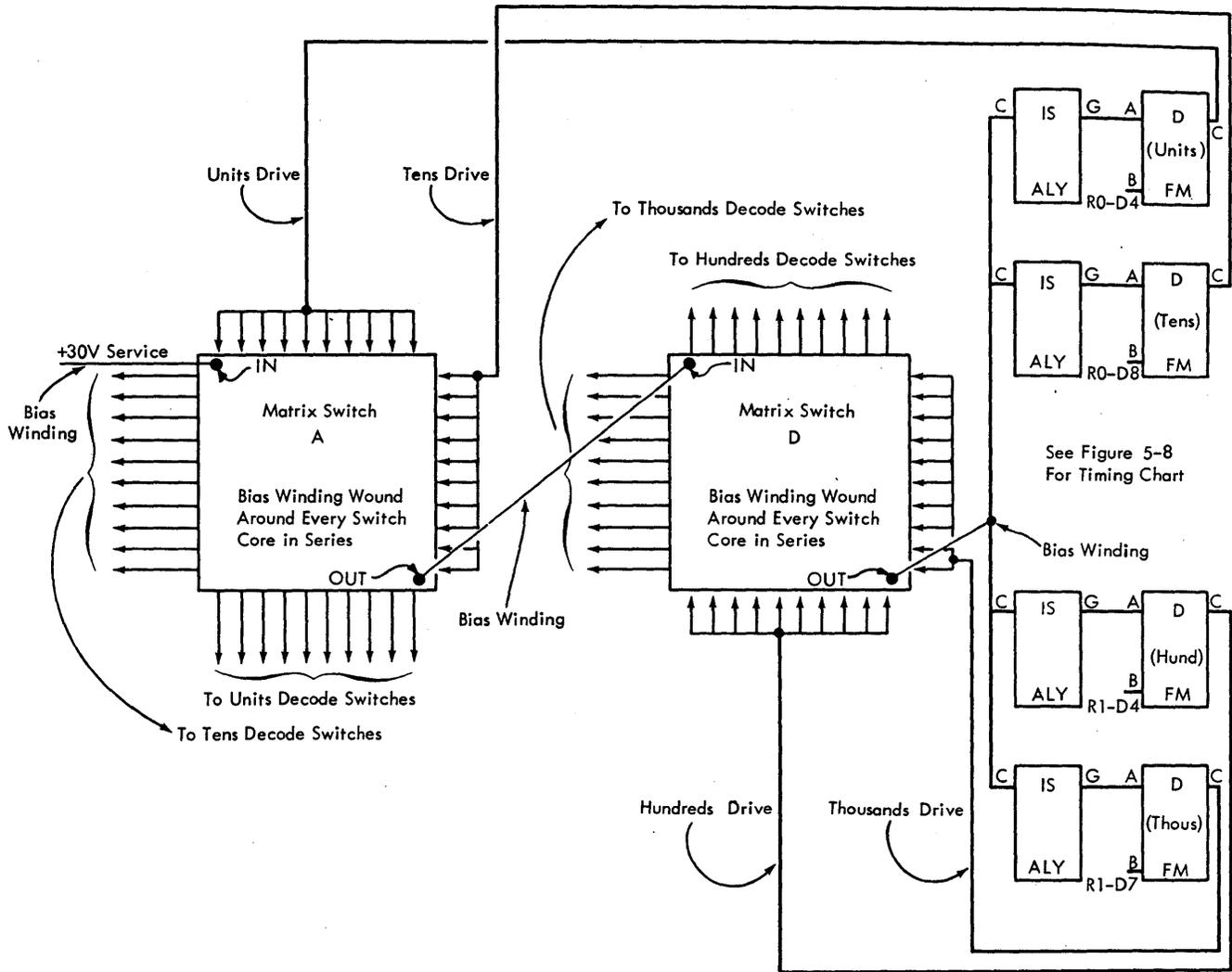


Figure 4-28. Matrix Switch Bias Winding

CIRCUIT OPERATION WITH SWITCH CORE NOT SELECTED

Starting at +30v, (1) Figure 4-29) the bias current passes through all switch cores in series to pin C of the current source (ALY) cards. From pin C, the current passes through a noise elimination inductor, two 90Ω 5W resistors in parallel, T1, noise elimination inductor, and resistor in parallel, to pin G of the ALY card. The +30v through the 4.7K resistor is for level setting. Pin G is wired to pin A of the FM card.

The current then flows through T2, which is conducting, to the -12v supply to complete the circuit.

SELECTING A SWITCH CORE

Prior to the selection of a switch core, the status of the transistors in the decode switch are as follows:

- (1) T4 conducting which places a minus potential on the base of T5.
- (2) T5 cut off by the action of T4. The

addressing circuits place -S (-12v) on pins A, B, C, and D. This reverse biases T4 and cuts it off, raising the potential at its collector to +30v. The current in the 1.25 μh inductor causes a sharp peak of potential to be placed on the base of T5 to bring it into conduction more rapidly.

With T5 in conduction, the ground at its emitter is available at pin E of the FL card, through the selected switch core to pin C of the FM card. However, current will not flow through this circuit until T2 is cut off thereby blocking the current path to -12v.

CIRCUIT OPERATION WITH SWITCH CORE SELECTED

As long as pin B of T3 is at +S, T3 will conduct, causing a forward bias to be applied to the base of T2 which in turn stays in conduction. When the -S gate (R0-D4 for units of matrix switch A) is placed on pin

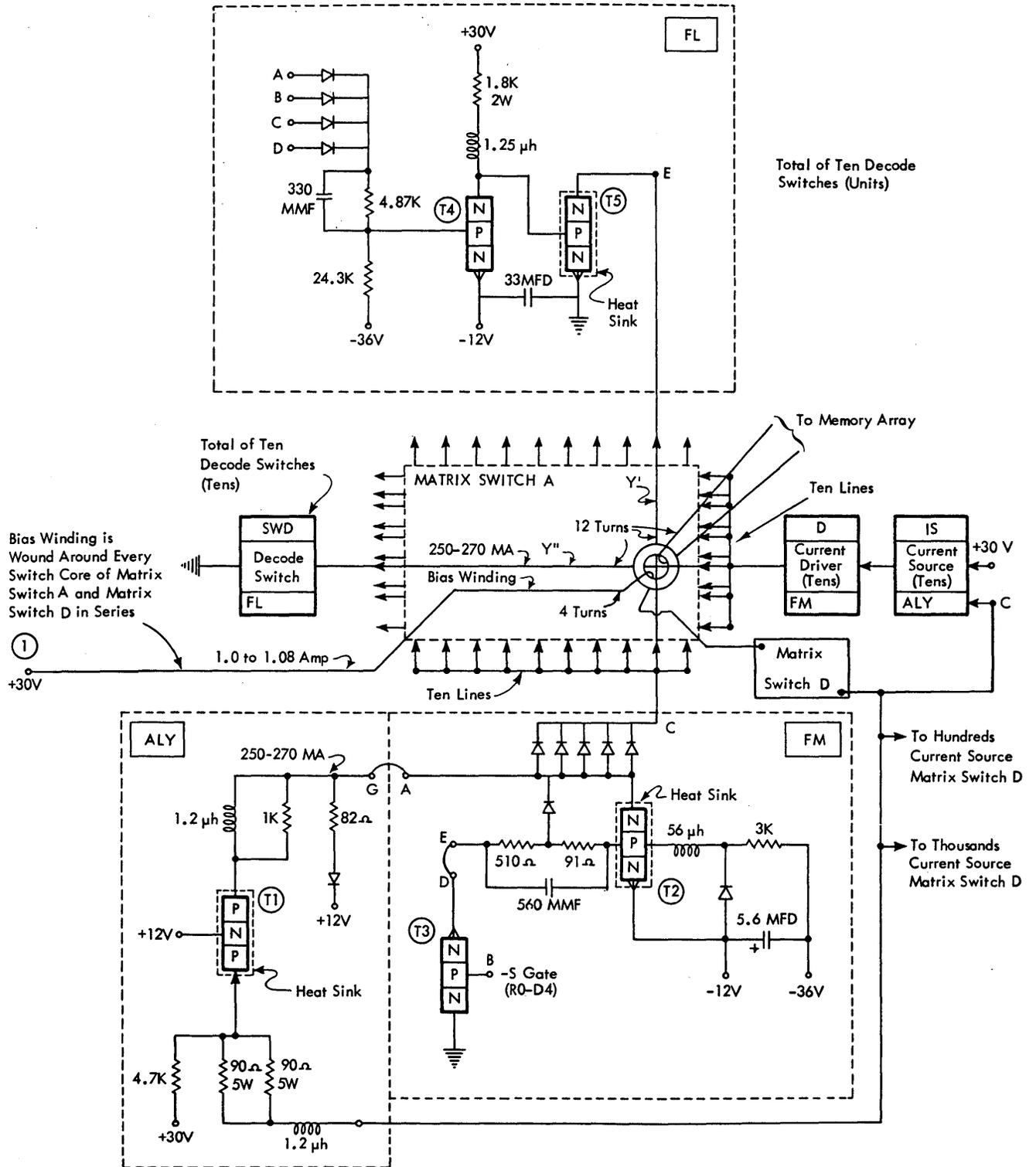


Figure 4-29. Matrix Switch Operation

B of T3, T3 is cut off. This removes the forward bias from T2 which in turn is cut off. With T2 cut off, the current seeks a new path which is found by passing through the diodes of the FM card, through the switch core, to ground through T5. Note: The direction of this coordinate current flow through the switch core is in opposition to the bias winding current flow.

When a switch core in matrix switch A flips to the "one" state, it induces a current pulse in the winding connected to the memory array. This induced current produces the Y coordinate for a "read" out of memory. Matrix switch D provides the X coordinate.

When the gate at pin B of T3 is removed (returns to +S), T3 will conduct which in turn forward biases the base of T2 causing T2 to conduct and provide a path for the current flow to -12v. Current will no longer flow from the current source (ALY) to the ground through T5. As soon as the Y' and Y" current flow is cut off, the effect of the bias current, which is still present, will flip the selected switch core back to the "zero" state. When a switch core in matrix switch A flips to its "zero" state, it induces a current pulse in the winding connected to the memory array. This current pulse is induced in the opposite direction to that induced when a switch core was flipped to the "one" state. This current produces the Y coordinate for a "write" into memory. Matrix switch D provides the X coordinate.

TCZ and DFA Cards

Each card contains two latches which are logically and functionally independent. Each latch is shown on the system diagrams as one block. Therefore, one TCZ or DFA card is represented as two separate blocks on the system diagrams (see Figure 4-30). On both TCZ and DFA cards, a -S input to pin K or pin L will turn on latch no. 1. A +S input to pin P will turn on latch no. 1 by collector pullover. Pin C is at a +S level when latch no. 1 is off and is at a -S level when the latch is on. Pin B is at a -S level when latch no. 1 is off and is at a +S level when the latch is on.

Latch no. 2 of both the TCZ and DFA cards is turned on by a -S input to the A or G pin. A +S input to pin F will turn on latch no. 2 by collector pullover. Output pin D is at -S when latch no. 2 is on and is at +S when the latch is off. Output pin E is at +S when the latch is on and is at -S when the latch is off. The TCZ and DFA latches are turned off as follows:

- TCZ -S applied to pin Q turns off circuit 1.
- S applied to pin H turns off circuit 2.
- S applied to pin R turns off circuits 1 and 2.
- DFA -S applied to pin Q turns off circuit 1.
- S applied to pin H turns off circuit 2.
- S applied to pin R turns off circuit 1.

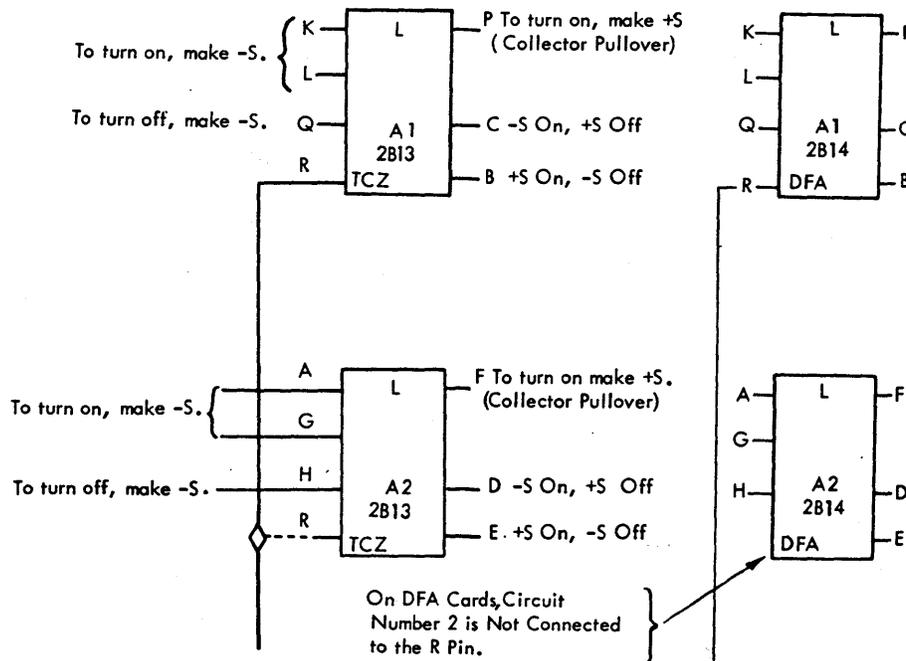


Figure 4-30. TCZ and DFA Latches

QU Cards

These differential amplifier cards are used in the SMS power supply regulators. All SMS power supplies in the 1620 are referenced to ground. Refer to *IBM Customer Engineering Manual of Instruction, 60 Cycle SMS Power Supply* (Form 225-6478). The circuits on this card are shown in Figure 4-31.

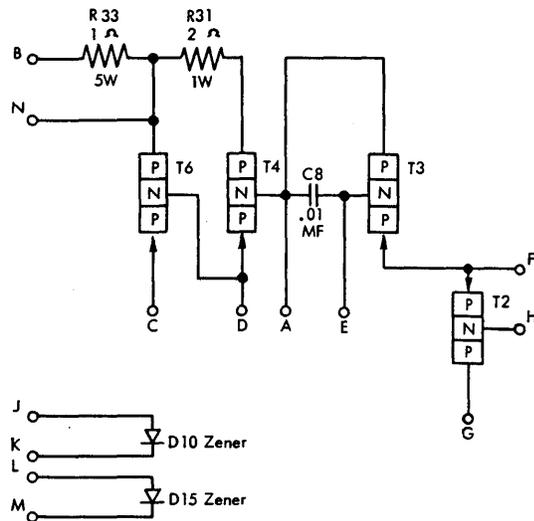


Figure 4-31. SMS Power Supply Differential Amplifier

WJ Cards

These alloy drivers are used as memory inhibit (Z) drivers, MARS bit drivers, MARS read drivers, and MARS write drivers (see Figure 4-32). When the inputs are at +S, a +R level is at the output. When the inputs are at -S, the output goes to a -R level.

Power Supplies

The 1620 requires an input voltage of 208v AC or 230v AC. The 208v supply can vary between the limits of 187v and 228v and the 230v supply can vary between 207v and 253v.

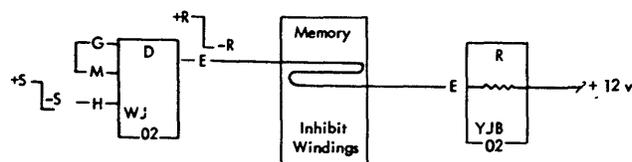


Figure 4-32. Alloy-Driver

SMS Power Supplies

The internal functions of the individual SMS power supplies and the theory of ferroresonant regulator transformers are discussed in the *IBM Customer Engineering Manual of Instruction, 60 cycle SMS Power Supply* (Form 225-6478). For normal operation, the 1620 SMS power supply DC output voltages must be maintained within the following levels:

+12	from +11.52 to +12.48	
+48	from +43.20 to +52.80	(Measured at power supply)
-36	from -34.56 to -37.44	
-12	from -11.52 to -12.48	
+3	from +2.94 to +3.06	

All machine functions must operate correctly when the +12M DC supply is set between +10.2 and +13.8 (Marginal test).

Power ON Sequence (Figure 4-33)

The following describes the sequence, objectives, and functions of operations necessary to place the 1620 in a ready status.

POWER CIRCUIT BREAKER (PCB) ON

Objectives:

1. Apply voltage to convenience outlets.
2. Put the "Normal" relay, R105, under control of:
 - a.) R104 which in turn is controlled by the gate and memory overtemp thermal trips.
 - b.) SMS power supply overload trip CB contacts.
3. Supply 24v DC for power ON sequence control circuits and DC monitor circuits.

Turning the PCB ON develops 24 and 48v AC. This voltage turns the thermal light on and picks K1. The points of K1 make, supplying voltage to the convenience outlets and to the rectifier which develops 24v DC for sequencing. The sequencing voltage supplies potential to the console power switch (01.05.03.1). Depressing RESET picks R104 which remains up until the PCB is turned OFF or a thermal switch opens. When R104 is picked the thermal light goes off and R105 is picked. Relay 104 is under control of the N/C contacts of the over-temp thermostats. Normal power on and power off operation does not affect the operation of R104. Therefore, under normal conditions, RESET need be operated only once after the PCB is turned ON.

CONSOLE POWER SWITCH ON

Objectives:

1. Bring DC voltages up to operating levels. These voltages are: -36, -12, +12, +48, ±3, +30. The +30v, used with memory addressing circuits, is brought up last to allow all biasing and level setting potentials to be up before the +30 is

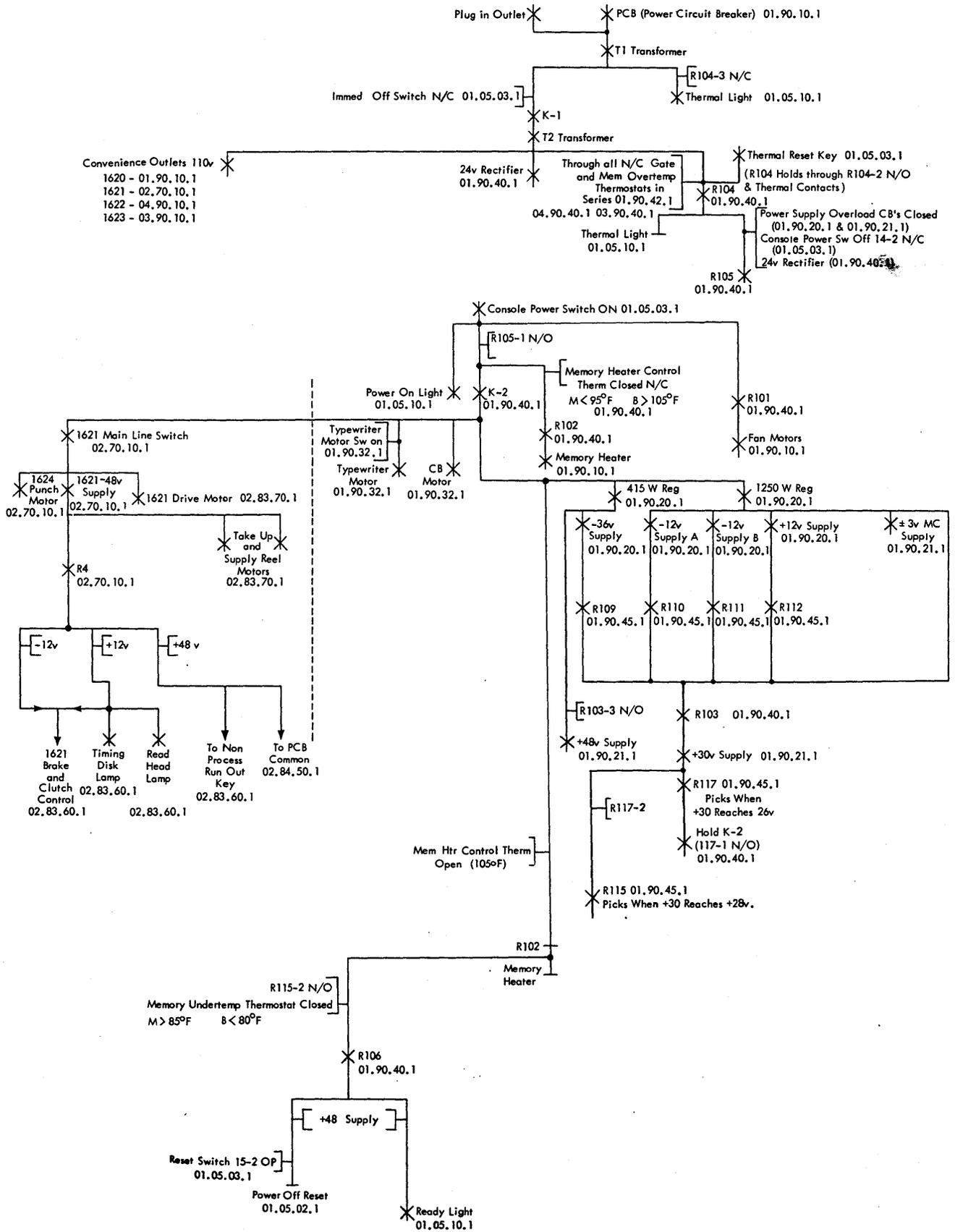


Figure 4-33. Power ON Sequence

applied. This prevents the +30 from disturbing memory when it is brought up. The +30 supply is dropped first in a power off sequence for the same purpose.

2. Develop a power ON reset to reset all functional units to their initial condition.
3. Turn on blower fans and CB drive motor.
4. Provide operating potentials to the 1621, 1622, 1623, and 1624. The 1621 contains its own -48v supply. The 1622 contains its own -20v, -12v, +12v, and ±3v (marginal check) supplies. The 1623 contains its own -12v, +12v, +30v, and ±3v (marginal check) supplies.
5. Place the memory heater and heater control relay (R102) in operation.
6. Provide a computer ready indication, the "Ready" light.

OPERATION OF R105, R102, AND THE MEMORY HEATER

Relay 105 is picked through R104-1, all SMS power supply overload trip contacts N/C, and the console power switch N/C. A circuit through R105-2 N/O parallels the power switch N/C points to hold R105 when the power switch is turned on. Turning on the console power switch picks R101, R102, and K2. Closing of R102-3 causes the memory heater to operate. The heater will continue to function until the temperature rises to 105°. At that time, the heater control thermostat will open to drop R102. The dropping of R102 turns off the heater and permits the pick of Ready R106 through the R115-2 N/O points and the memory undertemp thermostat. This puts the machine in a ready status and assures proper functioning of the cores.

The memory heater will now maintain a core temperature of 95° to 105°F as controlled by the memory heater control thermostat and Relay 102.

POWER ON RESET

When the -12v supply "B" is established, the machine reset lines become operative and reset all machine units. This reset will remain in operation until the Ready R106 is picked indicating the machine is ready for functioning.

Power Off Sequence

The following gives the status of the power supplies and control circuitry prior to, and after, a power off sequence. Turning the console power switch OFF is the normal way to cause a power off sequence. The other power off sequences are for machine failure conditions and immediate off switch operation

CONSOLE POWER SWITCH TURNED OFF (Figure 4-34)

The 1620 is in a normal ready status prior to turning switch OFF.

Relay picked: 101, 102 (depending on memory temperature), 103, 104, 105, 106, 109, 110, 111, 112, 115, 117, and K1, K2.

All voltages are at operating level.

CB motor and blower motors operating.

Potential supplied to the 1621 and 1624.

Power on light ON.

Status after power off sequence:

Relays picked: 104, 105, and K1.

24v sequencing remains up.

Convenience outlet voltage remains up.

THERMAL OVERTEMP TRIP (Figure 4-35)

The 1620 is in a normal ready status prior to thermal overtemp trip.

Status after power off sequence:

Relays picked: 101 and K1.

24v sequencing remains up.

Fan motor operating.

Thermal light ON.

Power on light ON.

Convenience outlet voltage remains up.

Restoring Procedure: Procedure for restoring 1620 to the ready condition after thermal trip is as follows:

1. Clear condition causing thermal trip.
2. Depress reset key (Picks R104).
3. Turn console power switch off then on to get normal power ON *sequence*. Turning the console power switch OFF completes a circuit through R104-1 N/O and the DC monitor network to pick R105.

SMS DC POWER SUPPLY OVERLOAD TRIP (Figure 4-36)

The 1620 is in a normal ready status prior to power supply overload trip.

Status after power off sequence:

Relays picked: 104, 101, and K1.

24v sequencing remains up.

Convenience outlet voltage remains up.

Fan motors operating.

NOTE: If the overload should trip on one of the 12v supplies, it can cause data in memory to be disturbed due to the +30 supply dropping after the 12v supply.

Restoring Procedure: Procedure for restoring 1620 to the ready condition after an overload trip is as follows:

1. Remove cause of overload.
2. Restore power supply circuit breaker.
3. Turn console power on switch OFF then ON to get normal power on *sequence*.

LOSS OF 24V DC SEQUENCING CONTROL

This allows random drop out of relays. Data in memory could be disturbed. The thermal light is turned ON and convenience outlets remain up.

"IMMEDIATE OFF" SWITCH ACTIVATED

This allows random drop out of relays which could disturb data in memory. The power on light remains ON and the thermal light is turned ON.

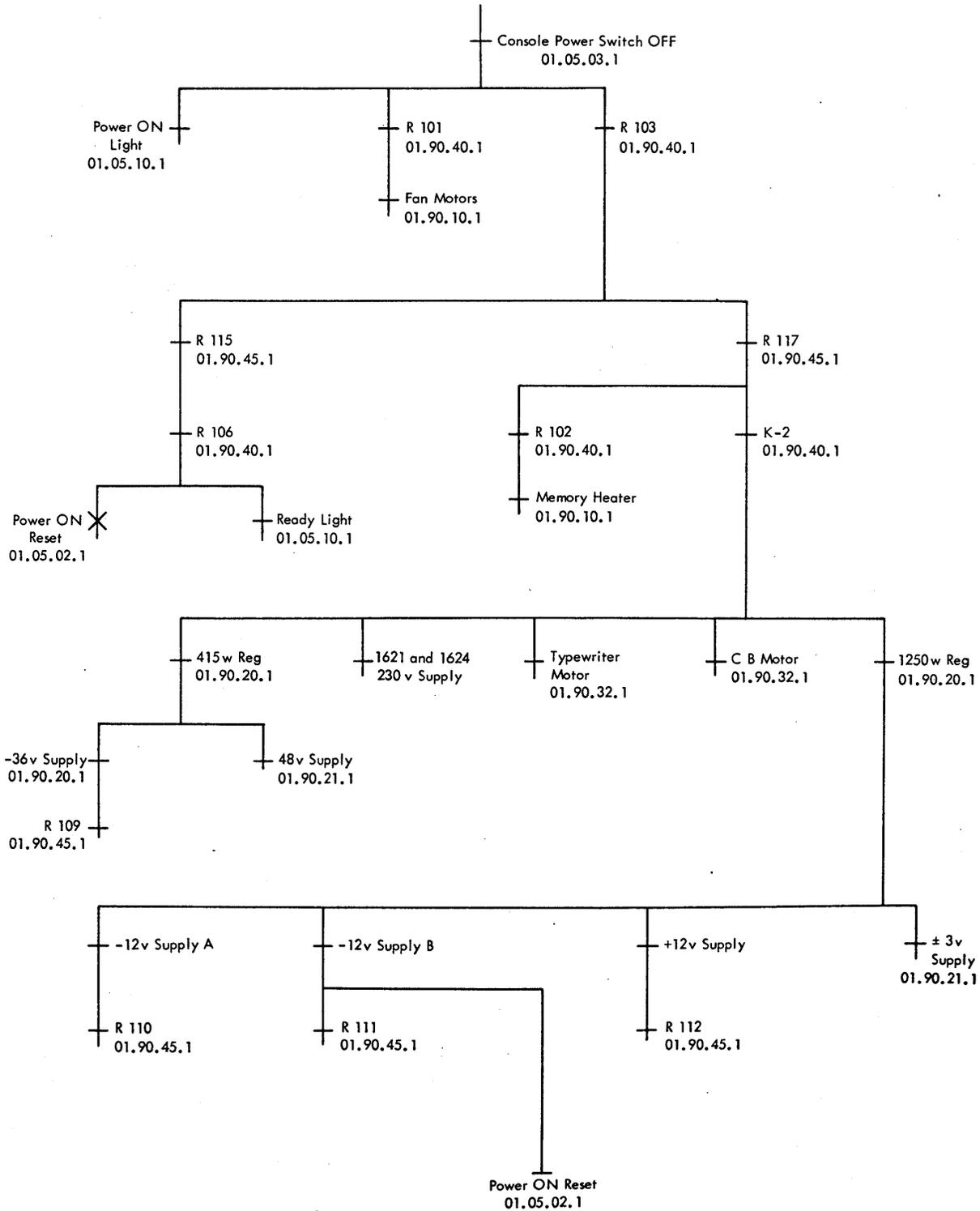
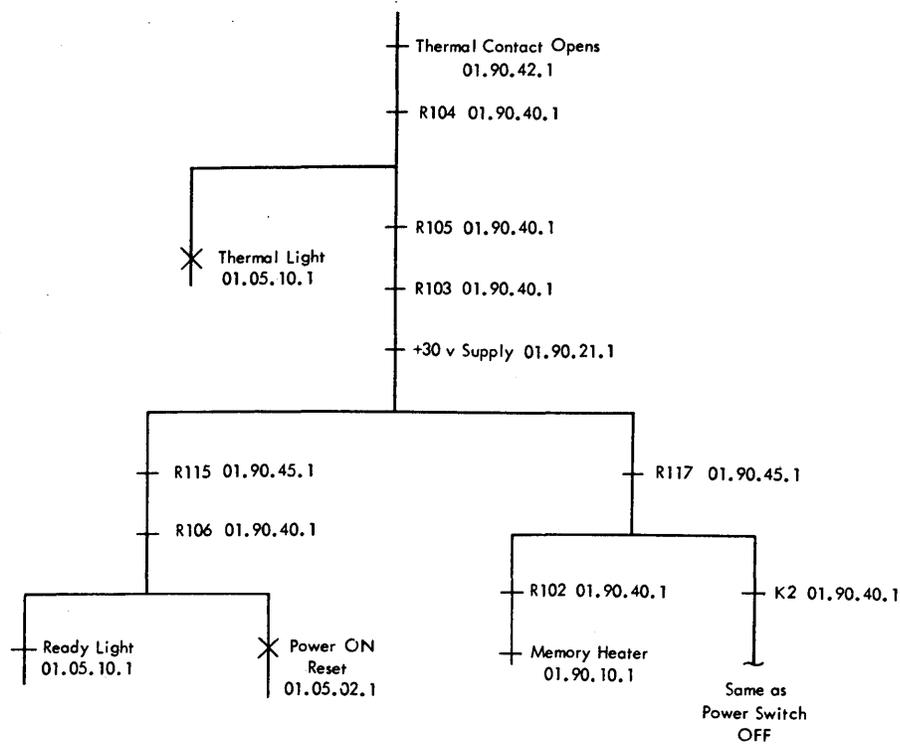


Figure 4-34. Console Power Switch Off



NOTE: R101 Stays up
Which Keeps Fan
Motors Energized

Figure 4-35. Thermal Overtemperature Trip

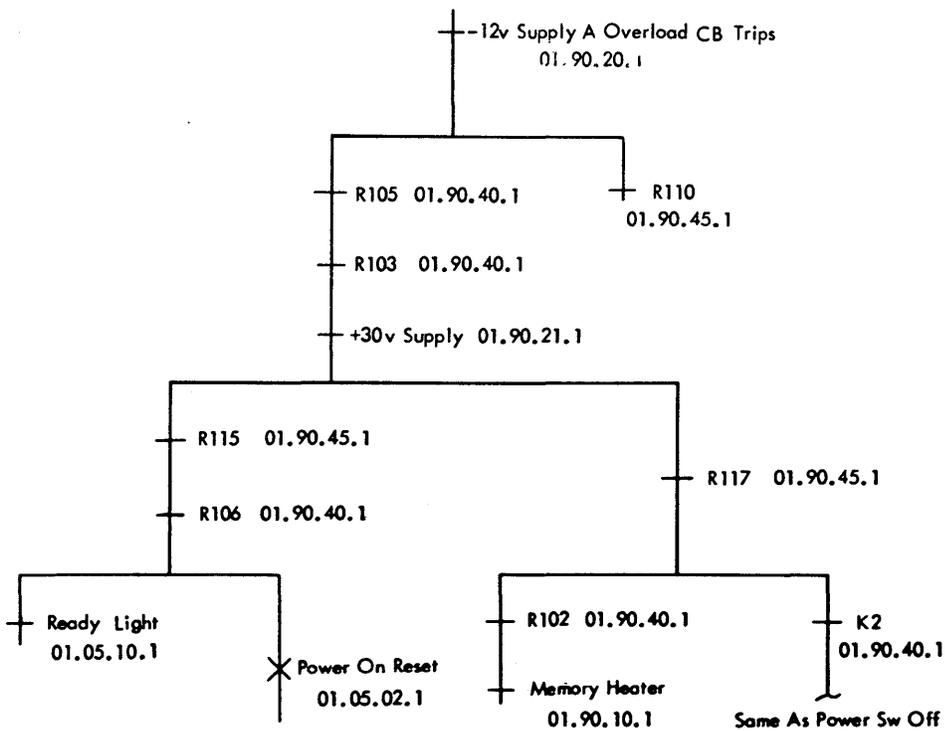


Figure 4-36. SMS DC Power Supply Overload Trip

Section 5 Functional Units

Clock and Timing Chart

A one megacycle free running crystal oscillator is the basic pulse generator for the machine.

The output of the crystal oscillator is connected to both the ON and the OFF side inputs of a binary trigger for the development of clock drive pulses. When the trigger is turned on, a clock drive "A" pulse is developed. One microsecond later when the trigger is turned OFF, a clock drive "B" pulse is developed. Clock drive A and B pulses are developed alternately in this manner as long as the run trigger is ON and the hold trigger is OFF. (The hold trigger is used in input-output operations to synchronize the input-output device with the computer.) See Figure 5-1.

The clock ring is composed of ten triggers which are turned ON at one microsecond intervals and remain ON for ten microseconds each. The ten triggers are turned OFF at one microsecond intervals and remain OFF for ten microseconds each. The odd numbered clock triggers (C-1, C-3, C-5, C-7, and C-9) are turned ON and

OFF by clock drive B pulses "anded" with the outputs of even numbered clock triggers. The even numbered clock triggers (C-2, C-4, C-6, C-8, and C-10) are turned ON and OFF by clock drive A pulses ANDED with the outputs of odd numbered clock triggers. The time required to complete a clock ring cycle is twenty microseconds.

Beginning with the clock drive B pulse which turns ON clock trigger C-1, the start of consecutive microsecond intervals are designated R_0 through R_6 , W_0 through W_6 , and T_0 through T_5 . T_5 is followed by R_0 .

Figure 5-2 shows a function chart and Figure 5-5 a sequence chart for the clock triggers.

The outputs of two of the ten clock triggers are ANDED to develop required pulses and gates from 1 μ sec to 19 μ sec in duration. Three examples are shown in Figure 5-3.

A machine cycle has a duration of 20 μ sec and consists of one clock ring cycle. The functional cycle in the computer is defined as a memory cycle and consists of a machine cycle which begins at W_6 clock time under control of the A/B trigger. Each memory cycle is a complete operation with respect to memory: that is, memory is addressed, read out of, and written into within the cycle.

The ON-OFF status of the A/B trigger is changed each memory cycle when the C-7 clock trigger is turned ON (R_6 clock time) for the development of "A advance" and "B advance" pulses (see Figure 5-4). When the A/B trigger is turned ON at R_6 time of a particular memory cycle, an "A advance" pulse is developed at W_6 time. On the following memory cycle, the A/B trigger is turned OFF at R_6 time, and a "B advance" pulse is developed at W_6 time. "A advance" and "B advance" pulses are developed in this manner on alternate memory cycles as long as the clock continues to run. The "A advance" and "B advance" pulses are used to initiate alternate memory cycles (see Figure 5-5).

Figure 5-6 shows the timing chart for machine operation.

Figure 5-7 illustrates the functions and timings for the reading of a memory address from MARS into MAR and the writing of the address from MAR back into MARS incremented or decremented.

Figure 5-8 illustrates the functions and timings for reading from memory, setting MBR and MDR, and writing back into memory.

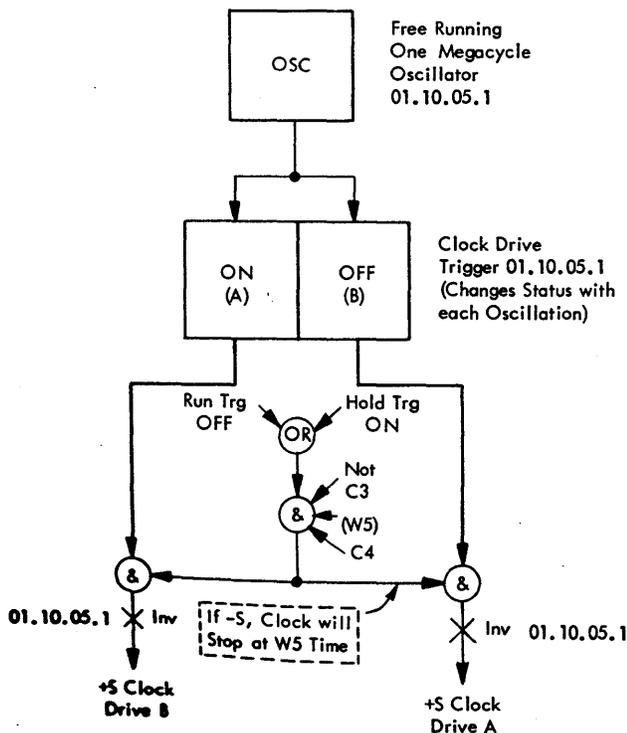


Figure 5-1. Clock Drive

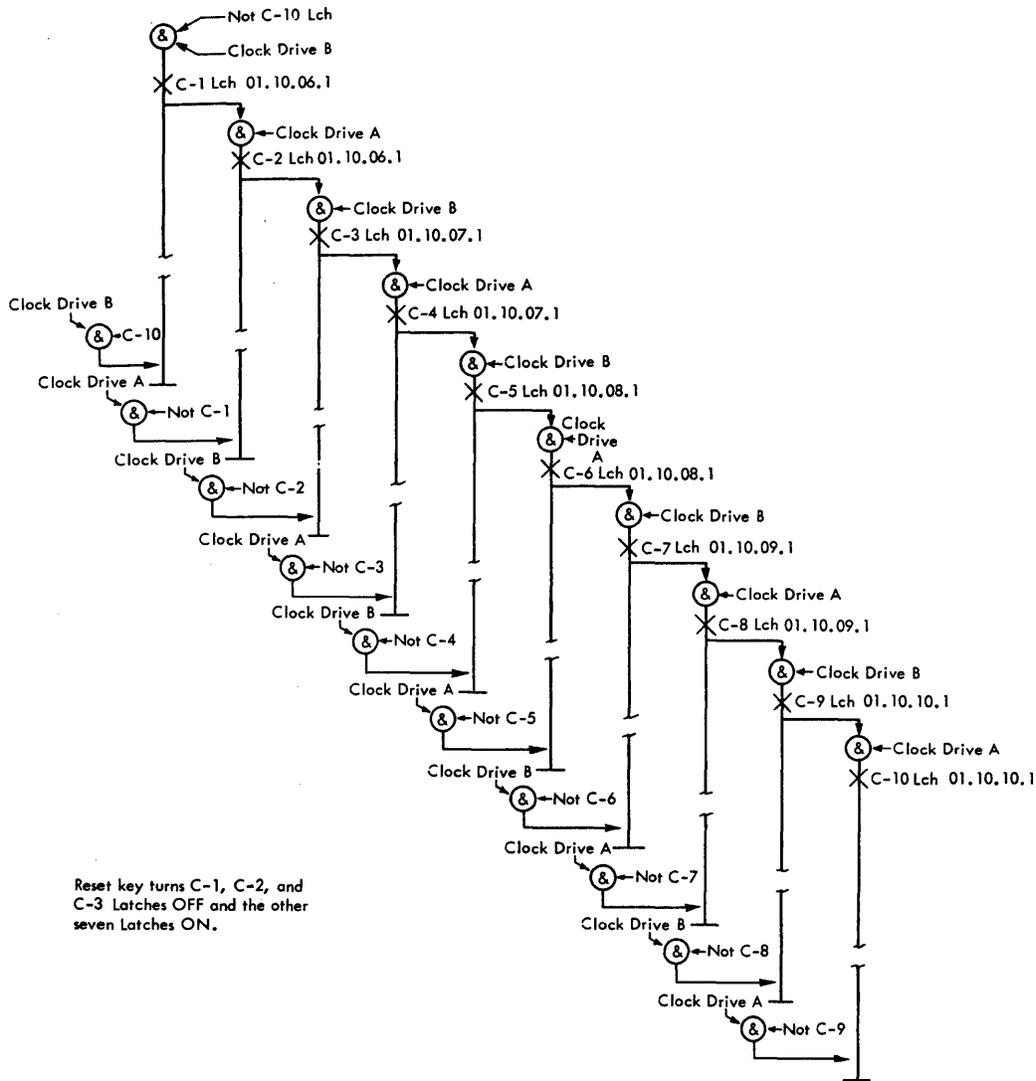


Figure 5-2. Clock Trigger Sequence

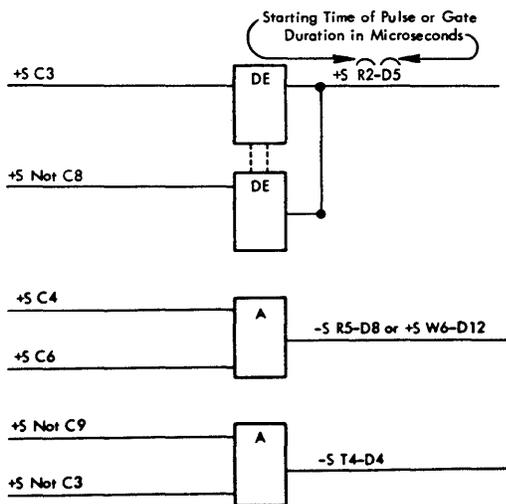


Figure 5-3. Development of Timing Gates

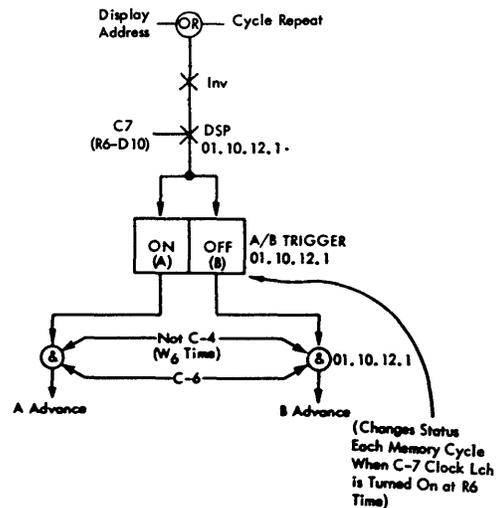


Figure 5-4. A/B Trigger

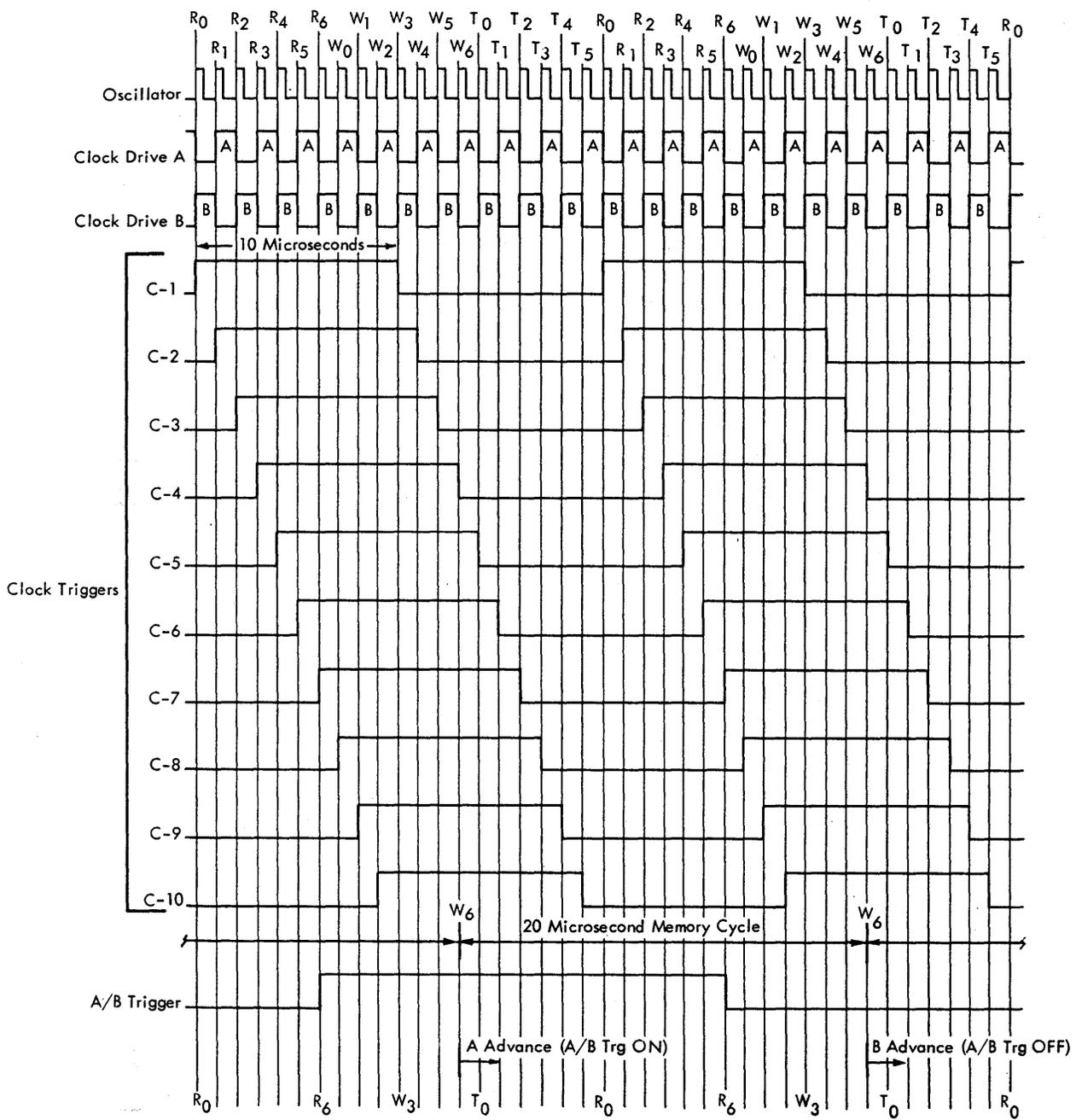


Figure 5-5. Clock Sequence and Timing Chart

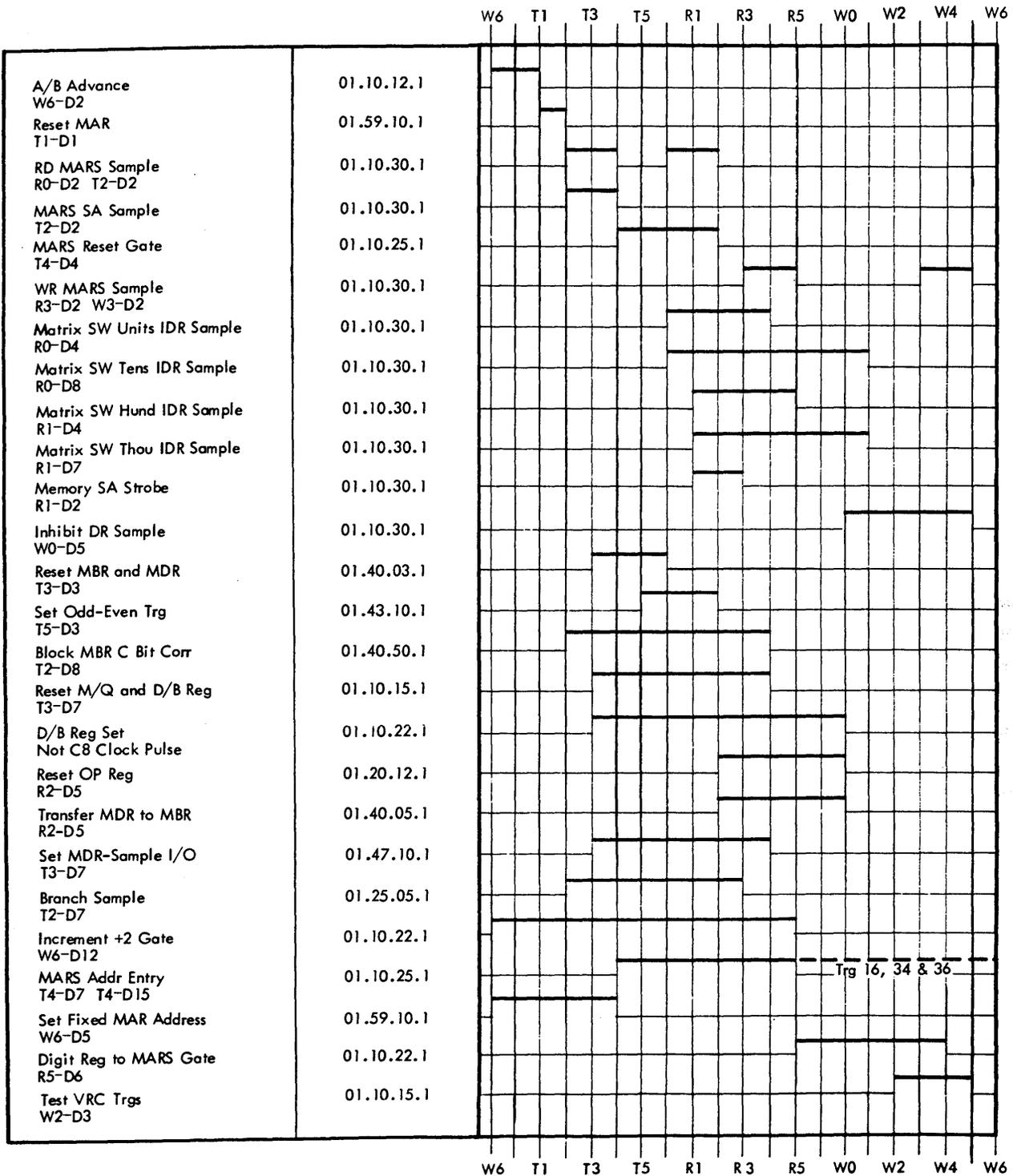


Figure 5-6. Timing Chart

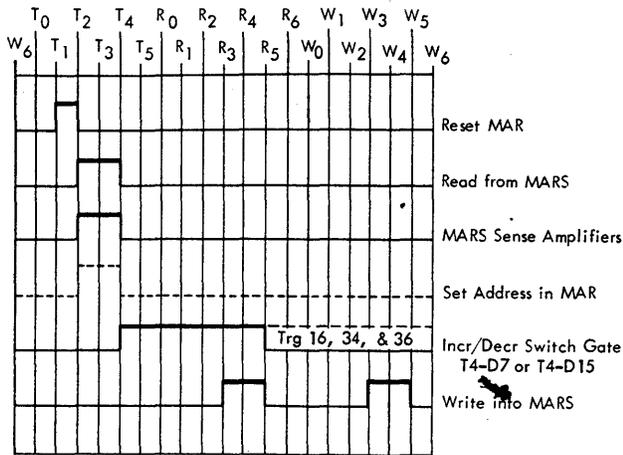


Figure 5-7. MARS-MAR-Increment/Decrement MARS Timing Chart

Memory

Memory (01.30.01.1 - 01.30.80.1) is a three-dimensional, coincident-current core system using ferrite cores. It consists of twelve planes of 10,000 cores each in a 100 x 100 square matrix. From front to back of the IBM 1620 machine the planes are arranged as follows: (see Figure 5-9)

- Plane 1 1 bit odd addresses
- Plane 2 1 bit even addresses
- Plane 3 2 bit odd addresses

- Plane 4 2 bit even addresses
- Plane 5 4 bit odd addresses
- Plane 6 4 bit even addresses
- Plane 7 8 bit odd addresses
- Plane 8 8 bit even addresses
- Plane 9 F bit odd addresses
- Plane 10 F bit even addresses
- Plane 11 C bit odd addresses
- Plane 12 C bit even addresses

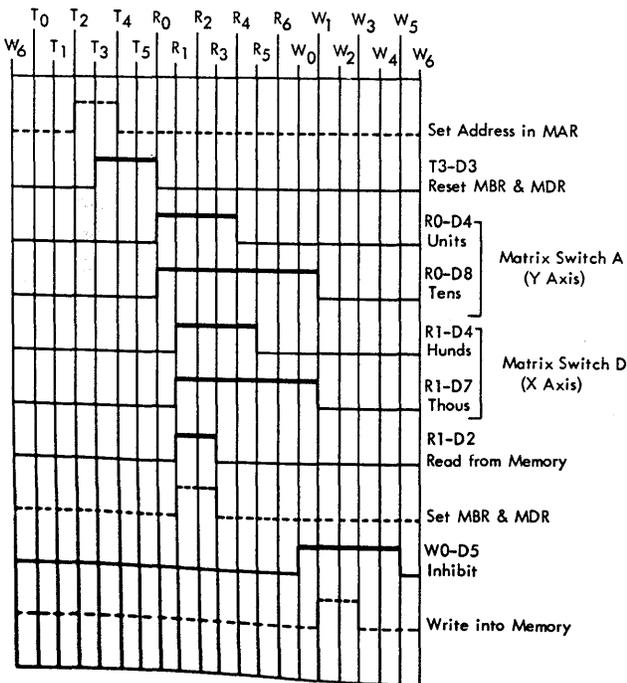
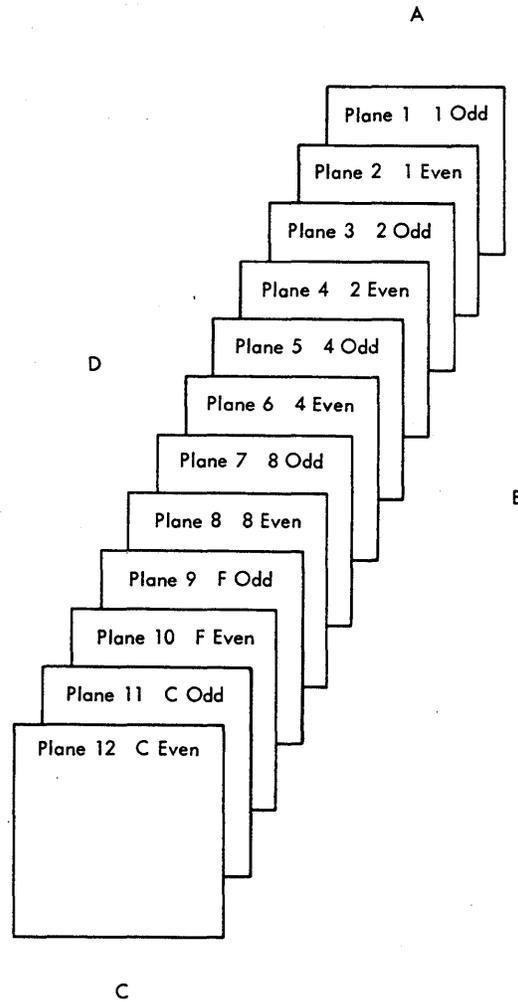


Figure 5-8. Memory "Read and Write" Timing Chart



Memory Array Viewed from Rear of 1620

Figure 5-9. Memory Array

Each memory core is threaded by an X and a Y address wire, a sense wire, and an inhibit wire. The X and Y wires form a rectangular coordinate system so that a particular pair of X and Y wires in one plane will intersect at just one core (see Figure 5-10). The core is selected by driving a half select current ($1/2 I_m$)

Memory Plane X Lines & Y Lines
Viewed from Rear of 1620

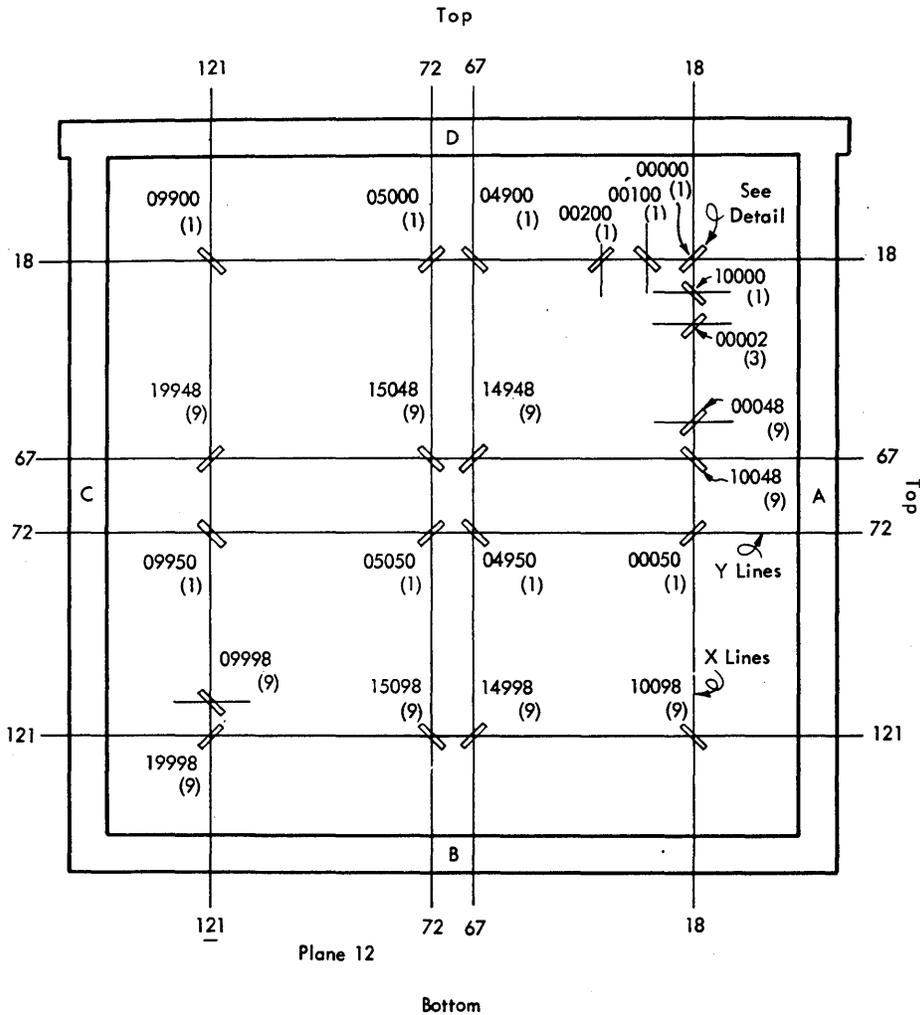


Figure 5-10. Memory Plane

along each of the two wires. Corresponding X and Y wires for each of the twelve planes are connected in series (see Figure 5-11).

Each of the 12 planes has a sense winding that in effect passes through every core in that plane. The sense winding of each plane is wound in a figure eight pattern and is physically divided into two parts, above 50 (>50) and below 50 (<50), resulting in two windings per plane. The output of each of these 24 sense windings is connected to a bipolar pre-sense amplifier. Each pair of pre-sense amplifiers associated with a given plane is then connected to a sense amplifier. The output of the sense amplifier is used to set a latch in MBR. Figure 5-12 shows the detail of a single memory core with all windings in place.

The output of a sense winding at "read" time can be either a positive going pulse or a negative going

pulse depending on the physical positioning of the cores in relation to the X and Y lines. The output of a sense winding is connected to a bipolar pre-sense amplifier which can accept pulses of either polarity. The sense winding will also have a positive going or negative going pulse induced in it at "write" time. Discrimination between "read" and "write" is accomplished by gating the pre-sense amplifier with a "memory sense amp-strobe", so that pulses sensed during "read" time only will be amplified and used to set an MBR latch.

To clear a location in memory, the memory sense amp-strobe is blocked from reaching the pre-sense amplifiers. Under this condition an MBR latch will not be set. Control of the memory sense amp-strobe is exercised by the "Read Y Point" or "Block Memory SA." When the machine function calls for clearing a single

Memory Array
 Schematic of Wiring for Addressing 00000-1
 Viewed from Rear of 1620

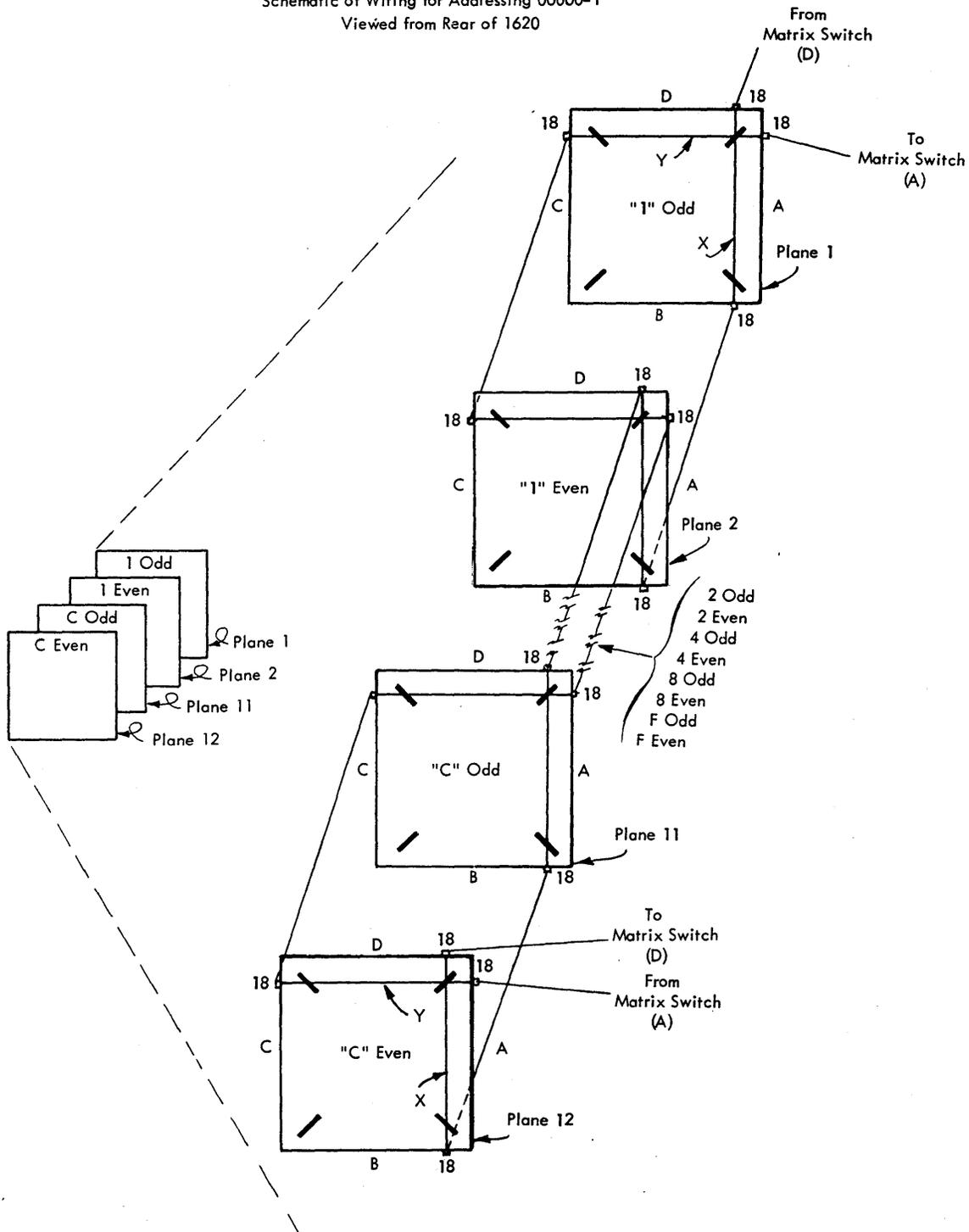
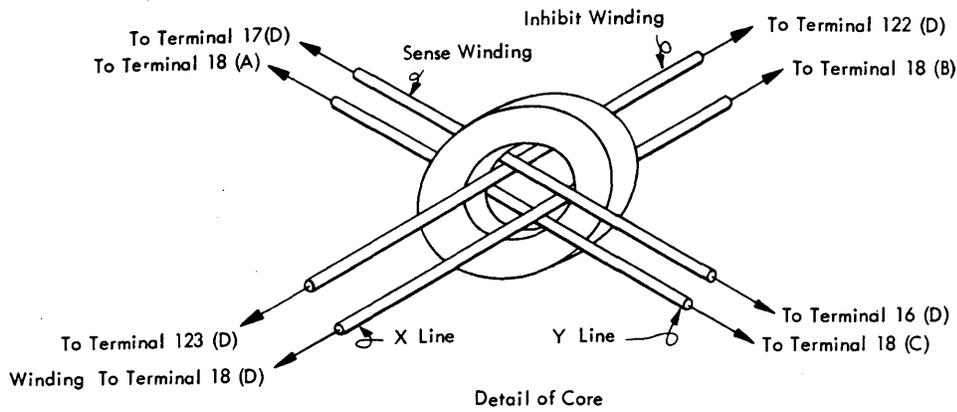


Figure 5-11. Memory Array—Schematic of Wiring



The Terminals Indicated Apply to the Twelve (12) Cores at Address 00000 and 00001

Figure 5-12. Detail of Core

location, odd or even, the "Read Y Point" is used. "Read Y Point" blocks the sense amp-strobe, either odd or even, depending on whether the address in MAR is odd or even. "Block Memory SA" blocks the sense amp-strobe to all pre-sense amplifiers (odd and even), regardless of the address in MAR. This effectively clears two locations in memory at one time (see Figure 5-15).

See Figures 5-13 and 5-15 for sense winding configuration and control of pre-sense amplifiers.

Each of the 12 planes also has an inhibit winding that in effect passes through every core in that plane. The inhibit winding of each plane is physically divided into two parts, resulting in two windings per plane. The two windings are wired in series and act as one. Both are controlled by one of the 12 latches in the memory buffer register (MBR) See Figure 5-14.

Figure 5-16 shows the circuits for controlling the greater than 50 (>50) trigger.

During the "read" part of the memory cycle, a current is sent along an X and a Y lines which intersects

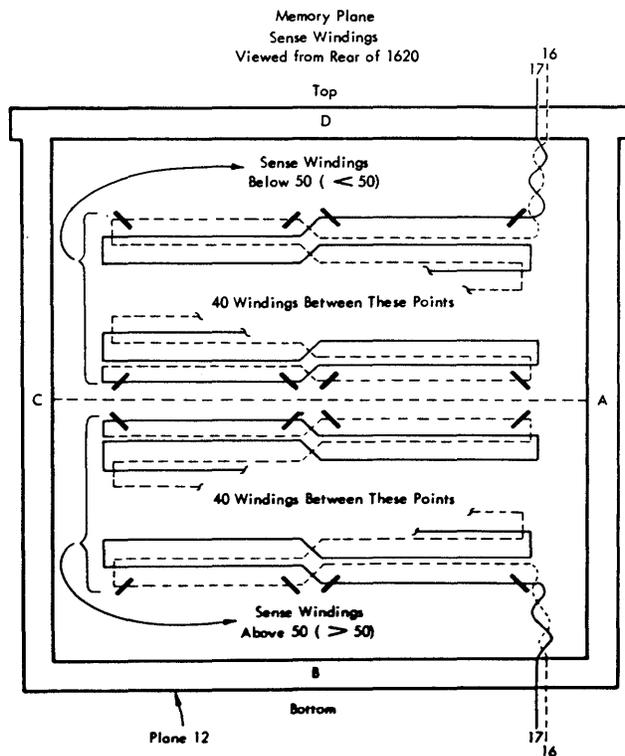


Figure 5-13. Memory Plane—Sense Winding

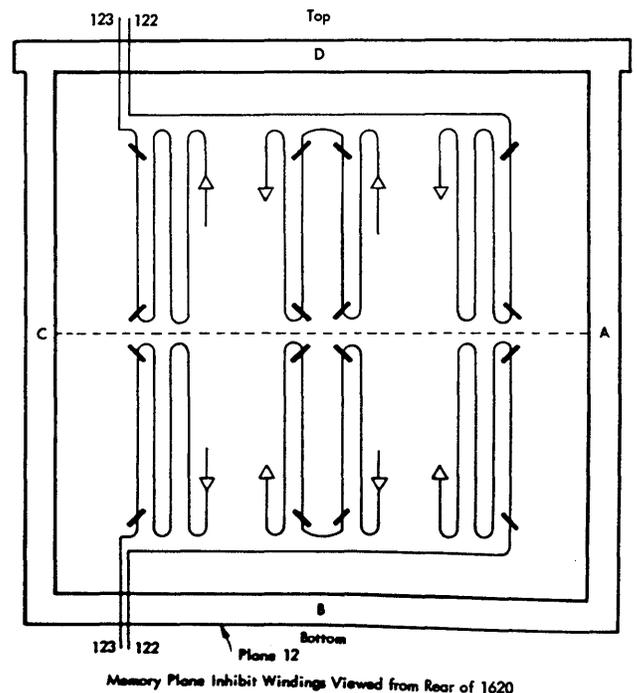


Figure 5-14. Memory Plane—Inhibit Winding

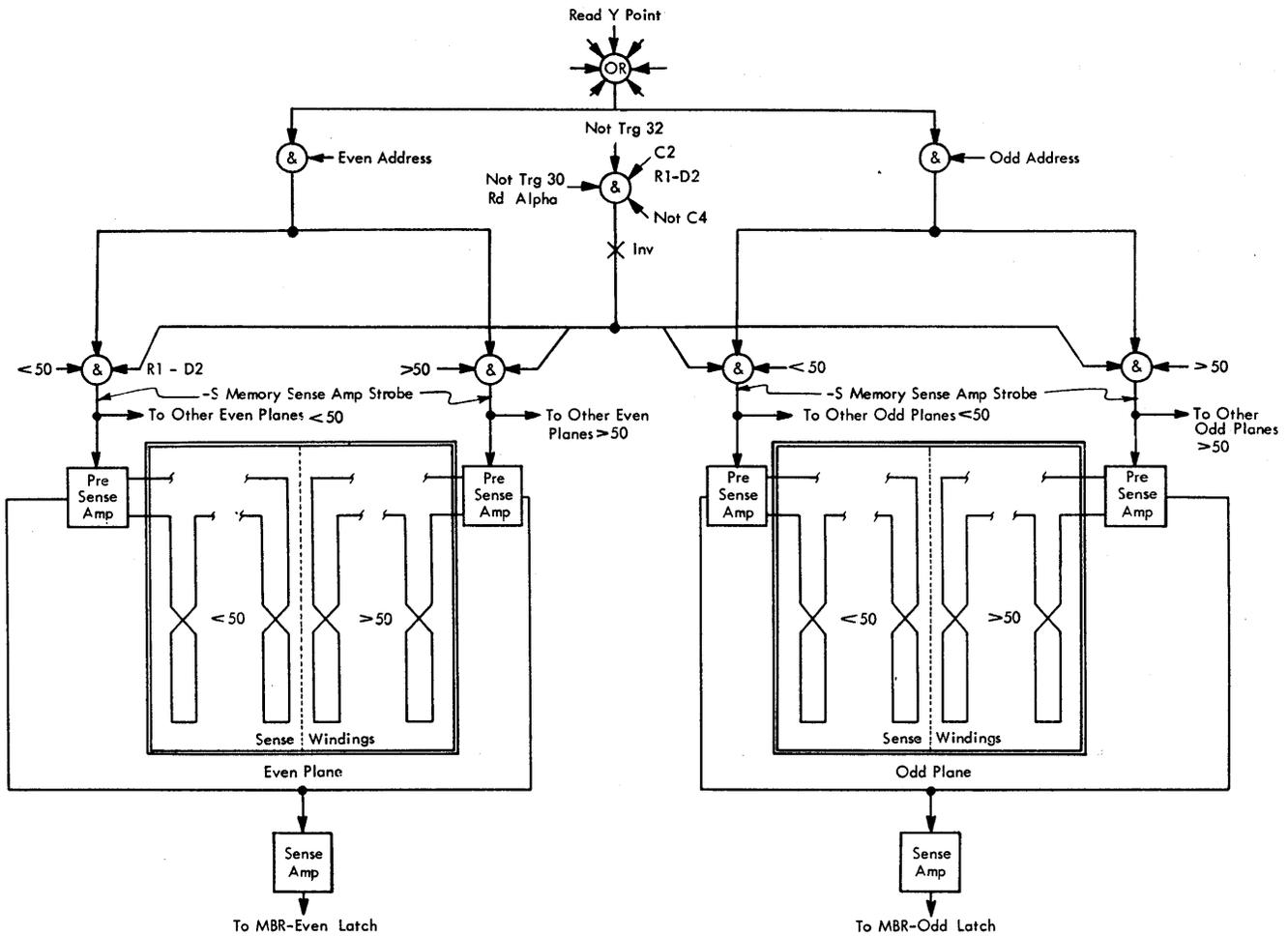


Figure 5-15. Readout Control of Memory Plane

at the selected core in each of the twelve planes. Each line carries $\frac{1}{2} I_m$ current.

The combined field of the X and Y lines reverses the magnetic polarity of the core if a "one" is stored but does not affect the polarity if a "zero" is stored.

When a "one" is read out, the change in polarity induces a voltage pulse in the sense winding which is rectified, amplified, and used to set a latch in MBR. The contents of twelve selected cores are thus stored in the twelve latches comprising MBR. The outputs of the MBR latches condition the inhibit drivers.

During the "write" portion of the memory cycle, a "one" is written back into all the selected cores unless prevented by the inhibit winding. "Writing" is accomplished by sending a $\frac{1}{2} I_m$ current along the selected X and Y lines in the opposite direction from the read currents. If it is not desired to write a "one" in the selected core of a particular plane, a $\frac{1}{2} I_m$ current is sent through the inhibit winding of that plane in such a direction as to oppose the write current. The core will therefore remain in its "zero" magnetic state.

The circuitry for selecting memory cores consists of a memory address register (MAR), decoding switches, and a magnetic core matrix switch. Each memory plane has 100 X lines and 100 Y lines and it is neces-

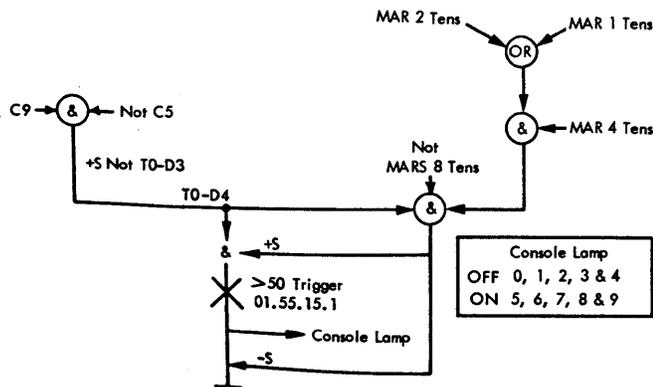


Figure 5-16. >50 Trigger

sary to select one line of each coordinate. This is accomplished by using two matrix switches, each of which has 100 switch cores arranged in 10 x 10 matrices. Here, as in the memory plane, it is necessary to select one line of each coordinate to select one switch core which in turn is connected to one line serving memory. Thus, by selecting four lines (two per matrix switch), it is possible to select any X and Y line in memory (see Figure 5-17).

The Y' lines of matrix switch A are addressed by the units position of the memory address register (MAR) and the Y'' lines are addressed by the tens position of MAR. The one bit trigger in the units position of MAR is not used for addressing the matrix switches. The one bit latch in the ten thousand (10K) portion of MAR is combined with the 8, 4, and 2 bit latches in the units position, to distinguish addresses above 10,000 from those below 10,000.

An address in MAR causes the decode switches to present a ground (zero volts) to one end of the Y' - Y'', X' - X'' lines in the matrix switch. The current source and current drivers under control of a gate will provide +30 volts to the other end. The current that flows in each line is capable of reversing the polarity of the switch core (I_m). A bias winding carrying I_m is wound on every switch core and is connected in such a manner as to try to maintain the "zero" condition of each switch core. When the Y' and Y'' or X' and X'' lines are energized (see Figure 5-8 for timing), one line overcomes the effect of the bias winding while the other causes the switch core to reverse its magnetic polarity, or flip, to the "one" state. When the switch core flips to the "one" state, it induces a current pulse by transformer action into the Y or X lines threading memory. This is the "read" current in memory as previously described. At "write" time for memory the Y' - Y'' and X' - X'' current is cut off. The bias winding then flips the switch core to its "zero" status. The return of the switch core to its "zero" status induces a current pulse into the X and Y lines in memory in a direction opposite to that of the "read" current. This current is the "write" current in memory as previously described.

Each odd memory address differs only from the next lower even address by having a one bit in the units position. Because the output of the one bit latch in the units position of MAR is not used for addressing memory, addresses 00000 and 00001 in MAR actually select the same X and Y lines in memory. The same thing is true of all even-odd addresses. Due to the 12 plane (2 digit) wiring scheme of memory, an even address will cause the readout of the digit at that specific address and the digit at the next higher odd

address. Conversely, an odd address will read out the digit at that specific address and the digit at the next lower even address.

Memory Buffer Register (MBR)

The memory buffer register is a 2-digit storage unit which is used as follows: (1) to store data that is read from memory for availability to other registers; (2) to store data that is to be written into memory; and (3) to control the entry (writing) of data into memory by exercising control over memory plane inhibit lines.

MBR is divided into MBR-even (01.40.07.1 to 01.40.09.1) and MBR-odd (01.40.17.1 to 01.40.19.1), each of which is capable of storing a single digit. Each is composed of six latches (C, F, 8, 4, 2, and 1). When memory is addressed to read out, the digit at the even address is stored in MBR-even and the digit at the odd address in MBR-odd.

Early in each memory cycle MBR is normally reset to nothing (C, F, 8, 4, 2, and 1 latches OFF) so that new data may be stored later in the cycle. Near the end of each cycle the two halves of MBR are checked independently for odd parity.

Figure 5-18 indicates data paths to and from MBR.

Figure 5-19 shows example of MBR latch control of inhibit function.

Memory Data Register (MDR)

The memory data register is a single digit storage unit whose functions are as follows: (1) to store the specific even or odd digit that is designated to be read from memory by the address in MAR for availability to other registers; and (2) to store data that is to be written into memory via MBR-even or MBR-odd.

MDR (01.45.05.1 to 01.45.07.1) consists of six latches, one for each of the C, F, 8, 4, 2, and 1 bits.

MDR is normally reset to nothing (C, F, 8, 4, 2, and 1 latches OFF) early in each memory cycle so that new data may be stored later in the cycle.

Figure 5-20 shows data paths to and from MDR.

Memory Address Register Storage (MARS)

MARS is a coincident-current core system consisting of a single plane in a 24 x 16 rectangular matrix (System Diagrams 01.57.05.1 to 01.57.32.1). The basic 1620 machine uses only a part (24 x 8) of the available storage.

Figure 5-21 shows the "read" winding. The "read" winding is threaded through each core twice, each

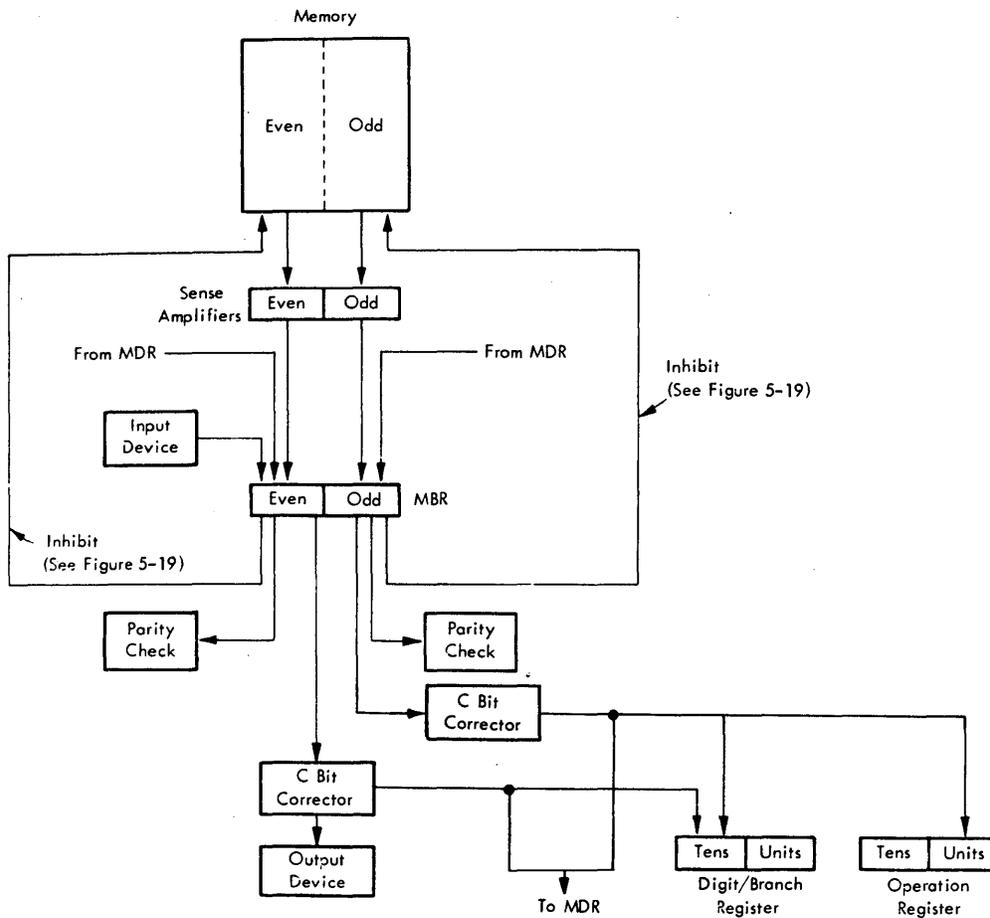


Figure 5-18. Data Paths To and From MBR

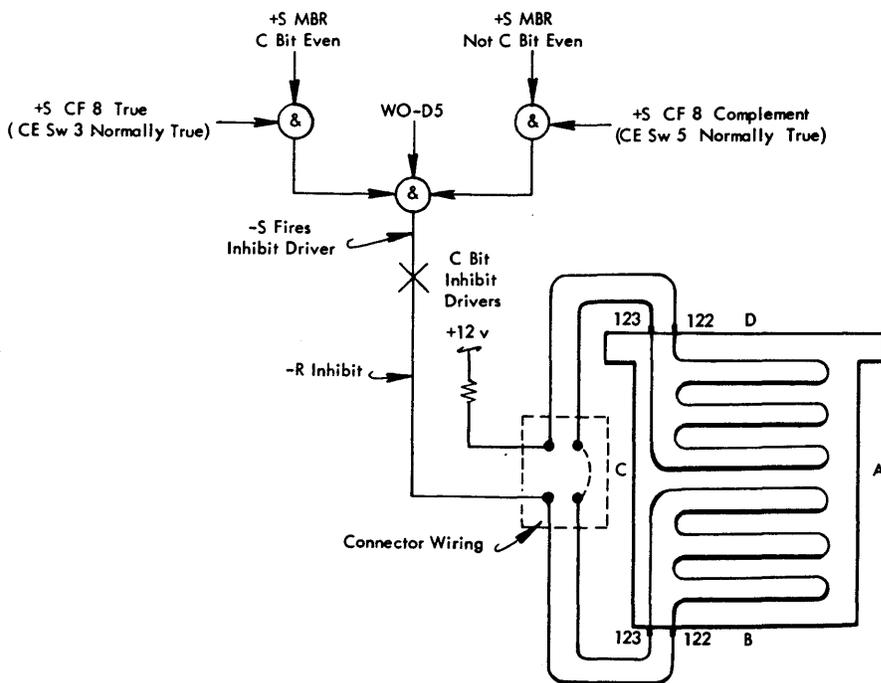


Figure 5-19. MBR Control of Inhibit Function

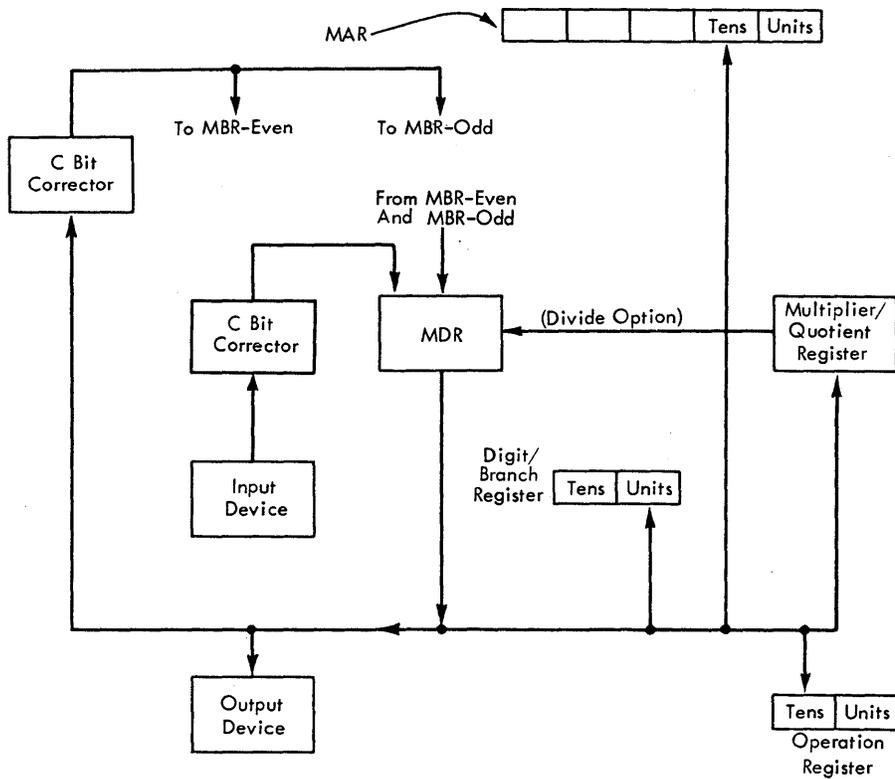


Figure 5-20. Data Paths To and From MDR

traverse supplying $\frac{1}{2} I_m$. The two turns supply full I_m which will flip the core to its "zero" state if a "one" was stored in it.

Figure 5-22 shows the sense winding. The two ends of the sense winding are connected to a bipolar sense amplifier which will accept positive or negative pulses. The output of the sense amplifier sets a trigger in MAR. At "read" time any cores that flip will induce a pulse in the sense winding. A core that is flipped during "write" time will also induce a pulse in the sense winding. To discriminate between "read" and "write" time, the sense amplifier is gated by the MARS sense amplifier gate. To reset a MARS register, the read winding for that register can be energized at a time when the MAR sense amp gate is not available. An induced pulse in the sense winding at this time is not detected.

Figure 5-23 shows the "write" lines and "bit" lines which form the coordinates for writing into a core. To write into a specific MARS register, coincidence of the "write" lines for that register and the "bit" lines is required. The bit drivers for each position (units, tens, hundreds, etc.) are controlled separately to permit writing into one or two positions of a MARS register without affecting the other positions.

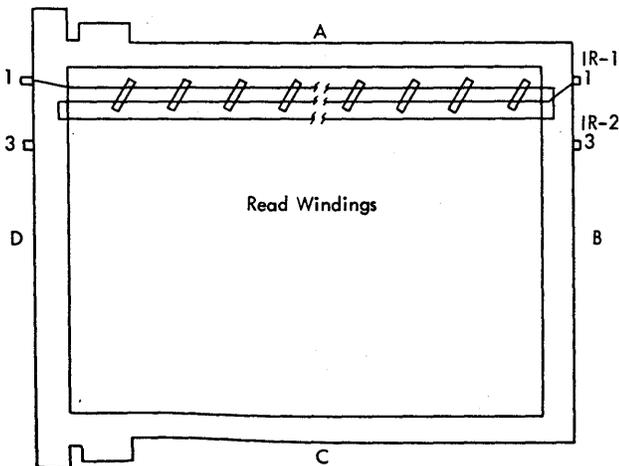


Figure 5-21. MARS Plane—Read Winding

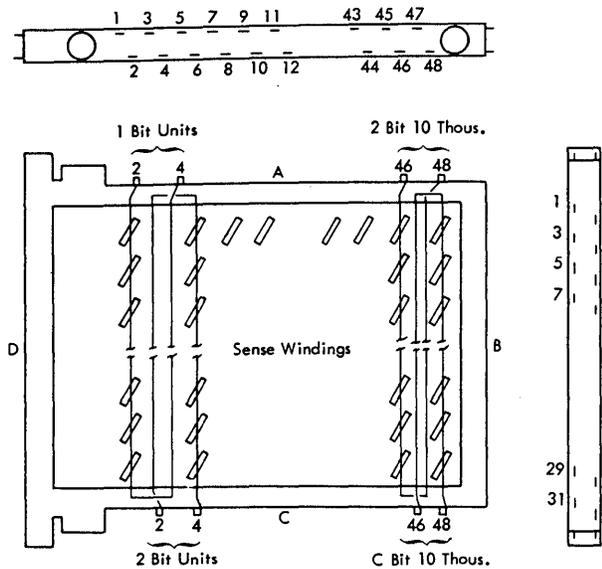


Figure 5-22. MARS Plane—Sense Winding

Memory Address Register (MAR)

The memory address register (01.57.05.1 - 01.57.32.1) is a 5-digit storage unit which is used to address memory (locate X and Y axes) for reading from memory or writing into memory. The 1 bit latch in the units position controls the odd-even trigger. The odd-even trigger has two functions: (1) to determine whether the content of MBR-odd or MBR-even is transferred to MDR to place the specific digit addressed by MAR in MDR; (2) to determine which memory sense amplifier (odd or even) is blocked for clearing a specific memory location.

The units, tens, hundreds and thousands positions of MAR each consist of five latches (C, 8, 4, 2, and 1).

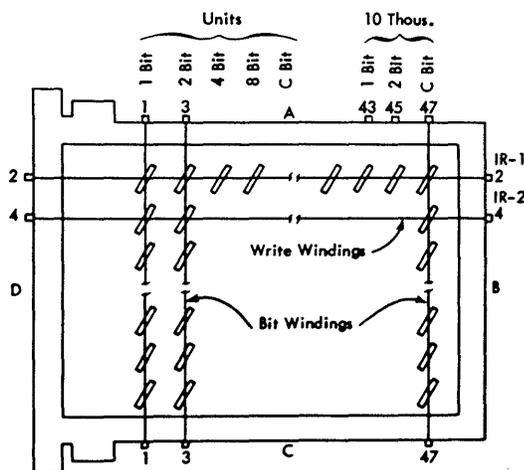


Figure 5-23. MARS Plane—Bit Winding and Write Winding

The ten-thousands position consists of four latches (C, 4, 2, and 1). The 1 bit latch in the ten-thousands position is used in addressing memory, and all four latches in the ten-thousands position are used in the detection of invalid memory addresses.

The ten-thousands position of MAR contains either a "one" or a "zero" for all valid memory addresses unless a 1623 is attached. Near the end of each memory cycle, MAR is checked for invalid digits in the ten-thousands position and for odd parity in each of the five positions.

MAR is normally reset to nothing (all latches OFF) at the beginning of each memory cycle, so that a new address may be entered from MARS for addressing memory.

In operations using the add table, 0, 0, 3 is set in the ten-thousands, thousands, and hundreds position, respectively, to define the add table area in memory. In operations using the multiply table, 0, 0 is set in the ten-thousands and thousands positions to complete the multiply table address. At the beginning of multiply and load dividend operations, "00080" is set in MAR to define the starting address for clearing the product area in memory. Depression of the insert key on the console sets "00000" in MAR to define the starting address for manual insertion of instructions.

Figure 5-24 shows data paths to and from MAR.

Increment/Decrement Switch

The status of the decrement latch (01.60.05.1) determines whether addresses in MAR are incremented or decremented before they are written in MARS by controlling the increment/decrement switch (OFF-increment, ON-decrement).

The increment/decrement switch (System Diagrams 01.55.02.1 to 01.55.45.1) is a network of AND and OR switches which provide increment plus 1, increment plus 2, or decrement minus 1 functions. A decrement -2 is supplied with the Indirect Addressing feature.

Increment plus 1 and decrement minus 1 are the normal functions of the increment/decrement switch, depending on the status of the decrement latch.

Where it is required to read out to MAR and return an address unchanged to MARS, a bypass around the increment/decrement switch is available.

Where it is required that an address in MAR be increased by 2 before placing it in a MARS register, an increment +2 function is available.

Figure 5-25 shows the data paths to, from, and bypassing the increment/decrement switch.

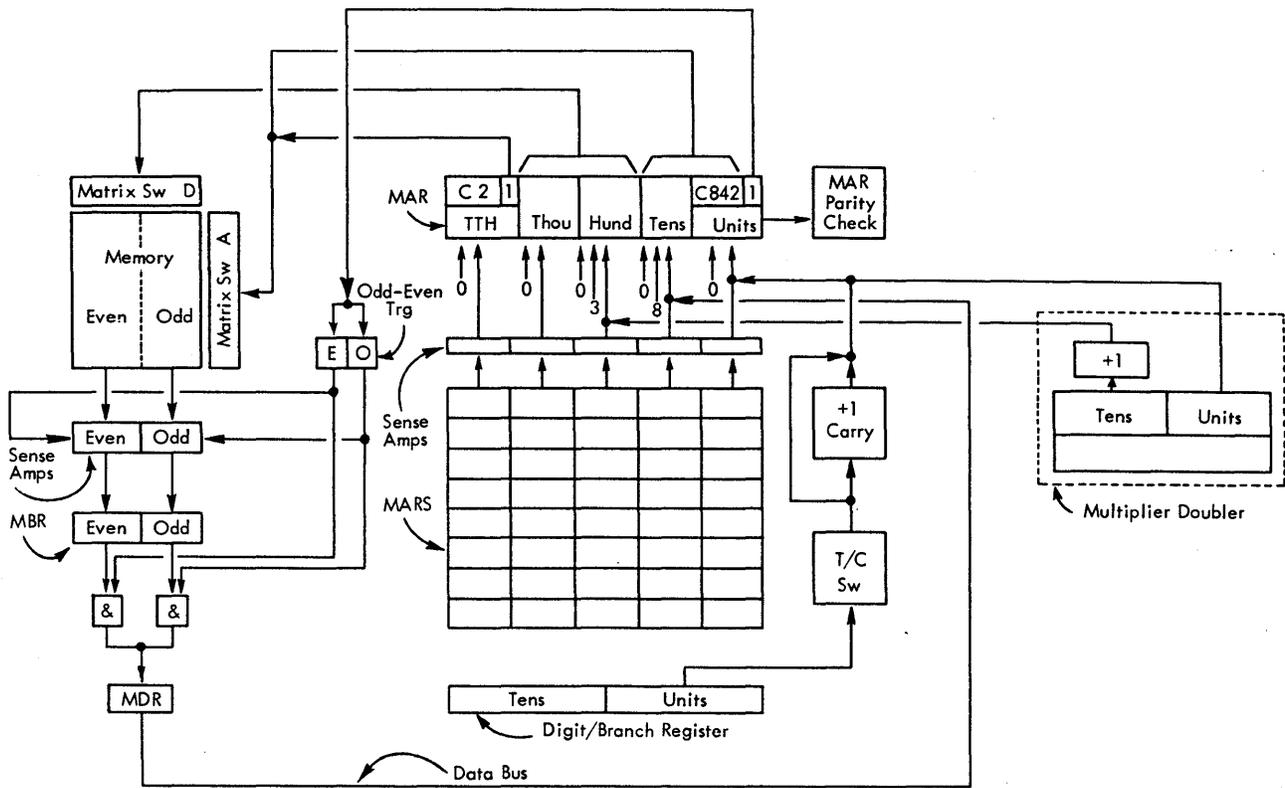


Figure 5-24. Data Paths To and From MAR

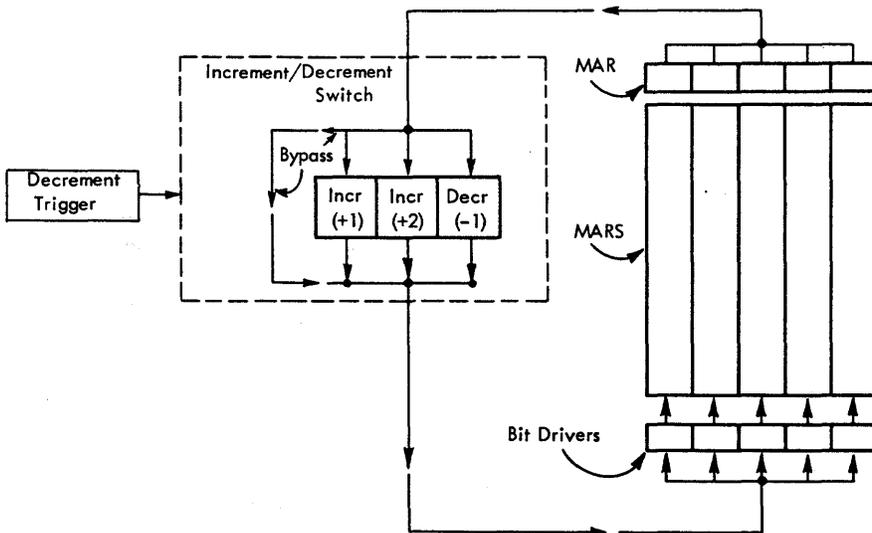


Figure 5-25. Data Paths To, From, and Bypassing Incr/Decr Switch

Operation Register and Decoder

The operation register (System Diagrams 01.20.07.1 to 01.20.12.1) is a 2-digit storage unit which is used to store the operation code (digits O_0 and O_1) of each instruction for the duration of the operation specified by the instruction. The contents of the operation register are decoded to determine the operation to be performed. The output of the decode network controls circuits to accomplish the operation objectives.

The units position of the operation register consists of five latches (C, 8, 4, 2, and 1 bits) and the tens position of the operation register consists of four latches (C, 4, 2, and 1 bits). The decoder (System Diagrams 01.20.13.1 to 01.20.42.1) is composed of a network of AND and OR switches connected to the outputs of the latches comprising the operation register.

The operation register is reset to zeros (C bit latches ON; 8, 4, 2, and 1 bit latches OFF) prior to entry of a new operation code. The C bit latches remain ON or are turned OFF as required by the bit combinations of the new operation code.

Figure 5-26 shows data paths to and from the operation register and decoder.

Digit/Branch Register and Decoder

The digit/branch register (System Diagrams 01.50.07.1 to 01.50.19.1) is a 2-digit storage unit which is used for: (1) loading of P and Q addresses in MARS during the instruction cycle; (2) storing factors in arithmetic operations for use in the development of add table and multiply table addresses; (3) storing the Q_s and Q_c digits of read, write, control, branch indicator, and branch no indicator instructions.

Each position (units and tens) of the digit/branch register consists of five latches (C, 8, 4, 2, and 1 bits).

Each digit/branch register position is controlled to store data or to reset to nothing (that is, C, 8, 4, 2, and 1 latches OFF), independent of the other position.

In multiply operations, it is required that the multiply table digit in the tens position of the digit/branch register be transferred to the units position for processing; therefore, a data flow path is provided to accomplish this transfer. The contents of the digit/branch register are decoded to determine: (1) the input or output unit to be used in read, write, and control instructions; and (2) the program sense switch or indicator to be interrogated in branch indicator and branch no indicator instructions. The decoder is composed of a network of AND and OR switches connected to the outputs of the latches comprising the digit/branch register.

Figure 5-27 shows the data paths to and from the digit/branch register positions.

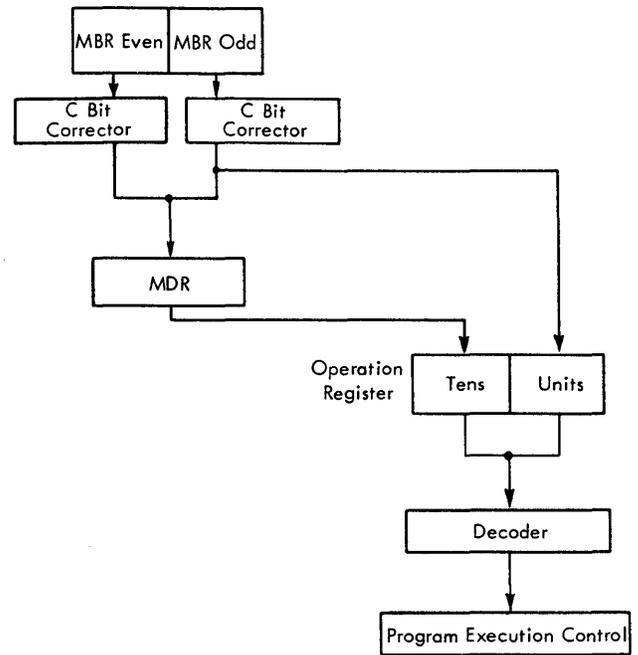


Figure 5-26. Data Paths To and From Operation Register and Decoder

True/Complement (T/C) Switch

The true/complement switch (01.60.34.1 - 01.60.38.1) is used on arithmetic operations in the development of add table addresses to be set in MAR.

The T/C switch consists of a network of AND and OR switches connected to the outputs of the latches comprising the units position of the digit register. Control of the T/C switch is exercised by the status of the true/complement trigger (01.63.20.1). With the T/C trigger ON, "true" control exists and digits presented to the T/C which appear unchanged at the output of the switch. With the T/C trigger OFF, "complement" control exists and the 9's complement of digits presented to the T/C switch appear at the output of the switch.

Digits at the output of the T/C switch bypass the carry switch if the carry in trigger is OFF. Digits (true or complement) at the output of the T/C switch are increased in value by one by the carry switch if the carry in trigger is ON, denoting: (1) a carry from addition of the two previous digits, or (2) the add table address for the units position of a sum or difference is being developed on a complement operation. (If the output from the T/C switch is a "9" and the carry in trigger is ON resulting in a zero (9 + carry) output from the carry switch, the carry out trigger will be turned ON to provide for increasing the value of the next digit to the left by one.)

Figure 5-28 shows the data paths to and from the T/C switch and carry switch.

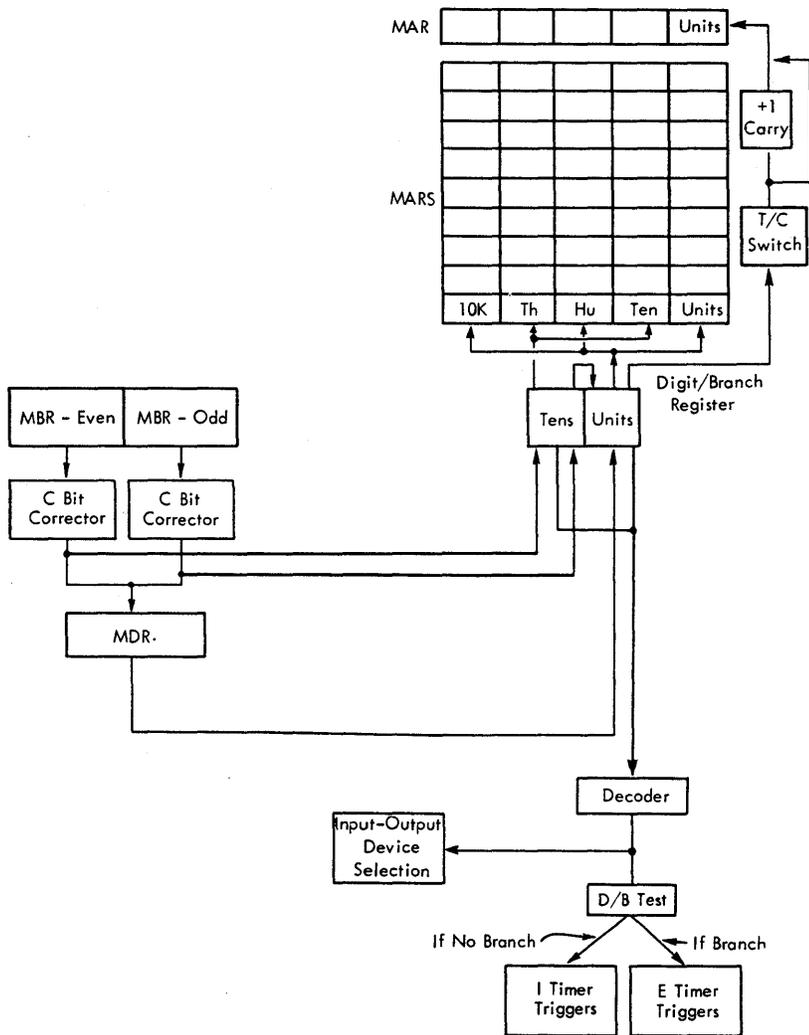


Figure 5-27. Data Paths To and From Digit Register Positions

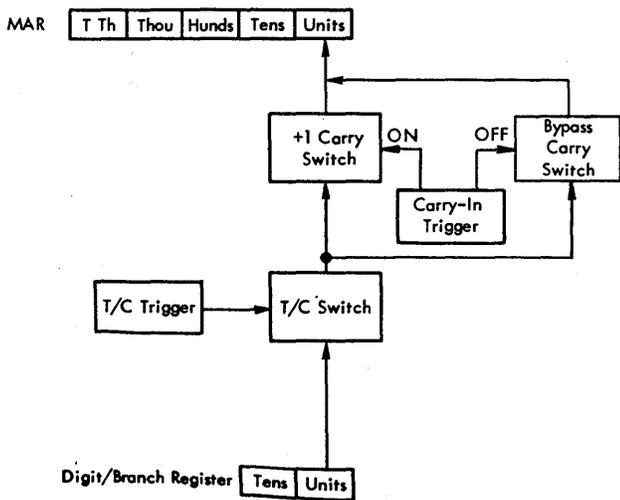


Figure 5-28. Data Path To and From True/Complement Switch and Carry Switch

Multiplier/Quotient Register

The multiplier/quotient register (01.62.50.1-01.62.52.1) is a single digit storage unit which is used in multiply operations to store each digit of the multiplier until it has been used with each digit of the multiplicand. It is also used to store the quotient digit developed on a divide operation. The multiplier/quotient register consists of five AND latches, one for each bit (C, 8, 4, 2, and 1).

The multiplier/quotient register is reset to zero (C bit latch on; 8, 4, 2, and 1 bit latches OFF) early in the memory cycle in which a new multiplier digit is to be stored. The C bit latch remains ON or is turned OFF as required by the bit combinations of the new multiplier digit.

Figure 5-29 shows data paths to and from the multiplier register.

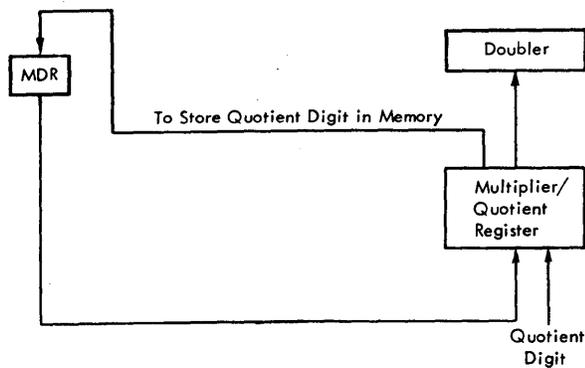


Figure 5-29. Data Paths To and From the Multiplier Register

Doubler

The doubler (01.62.55.1 - 01.62.60.1) is used on multiply operations in the development of multiply table addresses to be set in MAR. The doubler consists of an AND network gated by trigger 35 and connected to the outputs of the triggers comprising the multiplier/quotient register. Its function is to double the value of the multiplier digit and increase the value of the resulting tens position digit by one.

Figure 5-30 shows data paths to and from the doubler and gives examples for specific multiplier digits.

C Bit Correctors

C bit correctors (01.40.50.1 and 01.45.50.1) perform the function of maintaining correct parity where it is desired to clear an F bit from a digit or set an F bit with a digit. Any digit in MDR will be correct, exclusive of the F bit. Although an even number of bits may exist in the MDR latches, the console lights will always display an odd number of bits.

F bits are not used in the operation register, the

digit/branch register, the memory address register (MAR), or the multiplier/quotient register. Data is routed from memory to these registers through MBR-even, MBR-odd, or MDR.

MBR to MDR

A digit will have odd parity in MBR but a C bit corrector may change the C bit to maintain odd parity "exclusive" of the F bit in MDR (see Figure 5-31). The F bit and C bit corrector network between MBR-even or MBR-odd and MDR functions as follows:

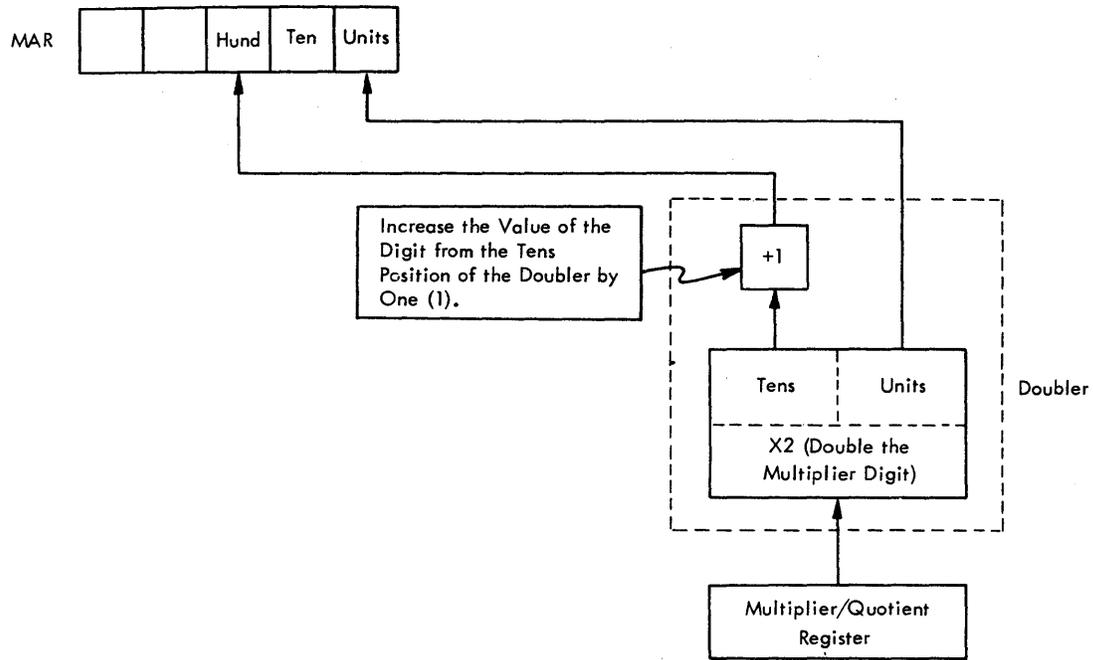
1. An F bit in MBR will turn on the MDR F bit latch.
2. An 8, 4, 2, or 1 bit in MBR will turn on the respective MDR latch.
3. The MBR F bit and C bit latches are analyzed by the "exclusive or" network.
 - a. If they are both on or both off, no C bit will be set into MDR.
 - b. If either one is on and the other off, a C bit will be set into MDR.

Because digits in MDR have correct parity exclusive of the F bit, these digits may be transmitted from MDR to the operation register, D/B register, MAR, and the M/Q register without additional C bit correction.

MDR to MBR

Since the F bit does not enter into the parity of MDR, the C bit must be corrected in the process of transferring data from MDR to MBR. Data flow from MDR to MBR is shown in Figure 5-32. Note that the MDR C bit console lamp operates from the output of the C bit corrector, thus indicating odd parity in MDR although an even number of latches may be on. The C bit is corrected through an exclusive or circuit as follows:

1. If the MDR F bit and C bit latches are both on or both off, no C bit will be set into MBR.
2. If either one (F bit or C bit latch) is on and the other is off, a C bit will be set into MBR.



Multiplier Digit	Doubler			MAR				
	Tens	Tens +1	Units	T Th	Thous	Hunds	Tens	Units
1	0	1	2			1		2
4	0	1	8			1		8
5	1	2	0			2		0
8	1	2	6			2		6

Figure 5-30. Multiplier Doubler

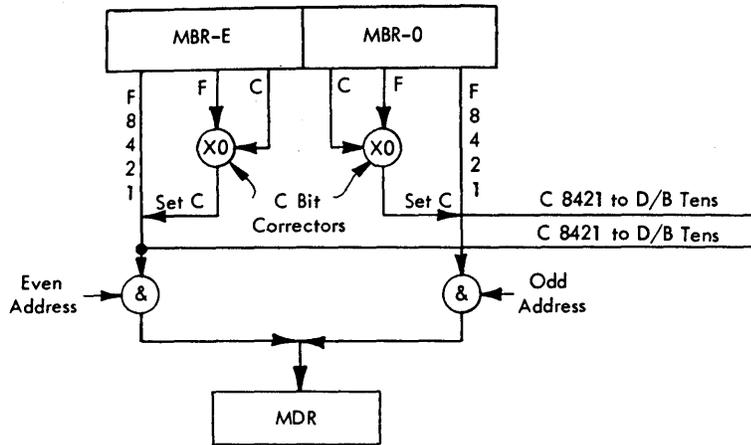


Figure 5-31. C Bit Corrector—From MBR-Odd and MBR-Even

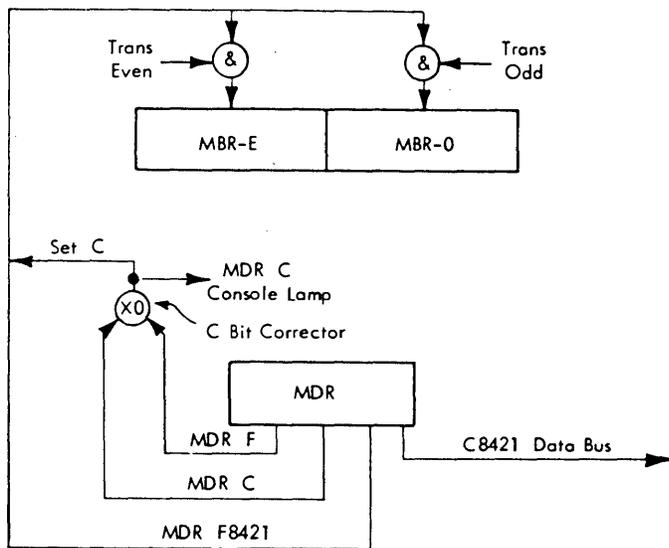


Figure 5-32. C Bit Corrector—From MDR

Section 6 Console and CE Panel

Console

The console is an integral part of the 1620 Computer unit and consists of the typewriter and the console panel with its indicator lights, switches, control keys, and signal lights.

The typewriter is both an input and an output device. As an input device, the typewriter keyboard is used to enter instructions and data into memory. As an output device, the typewriter is used to print information from memory. Typewriter functions are described in Section 13, Input-Output Devices, and Section 14, Input-Output Operations.

The console panel is shown in Figure 6-1 and the functions of its lights, switches and keys are discussed in this Section.

Register Displays

The contents of the registers within the computer are displayed on the console panel by means of small incandescent lights. Each bit in each register position is represented by a light, and the light is ON when the bit which it represents is present in the digit displayed. The following registers are displayed:

Memory Address Register (MAR): Five rows of five indicator lights each display the bit configuration of the five-digit address that is placed in MAR.

Memory Buffer Register (MBR): Two rows of six indicator lights display the bit configuration of the addressed digit and its associated digit. The digit in the even numbered memory location is displayed in the row of lights labeled "E"; the digit in the next higher numbered memory location is displayed in the row of lights labeled "O".

Memory Data Register (MDR): One row of six indicator lights displays the bit configuration of the specific digit that is addressed. The digit displayed in the MDR lights is duplicated in the MBR-even or MBR-odd lights, depending on whether the address in MAR is even or odd.

Digit/Branch Register: Two rows of five indicator lights display the bit configuration of digits placed in the units and tens position of the digit/branch register.

Operation Register: Two rows of five indicator lights display the bit configuration of the two-digit operation code placed in the operation register during an instruction cycle.

Multiplier/Quotient Register: One row of five indicator lights displays the bit configuration of each successive multiplier or quotient digit while it is present in the register.

Machine Check Trigger Lights

The ON and OFF status of check triggers within the computer is represented by small incandescent lights on the console panel. The indicator light associated with a check trigger is ON when the check trigger is ON and OFF when the trigger is OFF.

When a check trigger is turned ON, the computer continues to operate or stops, depending on the setting of the console check switch associated with the check trigger. If the check switch is set to STOP, the computer stops after detection of the error condition. If the check switch is set to PROGRAM, the computer continues to operate and the status of the check trigger can be interrogated by branch indicator or branch no indicator instructions. If a check trigger that is ON is interrogated, the trigger is turned OFF as a result of the interrogation. Depression of the console reset key turns OFF all check triggers that are ON.

The functions of the various check triggers are described as follows:

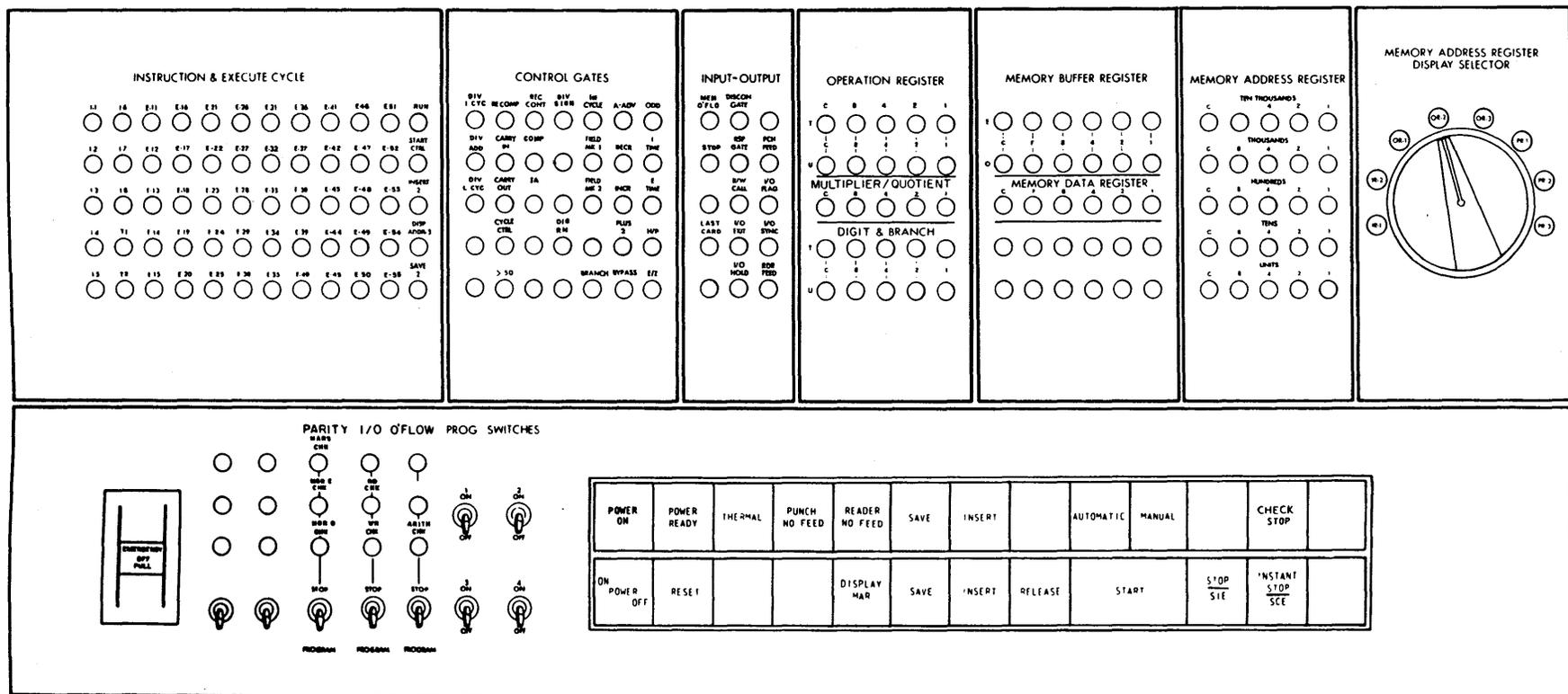
OVERFLOW CHECK

The overflow check trigger and console indicator light are turned ON when an overflow condition occurs in an add, subtract, compare, or divide operation. The overflow check switch on the console is associated with the overflow check trigger.

MEMORY ADDRESS REGISTER CHECK

The MAR check trigger and console indicator light are turned ON when a digit with incorrect parity or an invalid address is present in MAR. A MAR check will stop the computer at the end of the memory cycle in which the error occurs unless CE switch 9, MAR stop bypass, is operated. The console switch has no STOP or PROGRAM function with a MAR check.

NOTE: Each of the four lights in the following group is associated with triggers which check parity at points in the system data flow. MBR-E and MBR-O checks are associated with a console check switch. Read and write checks are associated with a console check switch.



61 Figure 6-1. Console

MEMORY BUFFER REGISTER—EVEN CHECK

The MBR-even check trigger and console indicator light are turned ON when the digit in MBR-even has incorrect parity.

MEMORY BUFFER REGISTER—ODD CHECK

The MBR-odd check trigger and console indicator light are turned ON when the digit in MBR-odd has incorrect parity.

READ (RD) CHECK

The RD check trigger and console indicator light are turned ON when a character with incorrect parity is presented to the input translator.

WRITE (WR) CHECK

The WR check trigger and console indicator light are turned ON if a character with incorrect parity is presented to an output device, an even number of typewriter bit relays are energized, an even number of holes are punched by the paper tape punch, or the output device fails to return a response signal to the computer.

Machine Timer Trigger, Auxiliary Trigger, and Control Gate Lights

The ON and OFF status of instruction cycle timer triggers (I timers), execution cycle timer triggers (E timers), certain auxiliary triggers, and certain control gates within the computer is represented by small incandescent lights on the console panel. The indicator light is ON when the trigger or control gate is ON, and OFF when the trigger or control gate is OFF.

These lights are grouped in the three upper left sections of the console panel and are titled "Instruction and Execute Cycle," "Control Gates," and "Input-Output." They are individually labeled for identification and are primarily intended for use by the Customer Engineer in machine trouble analysis. However, some of the indicator lights such as the high/plus (H/P) and the equal/zero (E/Z) in the Control Gates section, may have significance to the machine operator as well as to the Customer Engineer.

Switches

MAR DISPLAY SELECTOR SWITCH (01.05.05.1)

This eight-position rotary switch is used to select one of the eight MARS registers for display in the MAR indicator lights. When selection of the particular MARS register has been made by means of this switch, depression of DISPLAY MAR causes the contents of the selected register to be displayed in the MAR display lights. This switch is also used in conjunction with CE switch 7 (increment).

CHECK SWITCHES

The parity check, overflow check, and I/O check switches are associated with check triggers within the computer and provide a means of directing the computer to stop or continue with the program when a check trigger is turned ON. If a check trigger is turned ON with the associated check switch set to STOP, the computer stops after detection of the error condition. If the check switch is set to PROGRAM, the computer continues to operate, and the status of the check trigger can be interrogated by branch indicator or branch no indicator instructions.

Parity Check Switch: (01.05.02.1). This switch determines the function of the MBR-odd and/or MBR-even parity check triggers. When the parity check switch is set to STOP and either the MBR-even or MBR-odd check trigger is turned ON, the computer stops at the end of the memory cycle in which the parity error is detected, unless the execution cycle of an input-output operation is in progress. If one or both of the triggers is turned ON during an input-output operation, the computer stops only upon completion of the read or write operation.

I/O Check Switch: (01.05.02.1). When the I/O check switch is set to STOP and either the read (RD) or write (WR) check trigger is turned ON, the computer stops only upon completion of the read or write operation. (If a parity error is detected when punching paper tape, the computer remains in the automatic mode but suspends operation with the incorrect character still under the tape punches, without regard to the setting of the I/O check switch.)

Overflow Check Switch: (01.05.02.1). When the overflow check switch is set to STOP and the overflow check trigger is turned ON, the computer stops upon completion of the operation during which the overflow condition is detected.

PROGRAM SWITCHES (01.05.02.1)

The four program (console sense) switches, numbered 1 through 4, are toggle switches with ON and OFF settings for use with branch indicator and branch no indicator instructions. When a switch is set to ON and interrogated by a branch indicator instruction, the branch occurs. If the switch is set to OFF, the branch does not occur and the next instruction in sequence is executed.

If a branch no indicator instruction is used to interrogate one of the switches, the branch occurs when the switch is set to OFF.

CONSOLE POWER SWITCH (01.05.03.1)

The power switch on the console is a double pole, single throw toggle switch. When turned ON, AC voltages are applied to the power supplies and DC voltages are properly sequenced.

When the power switch is set to OFF, AC voltages are removed from the power supplies and DC voltages are sequenced OFF.

CONSOLE IMMEDIATE OFF SWITCH (01.05.03.1)

When the immediate off switch on the console is operated, all power is immediately removed from the machine. **IT IS TO BE OPERATED ONLY IN CASE OF AN EMERGENCY.** If operated, the switch must be reset by the Customer Engineer after pivoting the console panel forward to permit access to the back of the switch.

Control Keys

RESET (01.05.03.1)

Depression of the reset key restores all triggers, indicator lights, and signal lights to their initial or reset condition.

DISPLAY MAR (01.05.01.1)

The objective of the display MARS operation is to read into MAR the address contained in a selected MARS register for display in the MAR indicator lights. The operation can be performed only when the computer is in the manual mode. Depression of the STOP/SIE key places the computer in the manual mode.

To display a MARS register, the MAR display selector switch is set to the desired register and the display MAR key is depressed.

The address in a second MARS register may be displayed by turning the switch to select the desired register and by again depressing the display MAR key.

Each address that is displayed during a display MARS operation is written back unchanged into the MARS register from which it came.

NOTE: When the INSTANT STOP/SCE key is used to stop the machine, both the manual mode and the automatic mode lights will be on, except when the stop occurs on the last E-timer trigger of the cycle prior to entering I-cycles (I-cycle entry status). With the computer stopped in the I-cycle entry status, only the manual light will be on. When both lights are on, the reset key must be depressed before depressing the display MAR key. Depressing the reset key causes any timer trigger left ON from the previous execution cycle to be turned off. The automatic mode light is turned off because the Start 1 trigger is reset off.

Figure 6-2 shows a function chart for the display MARS operation.

SAVE (01.05.01.1)

The machine operator can interrupt the execution of a program to enter instructions for another routine and execute that routine. When the objectives of the interruption have been accomplished, execution of the program can be resumed.

The objective of the save operation is to retain the address of the next instruction to be executed when the program is interrupted.

The save operation can be performed only when the computer is in the manual mode. Execution of the program is interrupted and the computer is placed in the manual mode by depression of the STOP/SIE key.

Depression of the save key then stores the address of the next instruction to be executed in the PR-1 MARS register and turns ON the save signal light.

The last instruction entered into the computer for the routine must be a branch back instruction. Execution of that instruction turns the save light OFF and directs the computer to resume execution of the program, beginning with the instruction at the saved address.

Instructions entered and executed for the routine should not demand multiply, multiply immediate, divide, divide immediate, load dividend, or load dividend immediate operations. The execution of such an instruction would destroy the saved address in PR-1.

The reset key should not be depressed between the time that the save key is depressed and the first branch back instruction is executed because depression of the reset key turns OFF the save control trigger. The saved address in PR-1 would not then be used in resumption of the program. (See Branch Back — Code 42, Section 11.)

Figure 6-3 shows a function chart for the save operation.

INSERT (01.05.01.1)

The objective of the insert operation is to activate the typewriter keyboard for direct entry of instructions into memory. Instruction digits are stored at memory location 00000 and successively higher locations. The operation can be performed only when the computer is in the manual mode.

Depression of the insert key sets zeros into the IR-1 and OR-2 MARS registers and activates the typewriter keyboard. The OR-2 register is incremented plus one for each entry of an instruction digit into memory to provide the address for storing the following digit. The IR-1 register retains the 00000 address throughout the insert operation.

The last instruction in the sequence should be either a branch back or branch (unconditional) instruction, depending upon whether or not the save key was used prior to entering the insert operation.

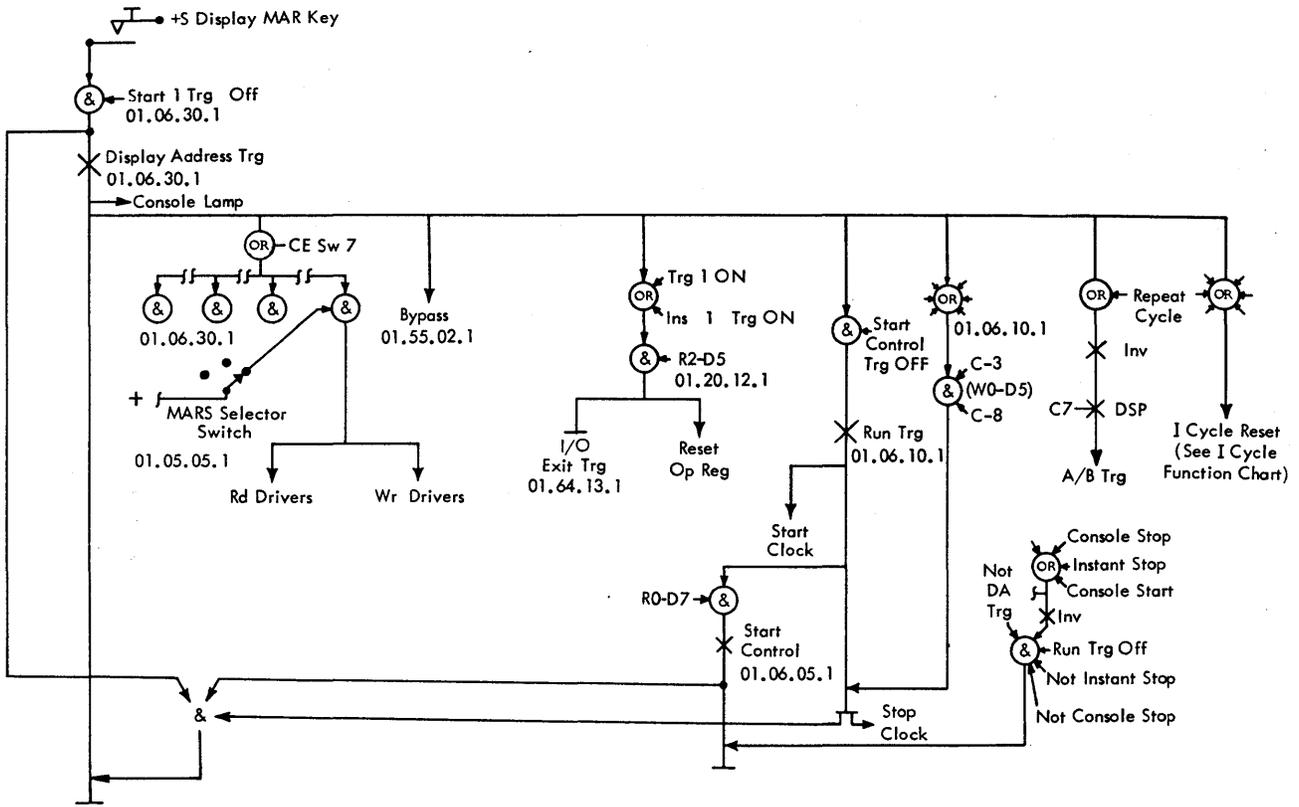


Figure 6-2. Display MAR

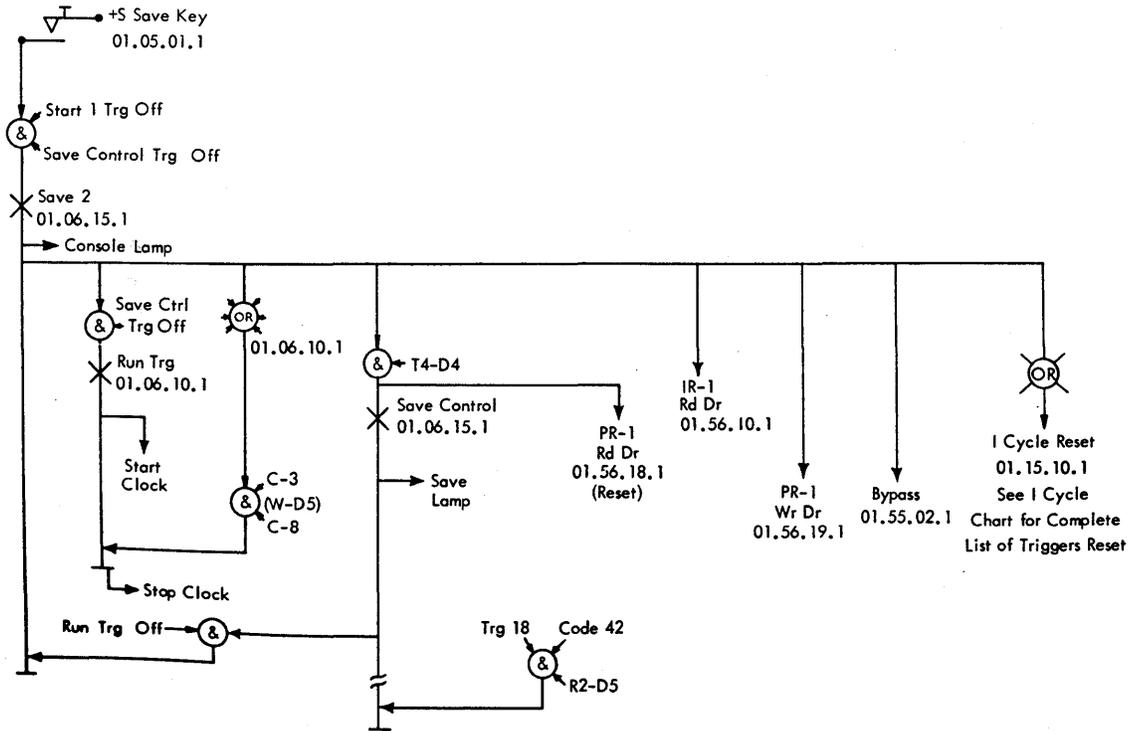


Figure 6-3. Save

The typewriter keyboard remains activated until the release key is depressed or the 100th digit is entered into memory. If the release key is not used to terminate the insert operation before the 100th digit is entered, the simulated read numerically operation is terminated, the I/O Exit and Stop 1 triggers are turned on, and the computer stops in the manual mode. Depression of the start key is required to restart the computer which will then execute the instruction starting in location 00000 (IR-1).

Figure 6-4 shows a function chart for the insert operations.

RELEASE (01.05.01.1)

The primary function of the release key is to terminate an input operation where the input device is the typewriter keyboard. If the input operation was initiated by the insert key, the insert light is turned OFF by the release key depression.

The release key can be used to immediately stop the computer during any input-output operation.

Figure 6-5 shows a function chart for release operation.

START (01.05.01.1)

Depression of the start key causes the computer to begin execution of a program. The address of the first instruction to be executed is contained in the IR-1 MARS register.

The start key is operative only when the computer is in the manual mode. Depression of the start key places the computer in the automatic mode; turns on the automatic light; and turns OFF the manual light.

Figure 6-6 shows a combined function chart for start key, stop/single instruction execute key, and instant stop/single cycle execute key operations.

STOP/SINGLE INSTRUCTION EXECUTE (SIE) (01.05.01.1)

Depression of the STOP/SIE key with the computer in the automatic mode, terminates the automatic mode and places the computer in the manual mode. The automatic light is turned OFF and the manual light is turned ON.

Successive depressions of the STOP/SIE key with the computer in the manual mode cause one instruction to be executed for each depression.

Figure 6-6 shows a combined function chart including the stop/single instruction execute key.

INSTANT STOP/SINGLE CYCLE EXECUTE (SCE) (01.05.01.1)

Depression of the INSTANT STOP/SCE key with the computer in the automatic mode, stops the computer. The cycle in which the computer stops depends upon the operation being performed. The function of the IN-

STANT STOP/SCE key differs between I/O operation codes and other operation codes.

For I/O operation codes:

1. Depression of this key during a read operation stops the computer at the end of trigger 30 time. Successive depressions cause a complete character input cycle, which is completed when the end of trigger 30 time is again reached.
2. Depression of this key during a write operation stops the computer with the Sync trigger on. Successive depressions cause a complete character output cycle which is completed when the Sync trigger is turned on again.

For other operation codes:

1. Depression of this key stops the computer at the end of the memory cycle in which the key is depressed. Successive depressions cause the computer to execute a single 20 μ sec memory cycle for each depression.

For the 1622 Card Read unit to single cycle 1622 I/O Operations, CE Switch No. 9 (MAR Stop Bypass) must be used in conjunction with the INSTANT STOP/SCE key. Depression of the INSTANT STOP/SCE key with CE Switch No. 9 ON will stop the computer at the end of trigger 30 time for both read and write operations. Successive depressions of the key cause a complete character input or output cycle which is completed when the end of trigger 30 time is again reached.

The manual light and the automatic light are both ON, except when the computer is stopped in an I-cycle entry status. In an I-cycle entry status only the manual light is ON.

Figure 6-6 shows a combined function chart including the instant stop/single cycle execute key.

R-S KEY

The R-S key (Release-Start) is installed on the typewriter, but is described here because it performs the functions of two console keys. Depression of this key during an Insert, Code 36 or Code 37, operation performs the same function as is performed by depressing the release key and then the start key.

Signal Lights

(01.05.10.1)

Signal lights on the console panel provide the machine operator with a visual indication of computer operating and error conditions.

POWER ON LIGHT

The power on light is turned ON when power is applied to the machine through the power on switch. It is turned OFF only when power is turned OFF.

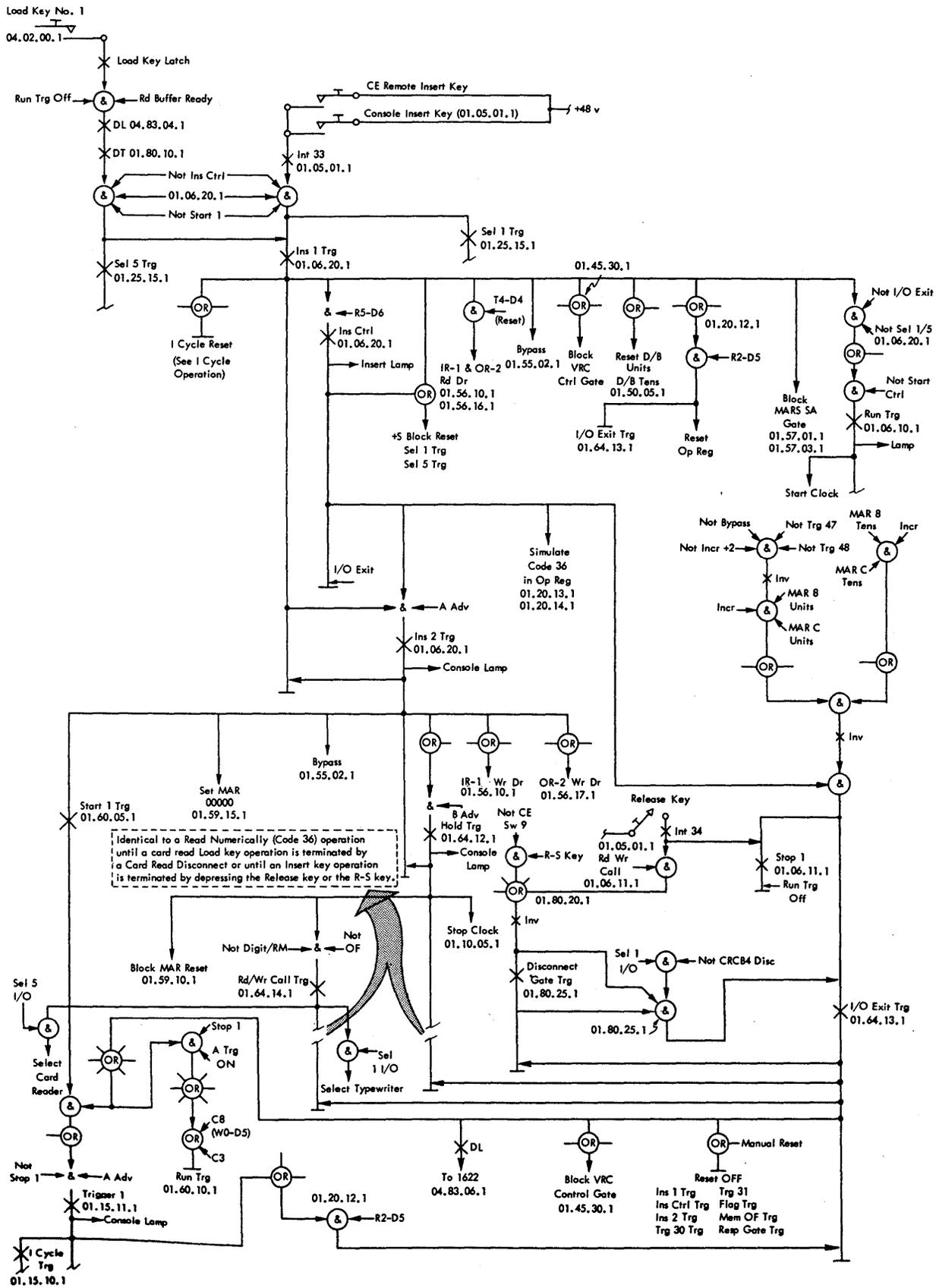


Figure 6-4. Insert

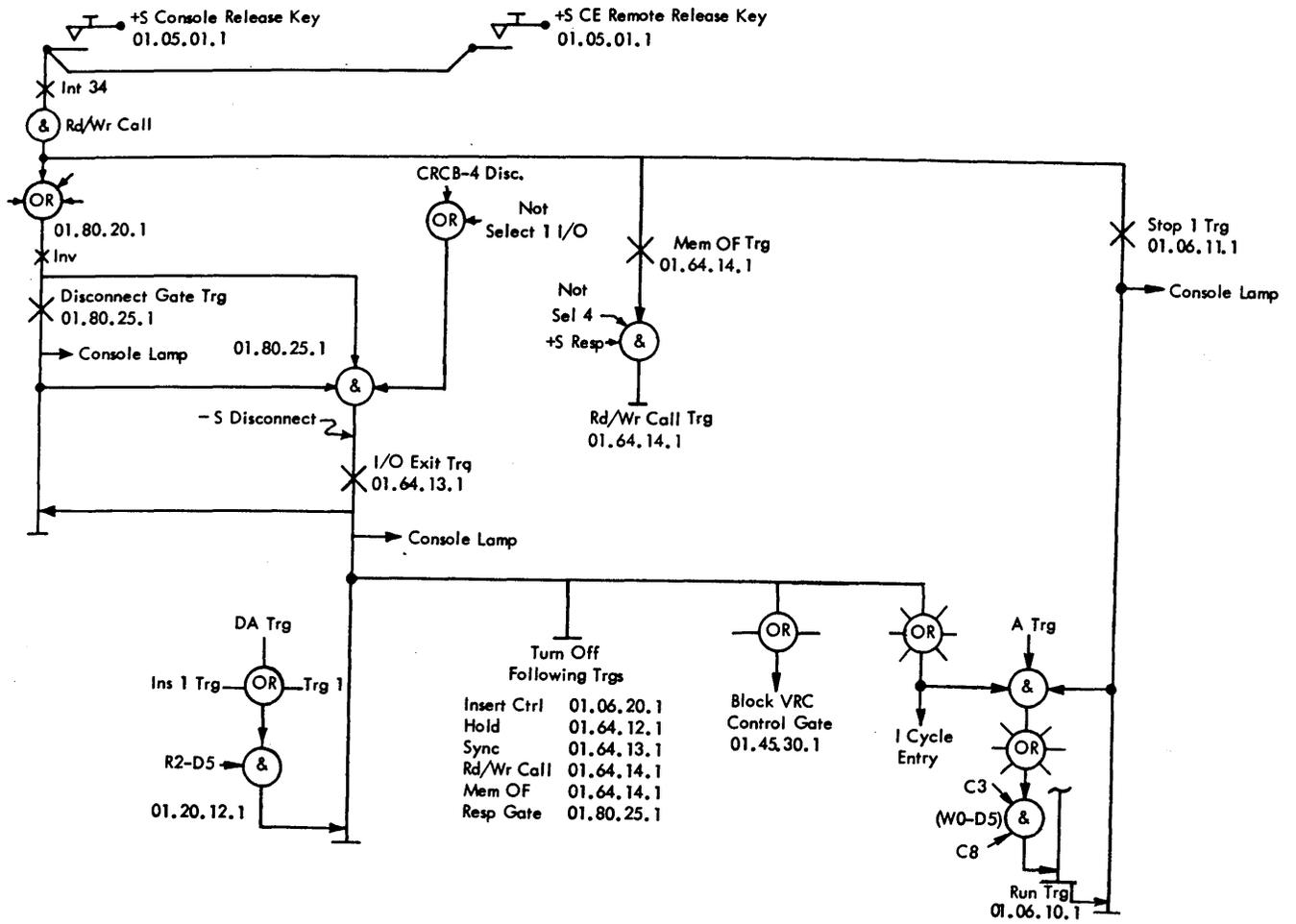


Figure 6-5. Release

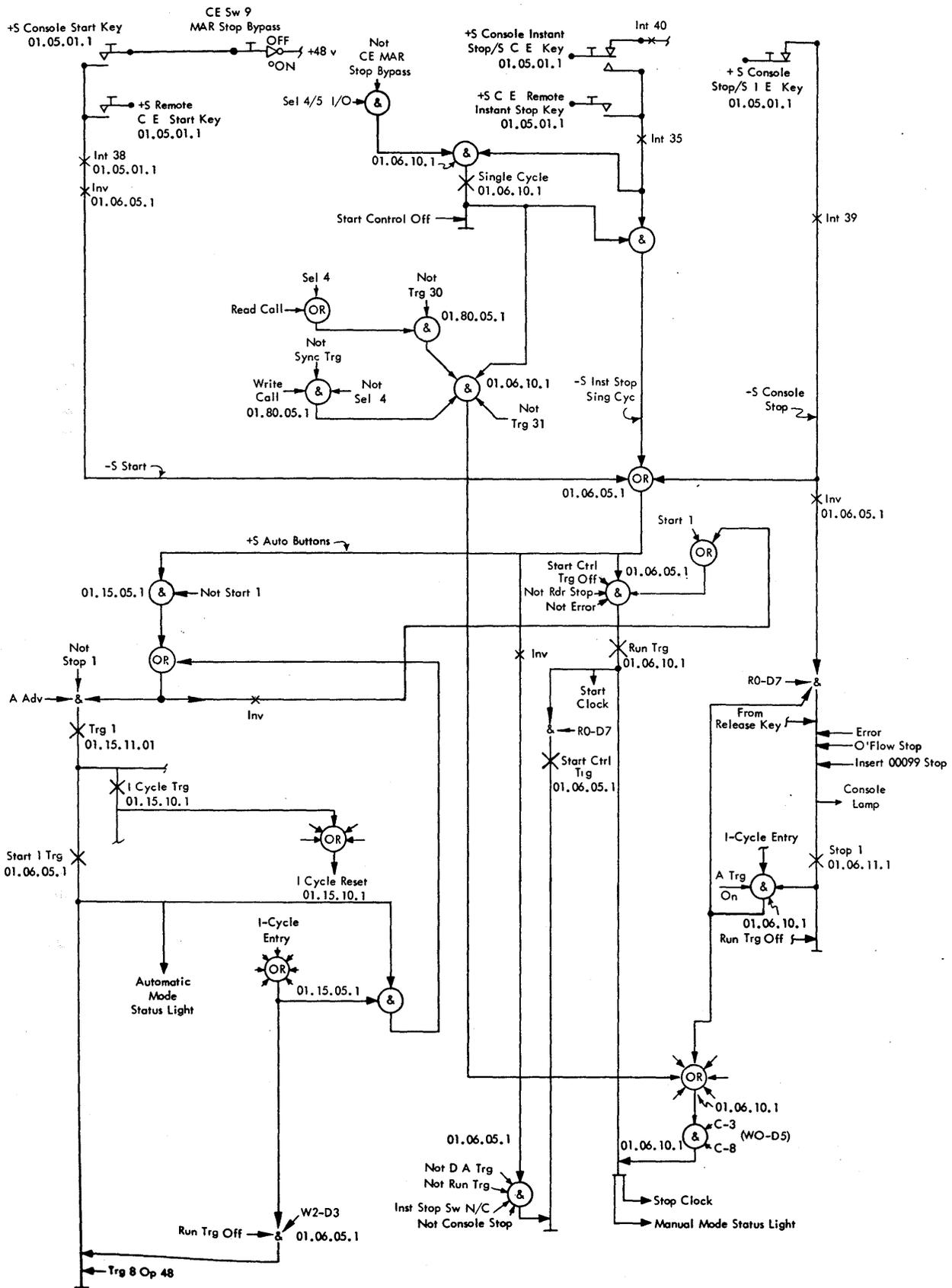


Figure 6-6. Start-Stop

POWER READY LIGHT

The power ready light is turned ON when all power supplies have been properly sequenced. It is turned OFF in the event of any power supply failure, any over temperature condition within the machine, or when the power switch is turned OFF.

MANUAL LIGHT

The manual light is ON when the computer has terminated all operation and is prepared to accept operator intervention. It is turned ON as a result of executing a halt instruction; depressing the release key, stop key, or instant stop key; or detecting an error condition (if the associated check switch is set to STOP).

The computer is normally in the manual mode when the manual light is ON. However, during single cycle execution of an instruction, both the manual and the automatic lights are ON, and the computer is in the automatic mode. Both lights are also ON, and the computer is in the automatic mode, if the instant stop key is depressed during the execution of an instruction at normal speed. In both of these instances, the computer enters the manual mode and the automatic light is turned OFF if the computer is stopped at the end of the last memory cycle required for the execution of an instruction.

AUTOMATIC LIGHT

The automatic light is ON when the computer is executing a stored program instruction. It is turned ON by depression of the start key, the insert key, the single instruction execute key, or the single cycle execute key. The machine is stopped and the automatic light is turned OFF upon entry into the instruction cycle for the next instruction in sequence, if a halt instruction has been executed; the release key or stop key has been depressed; an overflow condition occurs (with the overflow check switch set to STOP); or a parity error occurs on a read or write operation (with the data check switch set to STOP).

SAVE LIGHT

The save light is turned ON when the save key is depressed. It is turned OFF by the execution of the next branch back instruction.

INSERT LIGHT

The insert light is turned ON when the insert key or 1622 load key is depressed. It is turned OFF when the insert or load operation is terminated or when the reset key is depressed.

CHECK STOP LIGHT

The check stop light is ON if the machine stops as a result of detecting a parity error in the system data flow or an invalid address in MAR. (A CE switch is

provided to bypass the MAR error stop for testing purposes.)

THERMAL LIGHT

The thermal light is turned ON and power is removed from the machine if an over temperature condition is detected within the machine. When the thermal light goes OFF, power can again be applied.

READER NO FEED LIGHT

The reader no feed light is turned ON if the computer attempts to execute a read instruction which specifies the paper tape reader or card reader as the input device and the reader is not in a ready condition. The computer tests for the ready condition of the selected output device. The paper tape ready condition exists when the paper tape main line switch is on and the paper tape is properly loaded. The card reader ready condition exists when the card reader power switch is on, cards have been run in, and the buffer is ready.

PUNCH NO FEED LIGHT

The punch no feed light is turned ON if the computer attempts to execute a write instruction which specifies the paper tape punch or card punch as the output device and the selected device is not properly loaded. The light is also turned ON if a parity check error occurs while punching paper tape.

When one of the above conditions occurs, the computer hangs up in the automatic mode and the automatic light remains ON.

Customer Engineering Panel

The CE panel (Figure 6-7) contains switches and circuits for checking the operation of memory and associated circuits, operating the clock without a program, and repeating a specific cycle. The panel light, when on, indicates the clock is running.

CE Switch Functions

CE SWITCH 7 (INCREMENT MAR +1)

Transferring this switch (01.05.50.1) suppresses instruction cycles and allows MAR to increment by one when the clock is running. This switch may be used to start the computer cycling, beginning with any address stored in the MARS register selected by the MAR display selector switch. Depression of the insert key with the MAR INC switch ON will set the selected MARS register to zero.

CE SWITCH 8 (BYPASS MAR INCREMENT)

This switch, CE SW 8 on 01.05.50.1, when transferred, causes MAR to remain at the address which is stored in the MARS register selected by the display MARS switch.

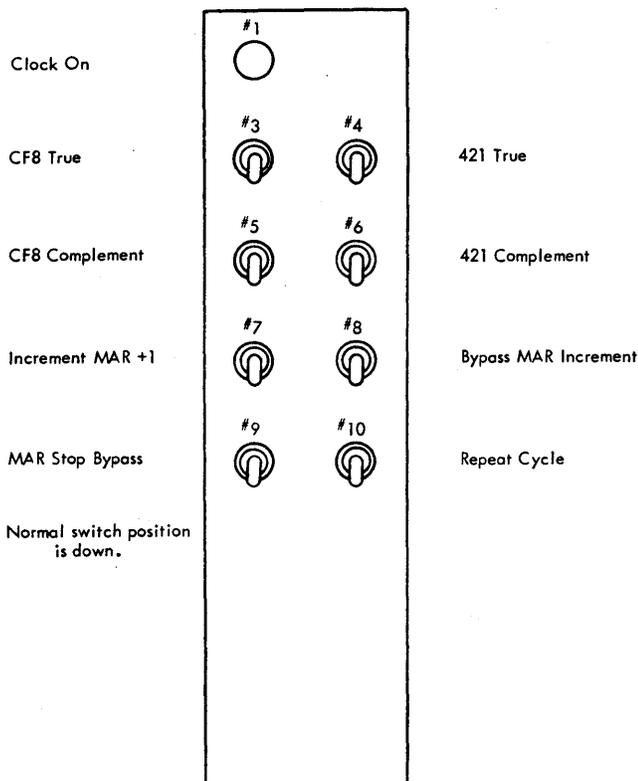


Figure 6-7. Customer Engineering Panel

With CE switch 8 and CE switch 7 transferred and the clock running, the 1620 will continually read and write at the same memory address, the address in the selected register.

TRUE/COMPLEMENT SWITCHES

The true/complement switches (CE switches 3, 4, 5, and 6) are seen on 01.05.50.1. These four switches are used in various combinations. They allow a CE to perform the operations described below:

True Operation. All four true/complement switches are set normal (down). At read time, data in memory is placed in MBR. At write time, the data in MBR is written into memory. The data in MBR controls the Z or inhibit drivers.

Complement Operation. All four true/complement switches are placed in the transferred (up) position. At read time the data in memory is placed in MBR. At write time the data in MBR is passed through the inhibit control logic. The bit complement of the data is then written in memory. For example, the C, F, and 1 bits are read from memory at read time. The 8, 4, and 2 bits are written back into memory.

Force Operation. The true/complement switches are set to control what is written into memory. The

T/C switches may be set to cause data to be stored in every bit position of a memory address and this is called "forcing all bits." If they are set to cause no data to be stored in each of the six positions, the operation is called "blanking memory." If they are set to cause data to be stored in the C, F, and 8 bit positions, this is called "forcing flag eights." If they are set to cause data to be stored in the 4, 2, and 1 bit positions, this is called "forcing sevens."

When forcing, data is read from memory into MBR at read time. At write time, MBR has no control over what is to be written back into memory. The true/complement switches control writing. The table below shows the proper switch settings for forcing operations.

CE Switches	#3	#4	#5	#6
Operation	CF8 True	421 True	CF8 Complement	421 Complement
Force all bits	Normal	Normal	Transferred	Transferred
Blank Memory	Transferred	Transferred	Normal	Normal
Force "flag 8's"	Normal	Transferred	Transferred	Normal
Force "7's"	Transferred	Normal	Normal	Transferred

WARNING: Set CE Switches 3, 4, 5, and 6 to true (down) before returning computer to customer.

CE SWITCH 9 (MAR STOP BYPASS)

Transferring this switch allows the clock to run during a MAR check. Transferring it also cripples the console start switch. With this switch transferred, the clock may be started (1) by the CE remote control box or (2) by depressing the console SIE or SCE key. (Transferring CE Switch 7 allows the console SIE key to substitute for the console start key.) It must be ON when a single cycle to or from the 1622 buffer is being performed.

This switch should be used to diagnose trouble in the clock, MAR, MARS, and increment/decrement circuitry. It is not recommended for use in analyzing memory troubles, because a MAR error adversely affects memory operation.

CE SWITCH 10 (REPEAT CYCLE)

When this switch is transferred, it prevents the A/B trigger from changing. It should be used when it is desired to repeat a particular cycle during machine analysis. The switch must be transferred in the cycle preceding the one to be repeated. For example, to repeat cycle E14:

1. Single cycle to the E13 cycle known to precede it.
2. Transfer (up) CE Switch 10.
3. Start clock.

NOTE: During this operation, addresses will change unless bypass MAR increment, CE Switch 8, is transferred. Also, certain logic conditions may have to be grounded to give a complete repeat cycle.

In the performance of stored program instructions, the computer proceeds through an instruction cycle (I-cycle), and generally an execution cycle (E-cycle) for each operation (see Figure 7-1). The function of the I-cycle is to read the 12-digit instruction from memory and interpret it by storing and decoding the operation code; by placing the P field and Q field addresses in MARS storage registers; and by storing and decoding specific digits of the Q part of the instruction for branch and input-output operations, where it is required. Indirect Addressing, an additional feature, modifies the standard instruction cycle. A description of indirect addressing is given in the *Customer Engineering Manual of Instruction, 1620 Data Processing System, Additional Features*, (Form 227-5513). A sequence block diagram for the I-cycle is shown in Figure 7-2. The I-cycle function charts are shown on pages 7, 8, and 9 in the *IBM Customer Engineering Instructional System Diagrams, 1620 Data Processing System* (Form 227-5631).

Objectives

1. Read the 12 digits of the instruction from the memory location specified by IR-1 and successively higher memory locations.
2. Store the digits in registers as follows:

Digits	Register
O ₀ and O ₁	Operation (OP-REG)
P ₂ , P ₃ , P ₄ , P ₅ , and P ₆ (P Field)	Operand (OR-2 and OR-3)
Q ₇ , Q ₈ , Q ₉ , Q ₁₀ and Q ₁₁ (Q Field)	*Operand (OR-1)
Q ₈ and Q ₉	*Also Product (PR-1) for DIVIDE
Q ₁₁	MDR **Digit/Branch (D/B)

*Q-part of an instruction is not entered into the OR-1 register (or the PR-1 register) if the operation is immediate. (See Objective No. 7.)

**The Q₈ and Q₉ digits are retained in the digit/branch register if the operation to be performed is control, read, write, dump, branch indicator, or branch no indicator.
3. Decode the OP REG contents to determine the operation to be performed.

4. Decode the D/B REG contents to determine the specific input-output device if the operation is control, read, write, or dump.
5. Decode the D/B REG contents to determine the indicator or sense switch to be interrogated if the operation is branch indicator or branch no indicator.
6. Decode the Q₁₁ digit in MDR to determine the control function to be performed if the operation is control.
7. Store the memory address of the Q₁₁ digit of the instruction in the OR-1 MARS operand register (and the PR-1 MARS product register for divide) if the operation is immediate.

Functions

The I-cycle for each instruction consists of eight 20- μ sec memory (machine) cycles during which the 12-digit instruction is read from memory and stored in registers as indicated under Objective No. 2. The contents of the operation register are decoded to determine the desired operation. The contents of the digit/branch register and MDR are decoded where applicable for the purposes stated in Objectives 4, 5, and 6. The P-address of an instruction is always entered into the OR-2 and OR-3 MARS registers by triggers 2, 3, and 4 of the I-cycle. The Q-address is entered into the OR-1 MARS register (and the PR-1 MARS register for divide) by triggers 5, 6, and 7 of the I-cycle, except in the case of immediate operations. Certain of the non-immediate operations do not use the stored P or Q address because the objectives of the operation do not require it.

Immediate operations require the use of digits of the instruction as data. The five Q part digits of the instruction are not placed in OR-1 (or PR-1) by triggers 5, 6, and 7 of the I-cycle. During trigger 8 time, the normal memory cycle reset of MAR is blocked and the memory address which was placed in MAR during trigger 7 time, is transferred to OR-1 (and PR-1 for divide). This address is the location in memory for the Q₁₁ digit of the instruction. Therefore, the address in OR-1 (and PR-1 for divide) at the end of the I-cycle designates the location in memory of a field whose units position is the Q₁₁ digit of the instruction itself. The field can contain as few as two digits

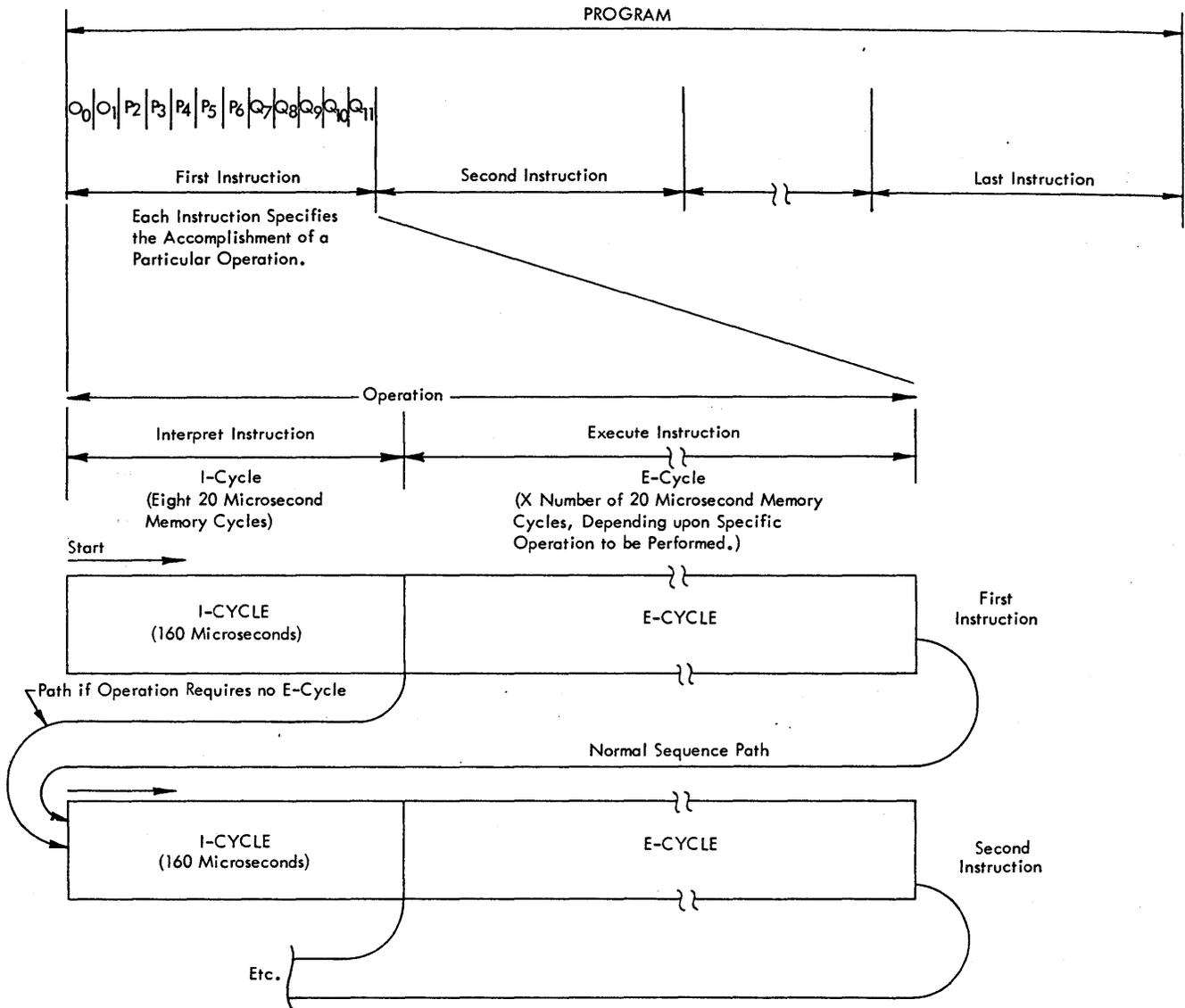


Figure 7-1. Program I Cycle—E Cycle

(Q_{10} and Q_{11}) or more than five, depending upon the location of the flag bit marking its high-order position.

Figure 7-3 illustrates the cycle-by-cycle storage of instruction digits in the operation register and MARS for the non-immediate instruction, 25 01357 02468. This instruction is assumed to be in memory at locations 00000 through 00011, and address 00000 is assumed to be in IR-1 at the beginning of the I-cycle.

Figure 7-4 illustrates the cycle-by-cycle storage for the immediate instruction, 15 01357 02468, under the same conditions as in Figure 7-3.

Auxiliary Triggers

STATUS

I-cycle Trigger. (01.15.10.1).

1. Turned ON by trigger 1 and remains ON through trigger 7 time.
2. Turned OFF by trigger 8.

Decrement Trigger. (01.60.05.1).

1. Turned OFF (increment) by the I-cycle trigger and remains OFF for the entire I-cycle.

E Cycle Entry Trigger.

1. Turned on by trigger 8.

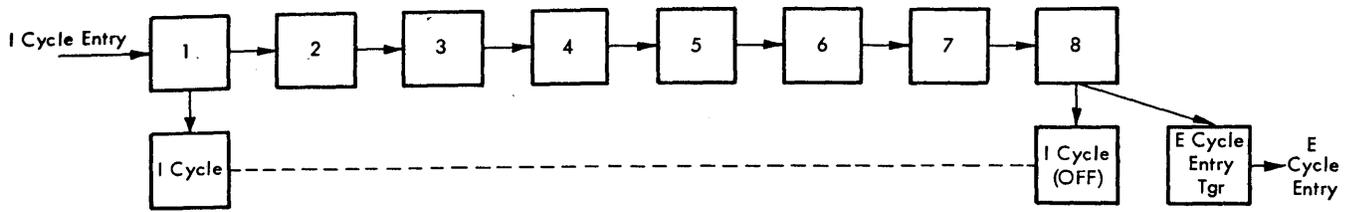


Figure 7-2. Instruction Cycle

Memory Cycle	Triggers	IR-1 at Beginning of Memory Cycle	MAR	Increment	IR-1 at End of Memory Cycle	OP Reg	OR-2	OR-1
1	I Cycle & 1	00000	00000	+2	00002	25		
2	I Cycle & 2	00002	00002	+2	00004	25	01	
3	I Cycle & 3	00004	00004	+2	00006	25	01 35	
4	I Cycle & 4	00006	00006	+1	00007	25	01 35 7	
5	I Cycle & 5	00007	00007	+2	00009	25	01 35 7	0
6	I Cycle & 6	00009	00009	+2	00011	25	01 35 7	0 24
7	I Cycle & 7	00011	00011	+1	00012	25	01 35 7	0 24 68 *
8	8	-----	00011	--	00012	25	01 35 7	0 24 68

* If the operation to be performed is control, read, write, dump, branch indicator, or branch no indicator the Digit/Branch register is not reset during Trigger 7 time. However, the contents of the Digit/Branch register is entered into OR-1 tens. With the example used in this figure, OR-1 will contain 02424 at the end of Trigger 7 time.

Figure 7-3. Instruction Cycle – Non Immediate

Memory Cycle	I Cycle Triggers	IR-1 at Beginning of Memory Cycle	MAR	Incr	IR-1 at End of Memory Cycle	OP Reg	OR-2	OR-1
1	I Cycle & 1	00000	00000	+2	00002	15		
2	I Cycle & 2	00002	00002	+2	00004	15	01	
3	I Cycle & 3	00004	00004	+2	00006	15	01 35	
4	I Cycle & 4	00006	00006	+1	00007	15	01 35 7	
5	I Cycle & 5	00007	00007	+2	00009	15	01 35 7	-----
6	I Cycle & 6	00009	00009	+2	00011	15	01 35 7	-----
7	I Cycle & 7	00011	00011	+1	00012	15	01 35 7	-----
8	8	-----	00011	--	00012	15	01 35 7	*00011

* Trigger 8. OR-1 Write Drivers (From MAR through Incr/Decr bypassed)

Figure 7-4. Instruction Cycle—Immediate

2. Turned OFF at T5 time of the first memory cycle after trigger 8 time except when a write operation with the card punch is to follow.
3. Turned OFF during the first trigger 30 time of a write operation with the card punch selected (Sel 4) as the output device.

OBJECTIVES

I-Cycle Trigger.

1. Resets OFF the following triggers for use during the execution cycle.
 - a. Digit/Record mark
 - b. Recomplement
 - c. Recomplement control
 - d. Field mark No. 1
 - e. Field mark No. 2
 - f. Dividend and remainder sign
 - g. Divide add
 - h. First divide cycle
 - i. Last divide cycle
 - j. Trigger No. 19
 - k. Trigger No. 29
 - l. Trigger No. 43
 - m. Trigger No. 45
 - n. Decrement
 - o. Trigger No. 12
 - p. Trigger No. 13
 - q. Trigger No. 14
 - r. Trigger No. 27
 - s. M/Q register
 - t. First cycle
 - u. Carry in
 - v. Carry out
2. Reset the units position of the digit/branch register for each of the memory cycles controlled by triggers 1 through 7. (See item 5.)
3. Read out of memory per IR-1 and store the digit in the units position of the digit/branch register for each of the memory cycles controlled by triggers 1 through 7. (See item 5.)
4. Write back into IR-1 from MAR incremented for each of the memory cycles controlled by triggers 1 through 7. (Increment plus one or plus 2 is controlled by the individual triggers 1 through 7.)
5. Block reset of D/B register units at trigger 7 time of OP codes 34, 35, 36, 37, 38, 39, 46, and 47.

E-Cycle Entry Trigger. (01.15.18.1).

1. Turn off trigger 8 on the next A Advance.
2. Turn off the τ/c trigger to change the initial status set by trigger 1 if the operation to be performed is subtract or compare.
3. Turn ON first divide cycle trigger if this instruction is Divide – Code 29/19.

4. Enter I-cycle for the next instruction in sequence if this instruction is No Operation – Code 41.
5. Enter E-cycle (turn on trigger 18) if this instruction is Branch Indicator – Code 46 and the branch test trigger is on.
6. Enter I-cycle for the next instruction in sequence if this instruction is Branch Indicator – Code 46 and the branch test trigger is off.
7. Enter E-cycle (turn on trigger 18) if this instruction is Branch No Indicator – Code 47 and the branch test trigger is off.
8. Enter I-cycle for the next instruction in sequence if this instruction is Branch No Indicator – Code 47 and the branch test trigger is on.

I-Timer Triggers

OBJECTIVES OF I-TIMER TRIGGERS

Trigger 1. (01.15.11.1).

1. Turn on the I-cycle trigger.
2. Read out of memory per IR-1 (function of I-cycle trigger) and store the even digit (O_0) in OP REG TENS (via MDR). Store the odd digit (O_1) in OP REG UNITS (via MBR-odd).
3. Write back IR-1 (function of I-cycle trigger) incremented plus 2.
4. Reset OR-1, OR-2, and OR-3 to permit setting of addresses later in the I-cycle.
5. Turn the true/complement (τ/c) trigger ON to set initial status for add, subtract, and compare operations.
6. Turn I/O exit trigger OFF.
7. Turn start 1 trigger ON.

Trigger 2. (01.15.12.1).

1. Read out of memory per IR-1 (function of I-cycle trigger) and write the even digit (P_2) into OR-2 and OR-3 ten thousands position (via MDR and D/B REG UNITS). Write the odd digit (P_3) into OR-2 and OR-3 thousands positions (via MBR-odd and D/B REG TENS).
2. Write back IR-1 (function of I-cycle trigger) incremented plus 2.
3. Reset branch test trigger OFF to permit setting the trigger during trigger 7 time of the I-cycle for branch operations.

Trigger 3. (01.15.13.1).

1. Read out of memory per IR-1 (function of I-cycle trigger) and write the even digit (P_4) into OR-2 and OR-3 hundreds position (via MDR and D/B REG UNITS). Write the odd digit (P_5) into OR-2 and OR-3 tens position (via MBR-odd and D/B REG TENS).

2. Write back IR-1 (function of I-cycle trigger) incremented plus 2.

Trigger 4. (01.15.14.1).

1. Read out of memory per IR-1 (function of I-cycle trigger) and write the even digit (P_6) into OR-2 and OR-3 units position (via MDR and D/B REG UNITS).
2. Write back IR-1 incremented plus 1 (function of I-cycle trigger).
3. Read out PR-1 to clear it for receiving (1) the Q field (divisor) address later in the I-cycle of divide operation, (2) the product area address on first E-cycle of load dividend operation, and (3) the product area address on first E-cycle of multiply operations.

Trigger 5. (01.15.15.1).

1. Read out of memory per IR-1 (function of I-cycle trigger) and write the odd digit (Q_7) into OR-1 (and PR-1 for divide) ten thousands position (via MDR and D/B REG UNITS). An immediate instruction prevents this digit from being written into OR-1 and PR-1. (See trigger 8, Objective 3 following.)
2. Write back IR-1 (function of I-cycle trigger) incremented plus 2.

Trigger 6. (01.15.16.1).

1. Read out of memory per IR-1 (function of I-cycle trigger) and write the even digit (Q_8) into OR-1 (and PR-1 for divide) thousands position (via MBR-even and D/B REG TENS). Write the odd digit (Q_9) into OR-1 (and PR-1 for divide) hundreds position (via MDR and D/B REG UNITS). An immediate instruction prevents these digits from being written into OR-1 and PR-1. (See trigger 8, Objective 3 following.)

2. Write back IR-1 (function of I-cycle trigger) incremented plus 2.

Trigger 7. (01.15.17.1).

1. For codes other than 34, 35, 36, 37, 38, 39, 46, and 47, read out of memory per IR-1 (function of I-cycle trigger) and write the even digit (Q_{10}) into OR-1 (and PR-1 for DIVIDE) tens position (via MBR-even and D/B REG TENS). Write the odd digit (Q_{11}) into OR-1 (and PR-1 for divide) units position (via MDR and D/B REG UNITS). An immediate instruction prevents these digits from being written into OR-1 and PR-1. (See trigger 8 Objective 3 following).
2. Write back IR-1 incremented plus 1 (function of I-cycle trigger).
3. For Codes 34, 35, 36, 37, 38, 39, 46, and 47, block the reset of D/B register tens. The reset of D/B units and tens is blocked for Codes 34 through 39, thus preserving the I/O device code for later use.
4. For Codes 46 and 47, turn on the branch test trigger if the indicator or sense switch designated by the D/B REG is ON.

Trigger 8. (01.15.18.1).

1. Turn I-cycle trigger OFF.
2. Block reset of MAR to retain the address of the Q_{11} digit of the instruction for use by immediate instructions.
3. For immediate instructions, write the address of the Q_{11} digit of the instruction into OR-1 (and PR-1 for divide) from MAR.
4. Stop the computer if this instruction is Halt — Code 48.
5. Turn E-cycle entry trigger ON.
6. Turn on the Decrement trigger if the operation code to be performed requires a decrement function.

Section 8 Execution Cycle

The performance of a computer operation is normally divided into two parts, an instruction cycle (I-cycle) and an execution cycle (E-cycle). (See Figure 7-1).

Each E-cycle is immediately preceded by its associated I-cycle during which the instruction is read from memory and interpreted. The I-cycle is described in Section 7.

The function of the E-cycle is to accomplish the objectives of the particular operation specified by the instruction.

Objective

Execute the instruction read from memory and interpreted during the associated I-cycle.

Functions

An E-cycle consists of an even number of 20 μ sec memory cycles under the control of E-timer triggers. The required number of memory cycles depends upon the specific operation to be performed and also upon the length of fields and records in arithmetic, transmit, and input-output operations.

The following instructions are complete at the end of their I-cycle and require no E-cycle for their execution:

1. Branch indicator, if no branch occurs
2. Branch no indicator, if no branch occurs
3. Halt
4. No Operation

When the E-cycle for an operation is complete, the computer is directed to enter the I-cycle for the next instruction in sequence.

E-Timer Triggers

The trigger directory in the Appendix lists the E-timer triggers and indicates their location in the *System Diagrams*.

In general, each E-timer trigger controls all of the functions which must be accomplished during the memory cycle for which it is ON. At the beginning of the following memory cycle it serves to turn ON the E-timer trigger which is to assume control and is turned OFF by the new E-timer trigger (see Figure 8-1).

Because identical or similar functions are accom-

plished during the execution of two or more different computer operations, many of the E-timer triggers are used with more than one operation. For example, the Branch (Code 49) operation is executed in two memory cycles, using E-timer triggers 18 and 19. The Branch On Digit (Code 43) operation is executed in four memory cycles, using E-timer triggers 28, 29, 18, and 19. Triggers 18 and 19 are used with both operations to accomplish identical functions. The E-timer trigger chart in the Appendix shows the E-timer triggers that are used with each computer operation.

In arithmetic, transmit, and input-output operations where fields or records are processed, a series of E-timer triggers is used to process the first digit (or character) of the field or record and then looped as many times as required to process each subsequent digit (or character) (see Figure 8-2).

The functions of individual E-timer triggers are described in Sections 9, 10, 11, 12, and 14 of this manual as they are used to accomplish the objectives of the various computer operations.

Auxiliary Triggers

Auxiliary triggers are used to establish or recognize conditions within the computer which must be considered to properly accomplish the execution of an instruction.

The auxiliary trigger chart in the Appendix shows the auxiliary triggers that are used with each computer operation. Auxiliary trigger locations on the *System Diagrams* are indicated in the trigger directory in the Appendix, under "Named" Triggers.

Functions of Auxiliary Triggers

Functions of the most commonly used auxiliary triggers are described below. The status of all auxiliary triggers are described in Sections 9, 10, 11, 12, and 14 of this manual as they are used in the various computer operations.

E Cycle Entry. (01.15.18.1). The output of the E-cycle entry trigger is used to develop several E-cycle timed gates. These gates are used to control turning on of various E-timer triggers. They are developed as shown on the function chart, Figure 8-3. Note that the E-cycle advance signal and the trigger 14 advance signal are developed only at A advance time.

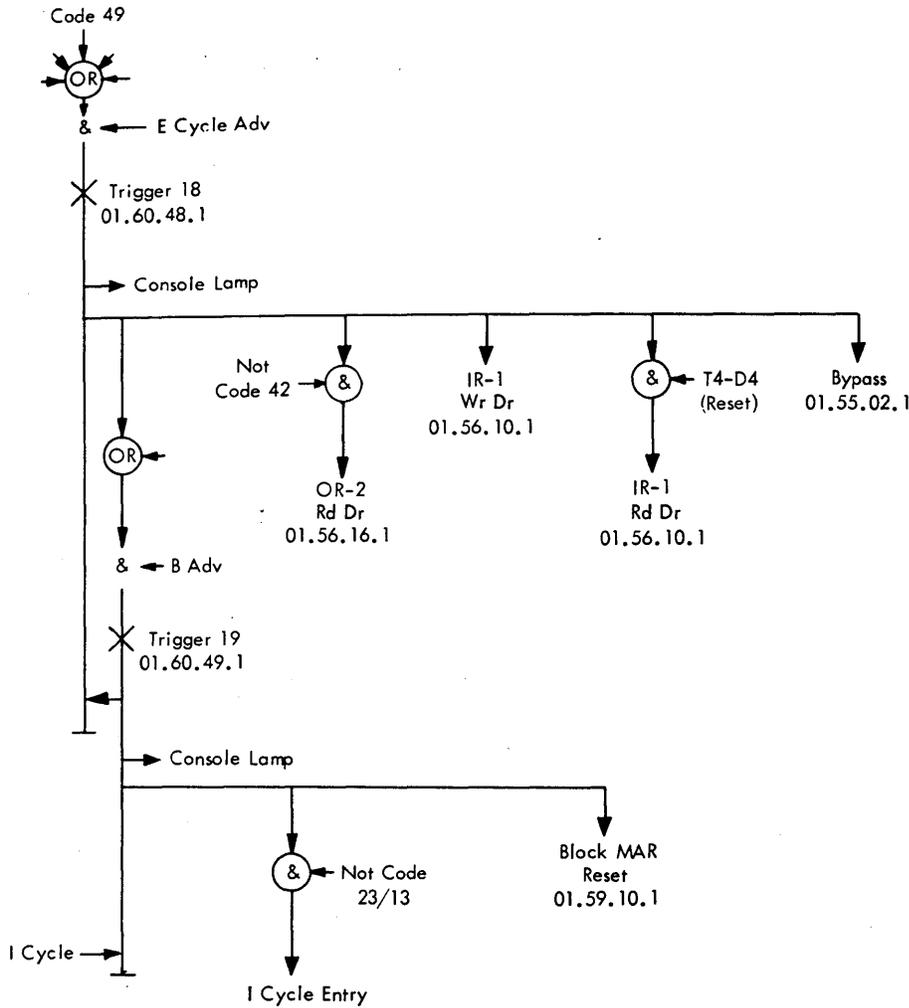


Figure 8-1. E Timer Functions

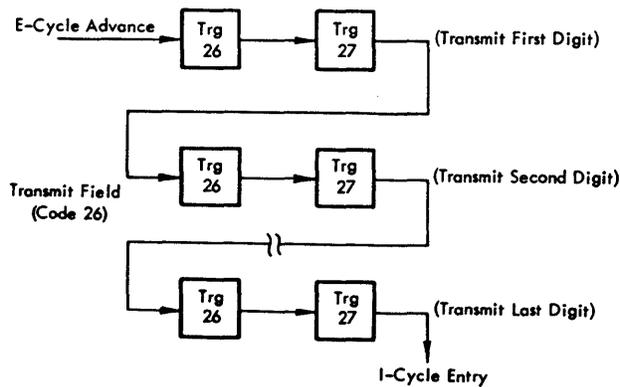


Figure 8-2. Transmit Field

Decrement (Decr). (01.60.05.1). The decrement trigger is turned ON (decrement) or OFF (increment) as necessitated by the requirements of each computer operation. All instruction cycles require increment operation. The increment/decrement requirements of the execution cycle for each operation are indicated on the auxiliary trigger chart in the Appendix. The function of the decrement trigger is to determine by means of its ON or OFF status whether an address in MAR is to be decremented or incremented before it is written into MARS. This function is accomplished by control of the increment/decrement switch.

First Cycle. (01.63.10.1). The primary function of the first cycle trigger is to permit a flag bit in the units position of a field to be interpreted as denoting the sign of the field rather than the high-order posi-

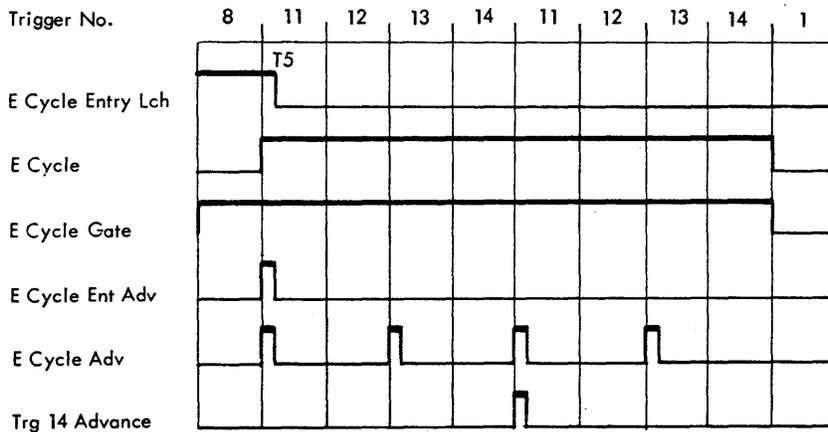
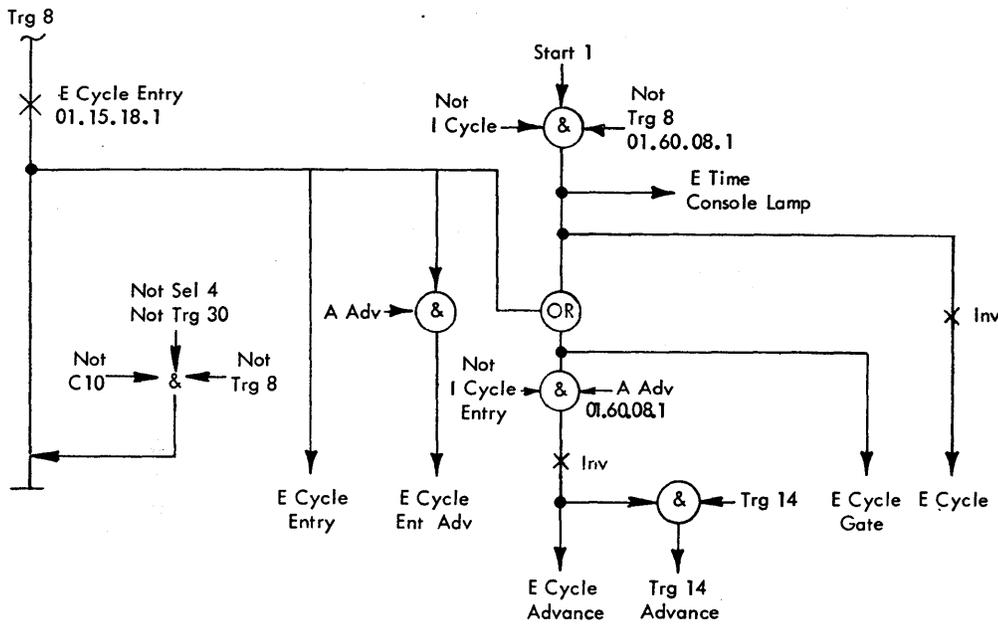


Figure 8-3. E-Cycle Gates

tion of the field. The first cycle trigger is ON only during the memory cycle (or cycles) in which the units position of fields is being processed in arithmetic and transmit field operations.

Field Mark No. 1. (01.63.30.1). Field mark No. 1 trigger is turned ON during arithmetic and transmit field operations when the high-order position of the Q field is read from memory to indicate that the entire Q field has been processed. In branch no flag operation, field mark No. 1 trigger is turned ON to recognize the presence of a flag bit in the memory location which is interrogated.

Field Mark No. 2. (01.63.30.1). Field mark No. 2 trigger is turned ON during add, subtract, compare, and multiply operations when the high-order position of the P field is read from memory to indicate that the entire P field has been processed. Field mark No. 2 trigger is turned ON during divide operations to indicate the end of a divide subtract cycle.

True/Complement (T/C). (01.63.20.1). The T/C trigger is manipulated during add, subtract, and compare operations to establish an initial condition and alter that condition according to the signs of the factors involved in the particular add, sub-

tract, or compare operation. In divide operations the τ/c trigger is turned OFF (complement) to perform subtractions of the divisor from the dividend, and ON (true) to add the divisor back into the dividend when an overdraw occurs. The function of the τ/c trigger is to determine by means of its ON (true) or OFF (complement) status whether digits presented to the τ/c switch appear at its output unchanged or complemented.

High/Plus. (01.60.40.1). The high/plus trigger is manipulated during arithmetic operations to cause the trigger to indicate the sign of the result by means of its status at the end of the particular arithmetic operation. ON indicates plus; OFF indicates minus. In compare operations, the status of the high/plus trigger indicates which of two fields is algebraically higher. A zero result of arithmetic or compare operations will turn the high/plus trigger OFF.

Equal/Zero. (01.60.41.1). The equal/zero trigger is turned ON at the beginning of arithmetic operations and turned OFF during the particular arithmetic operation if the result is not zero. In compare operations, if two fields are equal, the equal/zero trigger remains ON to indicate an equal condition. If a comparison is made of two fields having unlike signs and containing a significant digit, the equal/zero trigger is turned OFF.

Carry Out. (01.63.40.1). The carry out trigger is turned ON during arithmetic operations if a digit with a flag bit is read into MDR from the add table in memory or if a carry occurs into a position which contains a "9".

Carry In. (01.63.40.1). The carry in trigger is turned ON during arithmetic operations if the addition of the two previous digits resulted in a carry out or if the add table address for the units position of a sum is being developed on a complement operation.

Recomplement Control. (01.60.24.1). The recomplement control trigger is turned ON near the end of an add or subtract operation if the developed sum or difference is in a complement form. The primary function of the recomplement control trigger is to initiate recomplement of the complement sum or difference to the true form.

Recomplement. (01.60.32.1). The recomplement trigger is turned ON at the beginning of recomplement. Its functions are to continue recomplement until all digits of the complement sum or difference have been recomplemented and then to direct the com-

puter to enter the I-cycle for the next instruction in sequence.

Overflow. (01.60.40.1). The overflow trigger is turned ON in add, subtract, and compare operations if the length of the P field is inadequate. In divide operation, the overflow trigger is turned ON if "divide by zero" is attempted or if the first quotient digit developed is greater than 9.

Digit/Record Mark. (01.63.50.1). The digit/record mark (D/RM) trigger is used during add, subtract, and compare operations to detect a digit in MDR. The D/RM trigger is turned on to terminate a transmit record operation or a write operation when the record mark character is read from memory into MDR. The D/RM trigger is turned on if a record mark is decoded during a branch no record mark operation or if a digit is decoded during a branch on digit operation.

Branch Test. (01.25.35.1). The branch test trigger is turned ON during trigger 7 time of the I-cycle for branch indicator and branch no indicator instructions if the program switch or indicator interrogated by the instruction is ON. The status of the branch test trigger is then used to determine whether the computer proceeds to the next instruction in sequence or branches to another instruction.

Read Y Point

During the execution of certain computer operations, it is required that in one memory cycle a particular memory location be cleared, and to write into it, a digit which was stored in MDR during the previous memory cycle. A "Read Y Point" function is available to satisfy that requirement by accomplishing the following:

1. The reset of MDR is blocked to preserve the new digit.
2. Either the even or the odd sense amplifiers are blocked when the memory location designated by the address in MAR is read out. (The memory location is thus cleared and its content is prevented from reaching MBR.)
3. The digit in MDR is transferred to MBR-even or MBR-odd, depending upon whether the address in MAR is even or odd. The new digit is then written into the memory location designated by the address in MAR when the normal "write-back" into memory function is performed. The function chart shown in Figure 8-4 includes the Read Y Point function.

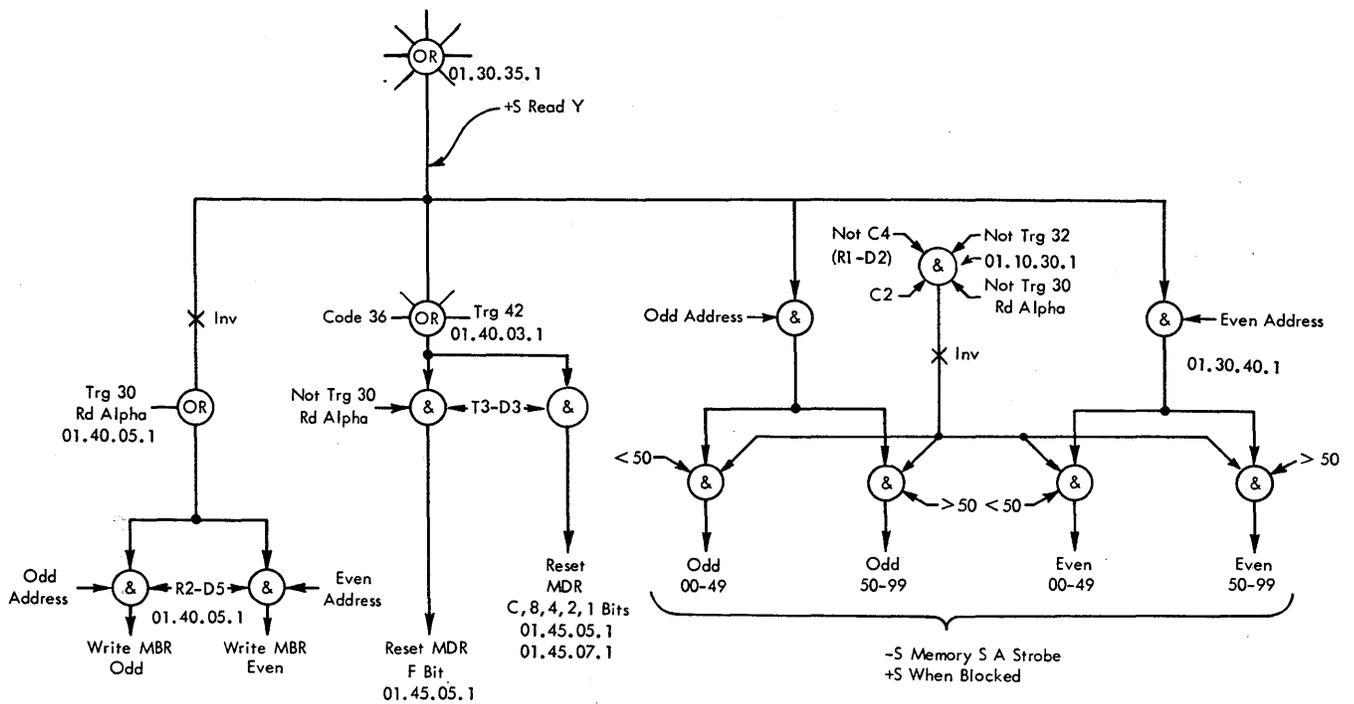


Figure 8-4. Read Y and Block Memory Sense Amplifiers

Block Memory Sense Amplifiers

Where it is required, during computer operations, that new information be placed into two adjacent memory locations in the same memory cycle, a "Block Mem SA" function is available to block both the even and odd memory sense amplifiers. Clearing the product area to zeros during multiply and load dividend operations and storing alphameric information during read alphameric operations requires the use of the Block

Mem SA function.

The function chart shown in Figure 8-4 includes the Block Mem SA function.

Significance of P & Q Parts of Instructions

Figure 8-5 indicates the significance of the P and Q parts of instructions for each of the operations that the computer is capable of performing.

Operation Code	P Address	Q Address
11—Add (I)*	Memory address of units position of Augend.	Q ₁₁ of instruction is units position of Addend.
21—Add	Same as Code 11.	Memory address of units position of Addend.
12—Subtract (I)*	Memory address of units position of Minuend.	Q ₁₁ of instruction is units position of Subtrahend.
22—Subtract	Same as Code 12.	Memory address of units position of Subtrahend.
13—Multiply (I)*	Memory address of units position of Multiplicand.	Q ₁₁ of instruction is units position of Multiplier.
23—Multiply	Same as Code 13.	Memory address of units position of Multiplier.
14—Compare (I)*	Memory address of units position of the field to which another field is to be compared.	Q ₁₁ of instruction is units position of the field to be compared with the field at the P address.
24—Compare	Same as Code 14.	Memory address of units position of the field to be compared with the field at the P address.
15—Transmit Digit (I)*	Memory address to which single digit is to be transmitted.	Q ₁₁ of instruction is the single digit to be transmitted.
25—Transmit Digit	Same as Code 15.	Memory address of single digit to be transmitted.
16—Transmit Field (I)*	Memory address to which units position of field is to be transmitted.	Q ₁₁ of instruction is the units position of the field to be transmitted.
26—Transmit Field	Same as Code 16.	Memory address of units position of the field to be transmitted.
17—Branch and Transmit (I)*	"P minus one" is the memory address to which the units position of the Q field is to be transmitted. "P" is the memory address of the high-order digit of the next instruction to be interpreted and executed.	Q ₁₁ of instruction is the units position of the field to be transmitted.
27—Branch and Transmit	Same as Code 17.	Memory address of units position of the field to be transmitted.
18—Load Dividend (I)*	Memory address in Product Area to which units position of field (Dividend) is to be transmitted.	Q ₁₁ of instruction is the units position of the field (Dividend) to be transmitted.
28—Load Dividend	Same as Code 18.	Memory address of the units position of the field (Dividend) to be transmitted.
19—Divide (I)*	Memory address at which first subtraction of the Divisor is to occur.	Q ₁₁ of instruction is the units position of the Divisor.
29—Divide	Same as Code 19.	Memory address of units position of Divisor.
31—Transmit Record	Memory address to which high-order position of record is to be transmitted.	Memory address of high-order position of the record to be transmitted.
32—Set Flag	Memory address at which flag bit is to be placed.	Not used.
33—Clear Flag	Memory address from which flag bit is to be cleared.	Not used.
34—Control	Not used.	Q ₄ and Q ₅ specify input-output device. Q ₁₁ specifies control function to be performed.
35—Dump Numerically	Memory address from which first numerical character is to be written.	Q ₄ and Q ₅ specify output device.
36—Read Numerically	Memory address at which first numerical character is to be stored.	Q ₄ and Q ₅ specify input device.
37—Read Alphamerically	Memory address at which numerical digit of first character is to be stored. (Zone digit of first character will be stored at "P minus one.")	Same as Code 36.
38—Write Numerically	Memory address from which first numerical character is to be written.	Same as Code 35.
39—Write Alphamerically	Memory address for numerical digit of first character to be written. (Zone digit of first character is at "P minus one.")	Same as Code 35.
41—No Op	Not used.	Not used.
42—Branch Back	Not used.	Not used.
43—Branch On Digit	Memory address of the high-order digit of the next instruction to be interpreted and executed, if Branch occurs. (Not used if Branch does not occur.)	Memory address to be interrogated for the presence of a significant digit (not a zero).
44—Branch No Flag	Same as Code 43.	Memory address to be interrogated for the presence of a flag bit.
45—Branch No Record Mark	Same as Code 43.	Memory address to be interrogated for the presence of a Record Mark character.
46—Branch Indicator	Same as Code 43.	Q ₄ and Q ₅ digits specify program switch or indicator to be interrogated for status.
47—Branch No Indicator	Same as Code 43.	Same as Code 46.
48—Halt	Not used.	Not used.
49—Branch	Memory address of the high-order digit of the next instruction to be interpreted and executed.	Not used.
** 71—Move Flag	Memory address of flag to be moved.	Memory address to which flag is to be moved.
** 72—Transfer Numerical Strip	Memory address of the units position of the alphameric field.	Memory address of the units position of the numerical field.
** 73—Transfer Numerical Fill	Same as Code 72.	Same as Code 72.

* Immediate.

** Additional Feature

Figure 8-5. Significance of P & Q Parts of Instruction

Section 9 Internal Transmission Operations

Internal transmission operations accomplish the transfer of a digit, a field, or a record from one memory location to another. Two memory cycles are required for processing each digit. During the first memory cycle, the digit to be transmitted is read out of memory and stored in MDR. During the second memory cycle, the digit in MDR is transferred to the desired memory location.

Function charts for 1620 operation codes are located in the *IBM Customer Engineering Instructional System Diagrams, 1620 Data Processing System (Form 227-5631)*.

Transmit Digit (Code 25 — TD)

The sequence block diagram for this operation is shown in Figure 9-1. The function chart for this operation is shown in the *Instructional System Diagrams*, page 10.

Objective

Replace the single digit at the P address (OR-2) with the digit and its F bit, if any, at the Q address (OR-1).

Functions

The digit at the P address (OR-2) prior to the transmission is obliterated, including its F bit, if any. The digit at the Q address (OR-1) remains unchanged. The operation is terminated when the single digit has been transmitted.

Auxiliary Triggers

None are used.

E-Timer Trigger Objectives

Trigger 26. (01.60.57.1).

1. Read out of memory per OR-1 and store the digit in MDR.

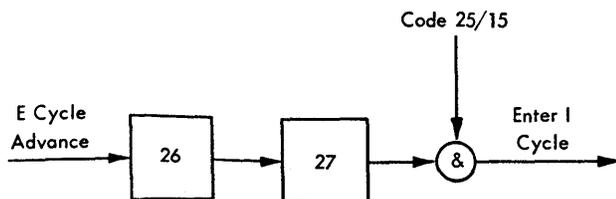


Figure 9-1. Transmit Digit—Code 25/15

Trigger 27. (01.60.57.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Transfer MDR to MBR.
4. Write into memory per OR-2 from MBR.
5. End operation and enter I-cycle for the next instruction in sequence.

Transmit Digit Immediate (Code 15 — TDM)

The sequence block diagram for this operation is shown in Figure 9-1. The function chart for this operation is shown in the *Instructional System Diagrams*, page 10.

Objective

Replace the single digit at the P address (OR-2) with the digit and its F bit, if any, in the units position (Q₁₁) of the transmit digit immediate instruction.

Functions

Prior to transmission, the digit at the P address (OR-2) and its F bit, if any, is obliterated. The digit at the Q address (OR-1) remains unchanged. The operation is terminated when the single digit has been transmitted.

Auxiliary Triggers

None are used.

E-Timer Triggers

See Transmit Digit — Code 25

Transmit Field (Code 26 — TF)

The sequence block diagram for this operation is shown in Figure 9-2. The function chart for this operation is shown in the *Instructional System Diagrams*, page 10.

Objective

Transfer the field, including F bits for sign and field definition, at the Q address (OR-1) to the memory

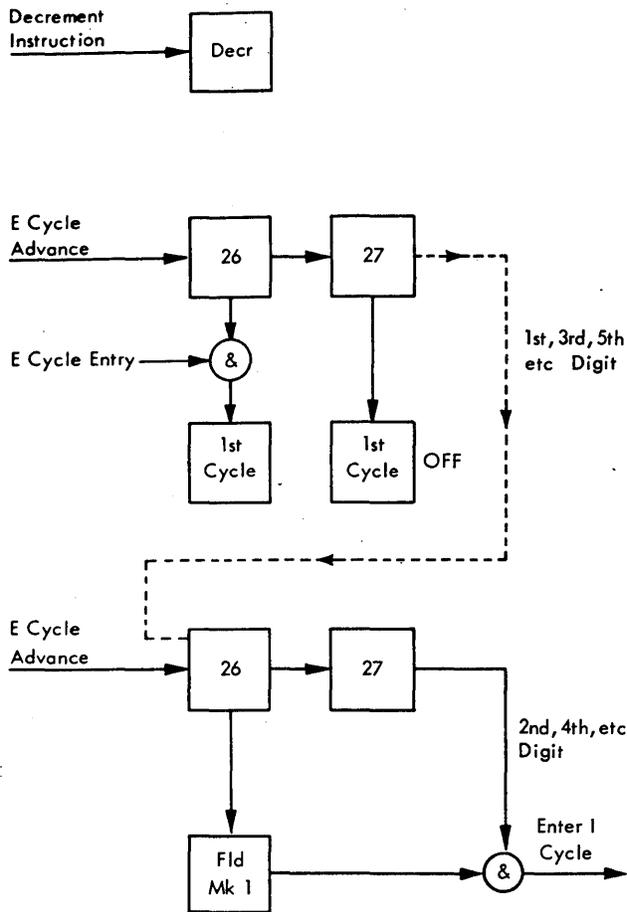


Figure 9-2. Transmit Field—Code 26/16

location designated by the P address (OR-2) and successively lower memory locations.

Functions

Transmission proceeds serially, one digit at a time, from low-order to high-order digit of the transmitted field until the operation is terminated by a flag bit (field mark No. 1) in the high-order position of the transmitted field. The flag bits in the high-order and units position of the transmitted field are duplicated in the field at P. The digits in the field at P prior to transmission are obliterated, including their flag bits, if any.

Auxiliary Trigger Status

First Cycle Trigger (01.63.10.1)

1. Turned ON by trigger 26 ANDed with E-cycle entry.
2. Turned OFF by trigger 27 on first transmit cycle.

Decrement Trigger. (01.60.05.1)

1. Turned ON (decrement) during trigger 8 time of the preceding I-cycle.
2. Remains ON until the next I-cycle is entered.

Field Mark No. 1 Trigger. (01.63.30.1)

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first transmit cycle (Q field must be a minimum of two digits).
3. Turned ON during trigger 26 time of the transmit cycle in which the high-order digit of the Q field is read out of memory.

E-Timer Trigger Objectives

Trigger 26. (01.60.57.1)

1. Read out of memory per OR-1 and store the digit in MDR.
2. Write back OR-1 decremented.

Trigger 27. (01.60.57.1)

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Write back OR-2 decremented.
4. Transfer MDR to MBR.
5. Write into memory per OR-2 from MBR. (Repeat trigger 86 and trigger 27, Objectives 1, 2, 3, 4, and 5 for each digit of the field).
6. End operation when all digits of the field have been transferred and enter I-cycle for the next instruction in sequence.

Transmit Field Immediate (Code 16 — TFM)

The sequence block diagram for this operation is shown in Figure 9-2. The function chart for this operation is shown in the *Instructional System Diagrams*, page 10.

Objective

Transfer the field, including F bits for sign and field definition, whose units position is the Q_{11} digit of the transmit field immediate instruction to the memory location designated by the P address (OR-2) and successively lower memory locations.

Functions

See Transmit Field — Code 26.

Auxiliary Triggers Status

See Transmit Field — Code 26.

E-Timer Triggers

See Transmit Field — Code 26.

Transmit Record (Code 31 — TR)

The sequence block diagram for this operation is shown in Figure 9-3. The function chart for this operation is shown in the *Instructional System Diagrams*, page 10.

Objective

Transfer the record, including F bits for sign and field definition, at the Q address (OR-1) to the memory location designated by the P address (OR-2) and successively higher memory locations.

Functions

Transmission proceeds serially, one digit at a time, from high-order to low-order digit of the transmitted record until the operation is terminated by a record mark in the low-order position of the transmitted record. All flag bits in the record at Q are duplicated in the record at P. The digits in the record at P prior to the transmission are obliterated, including their flag bits, if any. The record mark is duplicated in the low-order positions of the record at P.

Auxiliary Trigger Status

Decrement Trigger. (01.60.05.1).

1. OFF (increment) during the I-cycle.
2. Remains OFF throughout the E-cycle.

Digit/Record Mark Trigger. (01.63.50.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 26 time of the transmit cycle in which the record mark at the Q address (OR-1) is read out of memory.

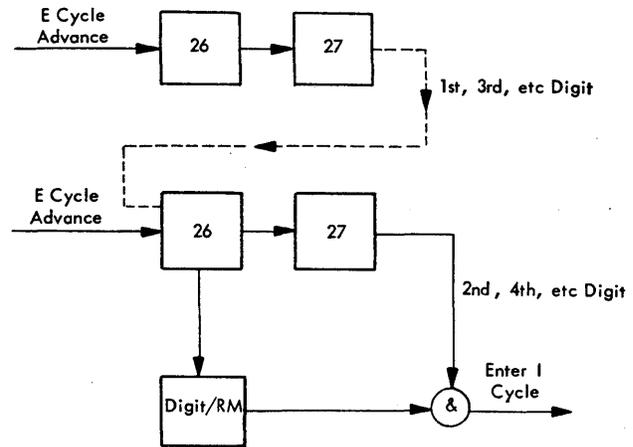


Figure 9-3. Transmit Record—Code 31

E-Timer Trigger Objectives

Trigger 26. (01.60.57.1).

1. Read out of memory per OR-1 and store the digit in MDR.
2. Write back OR-1 incremented plus 1.

Trigger 27. (01.60.57.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Write back OR-2 incremented plus 1.
4. Transfer MDR to MBR.
5. Write into memory from MBR. (Repeat trigger 26 and 27, Objectives 1, 2, 3, 4, and 5 for each digit of the record.)
6. End operation when all digits of the record and the record mark have been transferred and have entered I-cycle for the next instruction in sequence.

Section 10 Arithmetic Operations

Arithmetic operations are accomplished by referring to tables stored in memory. The Add Table, Figure 10-1, contains all possible two-digit sums, with a carry indicated by a flag bit with the table digit where the sum exceeds nine. The Multiply Table, Figure 10-2, contains all possible two-digit products. Each two-digit product read from memory consists of two adjacent digits from the Multiply Table area.

The address for a specific table location is developed by combining in a unique manner the two factors involved in a cycle of the arithmetic operation.

Subtraction is accomplished by complement addition using the Add Table.

Compare operations establish the relative value of two fields, with signs taken into consideration. Compare is essentially a subtract operation, except that the difference digits are not retained.

Division is accomplished by successive subtraction. The divide instruction is preceded by a load dividend operation.

Function charts for 1620 operation codes are located in the *IBM Customer Engineering Instructional System Diagrams, 1620 Data Processing System* (Form 227-5631).

High Order Positions of Address	Units Position of Address									
	0	1	2	3	4	5	6	7	8	9
0030	0	1	2	3	4	5	6	7	8	9
0031	1	2	3	4	5	6	7	8	9	0
0032	2	3	4	5	6	7	8	9	0	1
0033	3	4	5	6	7	8	9	0	1	2
0034	4	5	6	7	8	9	0	1	2	3
0035	5	6	7	8	9	0	1	2	3	4
0036	6	7	8	9	0	1	2	3	4	5
0037	7	8	9	0	1	2	3	4	5	6
0038	8	9	0	1	2	3	4	5	6	7
0039	9	0	1	2	3	4	5	6	7	8

Figure 10-1. Add Table

High Order Positions of Address	Units Position of Address									
	0	1	2	3	4	5	6	7	8	9
0010	0	0	0	0	0	0	0	0	0	0
0011	0	0	1	0	2	0	3	0	4	0
0012	0	0	2	0	4	0	6	0	8	0
0013	0	0	3	0	6	0	9	0	2	1
0014	0	0	4	0	8	0	2	1	6	1
0015	0	0	5	0	0	1	5	1	0	2
0016	0	0	6	0	2	1	8	1	4	2
0017	0	0	7	0	4	1	1	2	8	2
0018	0	0	8	0	6	1	4	2	2	3
0019	0	0	9	0	8	1	7	2	6	3
0020	0	0	0	0	0	0	0	0	0	0
0021	5	0	6	0	7	0	8	0	9	0
0022	0	1	2	1	4	1	6	1	8	1
0023	5	1	8	1	1	2	4	2	7	2
0024	0	2	4	2	8	2	2	3	6	3
0025	5	2	0	3	5	3	0	4	5	4
0026	0	3	6	3	2	4	8	4	4	5
0027	5	3	2	4	9	4	6	5	3	6
0028	0	4	8	4	6	5	4	6	2	7
0029	5	4	4	5	3	6	2	7	1	8

Figure 10-2. Multiply Table

Add (Code 21 — A)

Sequence block diagrams for this operation are shown in Figures 10-3 and 10-4. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

Objective

Add the field at the Q address (OR-1) to the field at the P address (OR-2), and store the algebraic sum at the P address (OR-2) and successively lower memory locations.

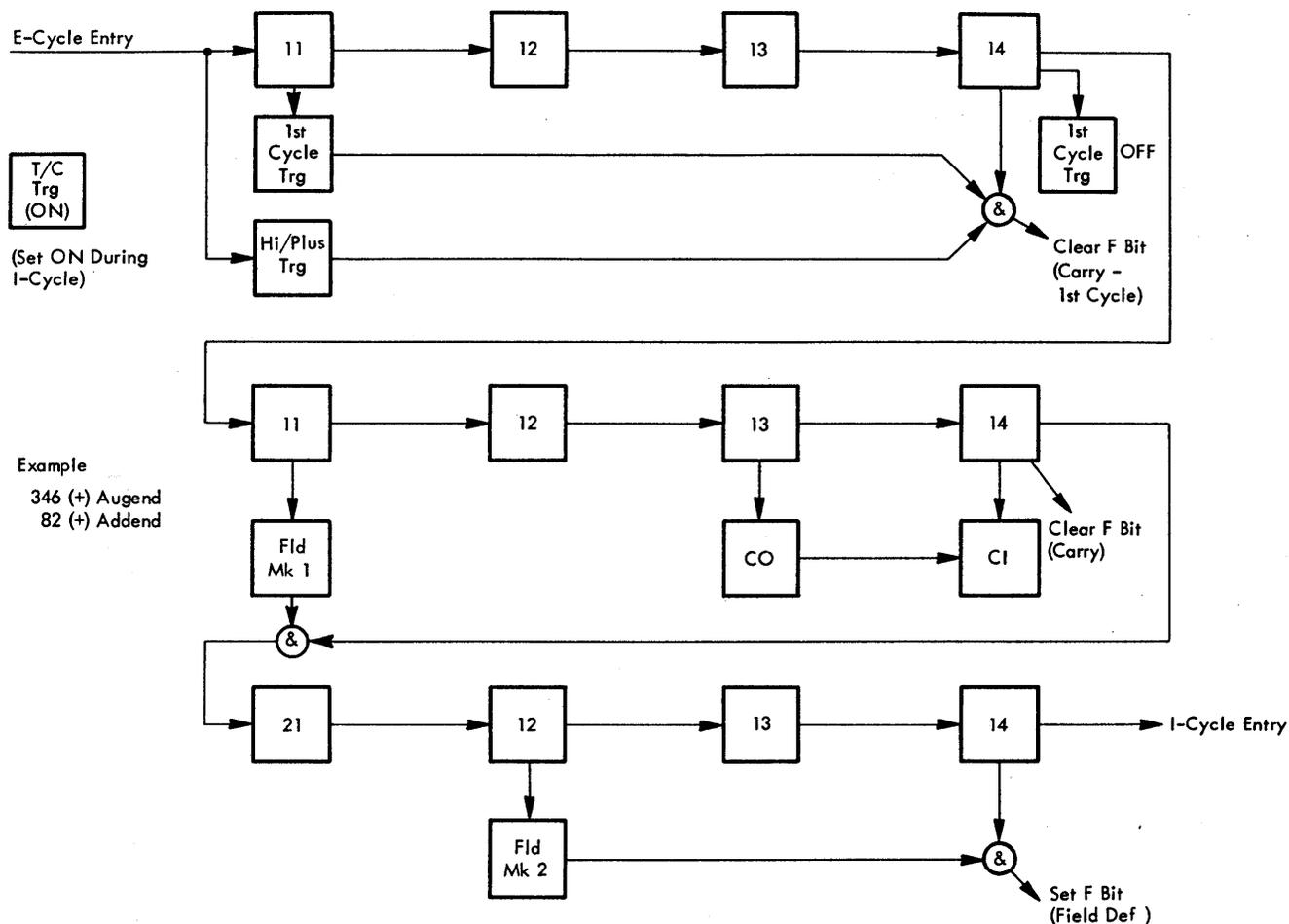


Figure 10-3. ADD—Code 21/11

Functions

The location of the addend (Q field) is specified by OR-1. The location of the augend (P field) is specified by OR-2. The algebraic sum is stored in the P fields specified by OR-2. The Q field remains unchanged. Minimum field lengths for either the P or Q fields is two digits.

Four memory cycles are required for the development of each digit in the algebraic sum, as follows:

1. One digit from the Q field is read from memory per OR-1 to the units position of the digit/branch register.
2. The corresponding digit of the P field is read from memory per OR-2 to MDR.
3. The P field digit is transferred from MDR to the tens position of MAR and the Q field digit stored in the digit/branch register is transferred to the units position of MAR. NOTE: the Q field digit may be complemented before it is placed in MAR depending upon the signs of the P and Q fields.

This manufactured address with a 3 placed into the hundreds position of MAR is used to read the one digit sum from the add table in memory, Figure 10-1, to MDR.

4. The one digit sum is then written back into memory from MDR to the correct digit position (per OR-2) replacing the original P field digit.

There are two types of carries:

1. A carry resulting from the addition of two digits is noted by a flag bit with the appropriate one digit sum in the add table. This flag bit turns on the carry out trigger.
2. A carry resulting from the addition of a previous carry to a "nine" output of the T/c switch is noted by turning on the carry out trigger.

Detection of a carry (carry out trigger on) sets a logical path for the next Q field digit so that it is increased by one before it is used to address the add table in the next add cycle.

Recomplement is required in any add operation where the sign of the P and Q fields are initially differ-

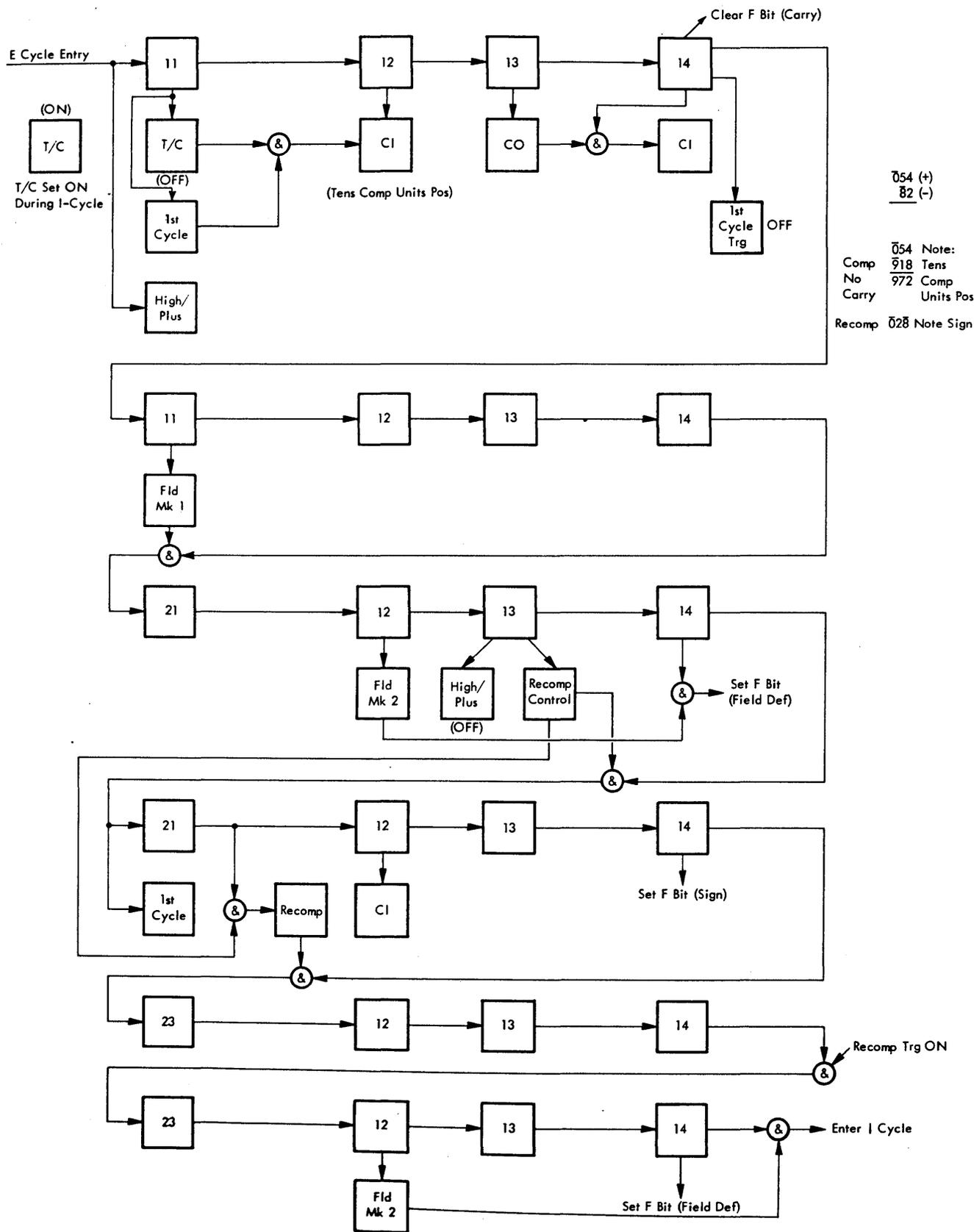


Figure 10-4. Add with Recomplement

ent and the P field is of less absolute value than the Q field.

Addition proceeds serially, one digit at a time, building up partial sums from low-order to high-order digit of the sum field until the operation is terminated by a flag bit (field mark No. 2) in the high-order position of the P field. The high-order digit of the sum is marked by storing a flag bit. The sign of the sum is marked by the presence or absence of a flag bit in the units position of the sum (a flag bit indicates a minus sign — no flag bit indicates plus sign).

The number of digits in the sum is equal to the number of digits in the P field. For a complete sum to be formed, the number of digits in the P field must be greater than or equal to the number of digits in the Q field. If this rule is violated, the overflow indicator is turned on, and the addition is performed using only as many Q field digits as there are positions in the P field (the extra digits in the addend are not used). The algebraic sum of the two equal length fields is then obtained. See chart, Figure 10-5.

Resulting Indicator Conditions

1. The high/plus indicator is ON if the sum is positive and is OFF if the sum is negative or zero.
2. The equal/zero indicator is ON if the sum is zero and is OFF if the sum is not zero.
3. The overflow indicator is turned ON if an overflow occurs; the overflow digit is lost. If the overflow indicator is ON as the result of a previous arithmetic operation, a no overflow condition of this add instruction will not turn it OFF.

NOTE: Once the high/plus or equal/zero indicators are turned ON (or OFF) by an arithmetic or compare operation, they will retain that state until the next arithmetic or compare operation is executed (unless the reset key on the console is depressed which turns off these indicators). Testing them with a branch indicator or branch no indicator instruction has no effect on their state. The overflow indicator will be turned OFF only by testing it with a branch indicator or branch no indicator instruction (or by depressing the reset key on the console).

Auxiliary Trigger Status

True/Complement (T/C) Trigger. (01.63.20.1)

1. Turned ON by trigger 1 during I-cycle and therefore will be ON when E-cycle is entered.
2. Turned OFF during trigger 11 time on first add cycle if Q field is negative.
3. Status changed during trigger 12 time on first add cycle if P field is negative.

4. Operation will be true if τ/c trigger is ON at the end of trigger 12 time of first add cycle and complement if τ/c trigger is OFF. Complement operation requires that the Q field be complemented (10's complement in units position; 9's complement in other positions) for the development of add table addresses.

Decrement trigger. (01.60.05.1)

1. Turned ON (decrement) during trigger 8 time of the preceding I-cycle.
2. Will remain ON until the next I-cycle is entered.
3. Where it is required during the E-cycle that a MARS address be written back in the same storage register or transferred to another MARS storage register without being decremented, a decrement switch bypass line will be made available by an E-timer trigger.

First Cycle Trigger. (01.63.10.1)

1. Turned ON by trigger 11 when E-cycle is entered.
2. Turned OFF at end of trigger 14 time on first add cycle.

If recomp is required:

3. Turned ON at the beginning of recomp operation in parallel with trigger 21.
4. Turned OFF by trigger 23 at end of first cycle of recomp.

High/Plus Trigger. (01.60.40.1)

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 12 time on first add cycle if P field is negative.
3. Status changed during trigger 13 time on add cycle in which field mark No. 2 is reached if re-complement is required. (No carry out on a complement operation.)
4. Turned OFF by trigger 14 when I-cycle is entered if the equal/zero trigger is on (zero sum or difference).

Equal/Zero Trigger. (01.60.41.1)

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 13 time if an add table digit is other than zero.

Field Mark No. 1 Trigger. (01.63.30.1)

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned on during first add cycle (Q field must be a minimum of two digits).
3. Will be turned ON during trigger 11 time on add cycle in which end of Q field is reached.

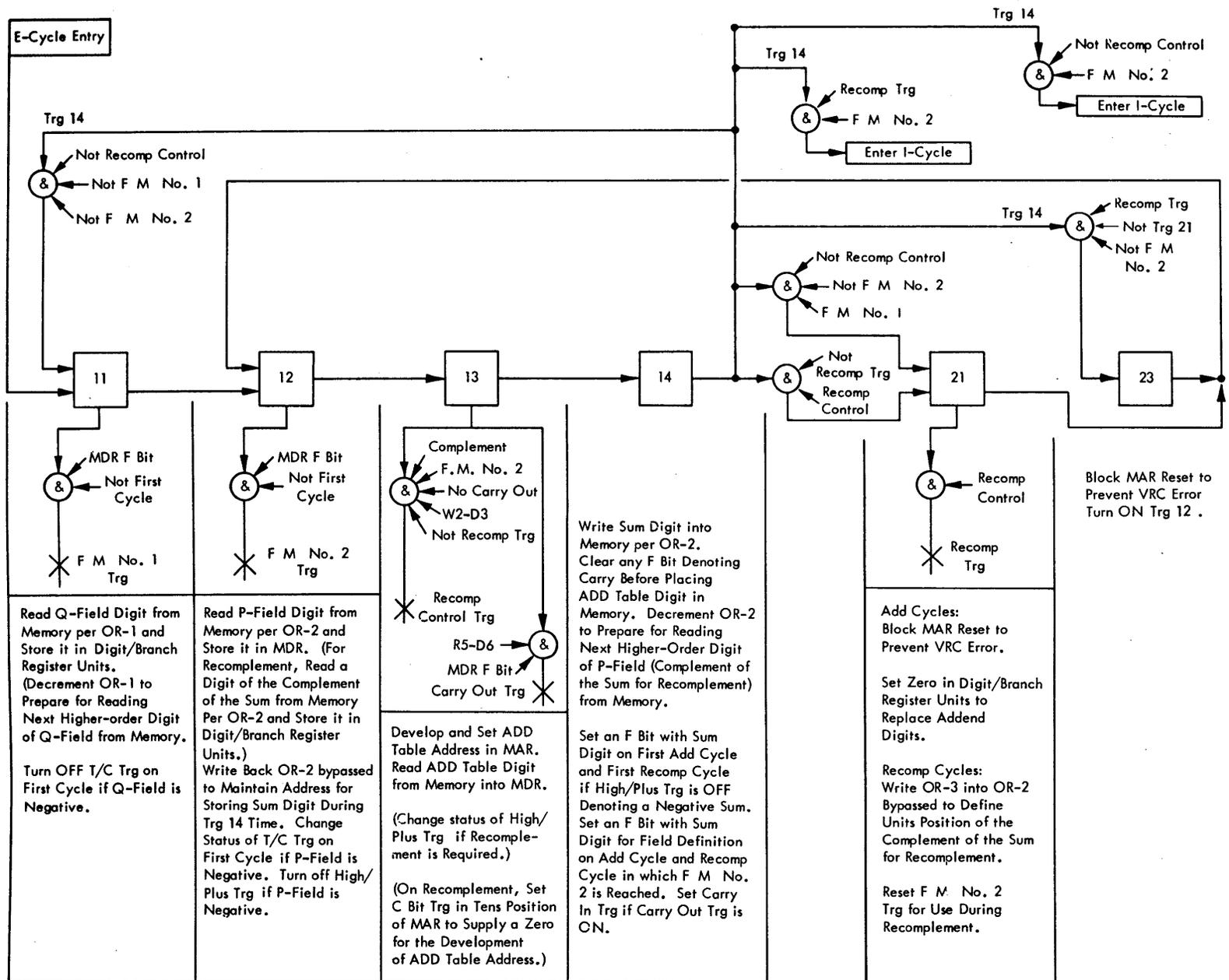


Figure 10-5. Add Flow Diagram

Field Mark No. 2 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first add cycle (P field must be a minimum of two digits).
3. Will be turned ON during trigger 12 time on add cycle in which end of P field is reached.

Carry Out Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
2. Reset OFF by triggers 11, 21, and 23.
3. Turned ON during trigger 13 time on add and subtract cycles where the output of the T/C switch is a "9" and the carry in trigger is ON.
4. Turned ON during trigger 13 time on add cycles where the add table digit in MDR contains an F bit (carry).

Carry In Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
2. Turned ON during trigger 12 time on first add cycle if operation is complement (to obtain 10's complement in units position).
3. Reset OFF during trigger 13 time.
4. Turned ON by trigger 14 if the carry out trigger is ON.
5. Turned ON during trigger 12 time on first recomp cycle (to obtain 10's complement in units position).

Recomplement Control Trigger. (01.60.24.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
2. Will be turned ON during trigger 13 time on a complement operation if there is no carry out on add cycle in which field mark No. 2 is reached.

Recomplement Trigger. (01.60.32.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON by trigger 21 at the beginning of a recomplement operation. (Recomplement trigger ANDs with trigger 14 to turn ON trigger 23 for the second and succeeding cycles of recomp.)

Sign Analysis

The sign analysis chart, Figure 10-6, shows the operation of the T/C, high/plus, and equal/zero triggers for an add operation. The chart shows the conditions requiring a recomplement operation and the conditions causing an overflow.

E-Timer Trigger Objectives

Trigger 11. (01.60.11.1).

1. Read out of memory per OR-1 and store the digit in D/B REG units.
2. Write back OR-1 decremented.
3. Turn ON field mark No. 1 trigger after first cycle when end of Q field is reached.
4. Turn OFF T/C trigger during first cycle if Q field is negative.

		ADD			
Sign of P Field (Augend)		+	+	-	-
Sign of Q Field (Addend)		+	-	+	-
Set High/Plus Trg (Store Sign of P Field)		ON +	ON +	OFF -	OFF -
Set T/C Trigger (True or Comp Op)		ON True	OFF Comp	OFF Comp	ON True
If Carry Out		Set Overflow	No Recomp	No Recomp	Set Overflow
If No Carry Out		No Recomp	Recomp	Recomp	No Recomp
Resulting Sign	No Recom	+	+	-	
	Recomp		-	+	
NOTES:		T/C Turned ON During I-Cycle High/Plus & Equal/Zero turned ON in Parallel with Trg 11			
If the Equal/Zero Trigger is ON (Zero Sum) when the I-Cycle following the Add Operation is Entered, the High/Plus Trigger is turned OFF (- Sign).					

Figure 10-6. Sign Analysis Chart

Trigger 12. (01.60.12.1).

1. Read out of memory per OR-2 and store the digit in MDR.
2. Write back OR-2 bypassed (to save address for writing the sum digit into memory during trigger 14 time).
3. Change the status of the τ/c trigger on first cycle if P field is negative.
4. Turn OFF high/plus trigger on first cycle if P field is negative.
5. Turn ON field mark No. 2 trigger after first cycle when end of P field is reached.

For complement operation or recomp:

6. Turn ON the carry in trigger during first cycle to obtain 10's complement in units position.

For recomp:

7. Set D/B REG units from MDR.

Trigger 13. (01.60.13.1).

1. Develop and set add table address in MAR (see Figure 10-7).
2. Read out of memory per add table address and store the sum digit in MDR.
3. Turn ON the carry out trigger if MDR contains an F bit (carry).
4. Turn on the carry out trigger if the output of the τ/c switch is a "9" and the carry in trigger is on.
5. Reset D/B REG units. (See trigger 21 Objective 2.)

For recomp:

6. Turn ON recomp control trigger if during complement operation there is no carry out on add cycle in which field mark No. 2 is reached.
7. Set C bit latch in tens position of MAR to supply a zero for development of add table address.
8. Change status of high/plus trigger if recombination is required.

Miscellaneous:

9. Set overflow trigger if:
 - (a) On a true operation a carry out is detected on the add cycle in which field mark No. 2 is reached.
 - (b) The P field is shorter than the Q field. (Field mark No. 2 trigger turned ON before field mark No. 1 trigger.)
10. Turn OFF equal/zero trigger if add table digit is not a "zero."

Trigger 14. (01.60.14.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Write back OR-2 decremented.

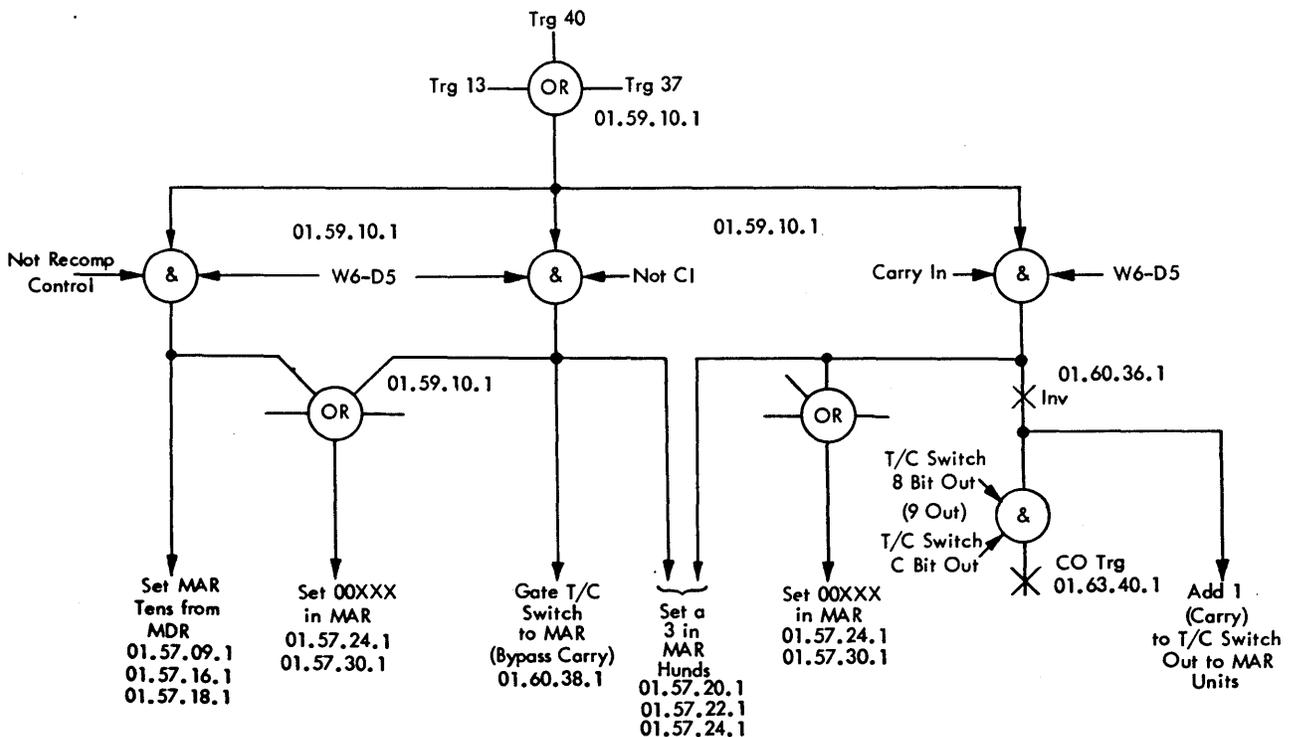


Figure 10-7. Set MAR Add Table Address

4. Set an F bit in MDR on first add cycle and first recomp cycle if high/plus trigger is OFF (minus sign).
5. Clear the F bit on first add cycle if high/plus trigger is ON (plus sign) and add table digit contains a carry.
6. Set an F bit in MDR for field definition on the add cycle and recomp cycle in which field mark No. 2 trigger is turned ON.
7. Clear F bit (carry) after first add cycle and before the add cycle in which field mark No. 2 trigger is turned ON.
8. Transfer MDR to MBR.
9. Turn ON carry in trigger if carry out trigger is ON.
10. Turn ON trigger 11 if neither field mark No. 1 nor field mark No. 2 has been reached to repeat the four memory cycles (triggers 11, 12, 13 and 14) for the next higher order digits.
11. Turn ON trigger 21 if field mark No. 1 has been reached and field mark No. 2 has not.
12. Enter I-cycle for next instruction if field mark No. 2 has been reached and recomp is not necessary.

For recomp:

13. Turn ON trigger 21 and first cycle trigger for the first recomp cycle.
14. Turn ON trigger 23 after the first recomp cycle if field mark No. 2 has not been reached.
15. Enter I-cycle for next instruction if field mark No. 2 has been reached.

Trigger 21. (01.60.31.1).

Not recomp: (Replaces trigger 11 after field mark No. 1 has been reached.)

1. Block MAR reset to prevent VRC error.
2. Set zero in D/B REG units. (This will provide zeros in place of the Q field digits previously set in D/B REG units by trigger 11.) The D/B REG units was reset during trigger 13 time.

Recomp: (Replaces trigger 11 on first recomp cycle.)

1. Transfer OR-3 (address of units position of P field) into OR-2.
2. Reset OFF field mark No. 1 trigger.
3. Reset OFF field mark No. 2 trigger.
4. Turn ON recomp trigger.

Trigger 23. (01.60.32.1).

Used only if a recomplement is required. (Replaces trigger 11 or 21 for first memory cycle of second and succeeding recomp cycles.)

1. Block MAR reset to prevent VRC error.
2. Turn ON trigger 12.

Add Immediate (Code 11 — AM)

Sequence block diagrams for this operation are shown in Figures 10-3 and 10-4. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

Objective

Add the field whose units position is the Q_{11} digit of the add immediate instruction to the field at the P address (OR-2) and store the algebraic sum at the P address (OR-2) and successively lower memory locations.

Functions

The add immediate operation differs from the Add-Code 21 operation only in the method of setting the Q field address in OR-1 during the I-cycle. See Add Code 21.

Auxiliary Triggers

See Add Code 21.

E-Timer Trigger Objectives

See Add Code 21.

Subtract (Code 22 — S)

Sequence block diagrams for this operation are shown in Figures 10-8 and 10-9. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

Objective

Subtract the field at the Q address (OR-1) from the field at the P address (OR-2), and store the algebraic difference at the P address (OR-2) and successively lower memory locations.

Functions

The location of the subtrahend (Q field) is specified by OR-1. The location of the minuend (P field) is specified by OR-2. The algebraic differences are stored in the P field specified by OR-2. The Q field remains unchanged. Minimum field length for either the P or Q fields is two digits.

Four memory cycles are required for the development of each digit in the algebraic difference as follows:

1. One digit of the Q field is read from memory per OR-1 to the units position of the digit/branch register.

$\overline{082}$ (+)
54 (+)

082
946
C.O. ← 028

$\overline{082}$ (-)
54 (+)

082
54
136

No Recomp on Carry Out

No Recomp on TRUE

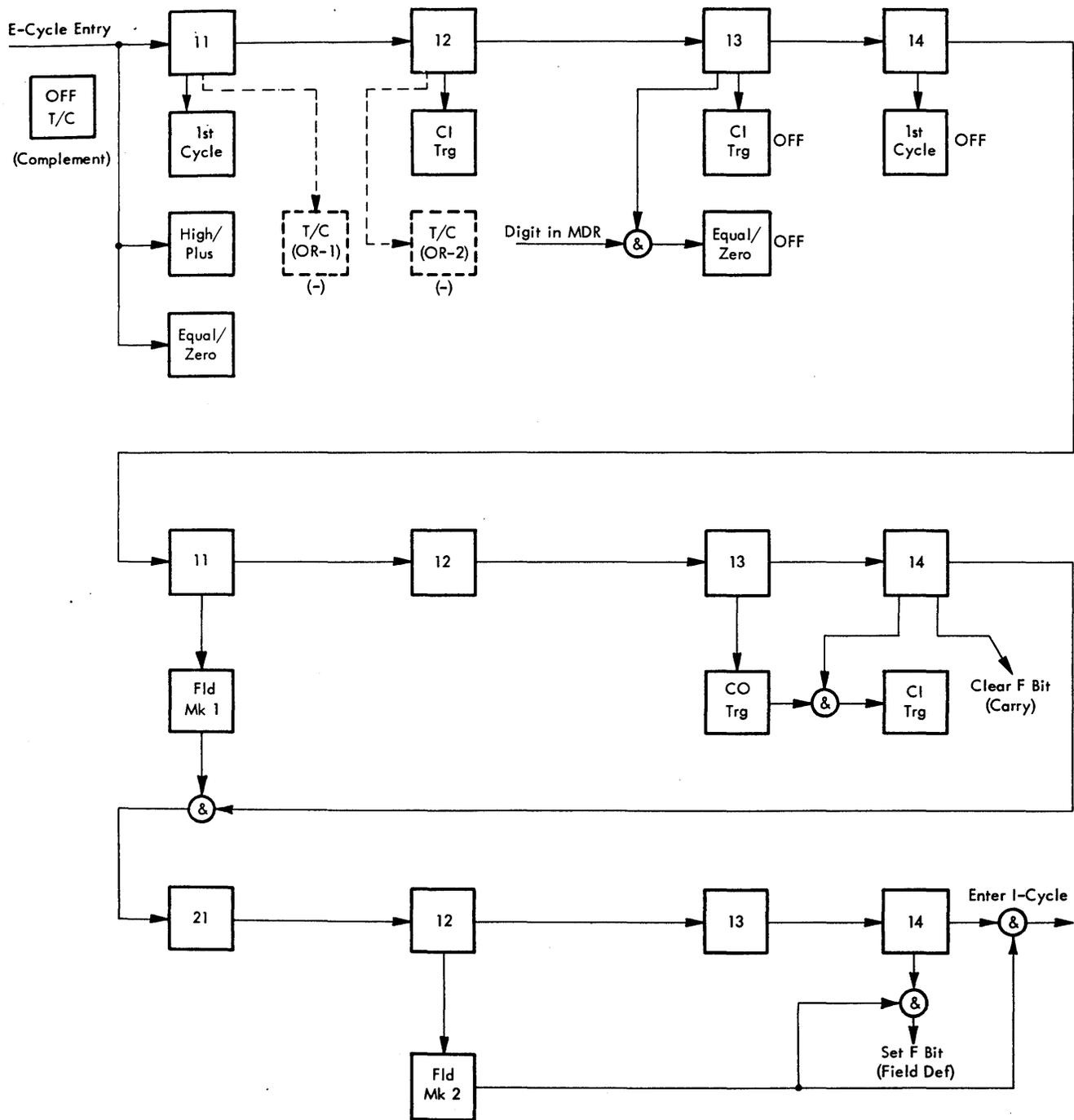


Figure 10-8. Subtract—Code 22/12

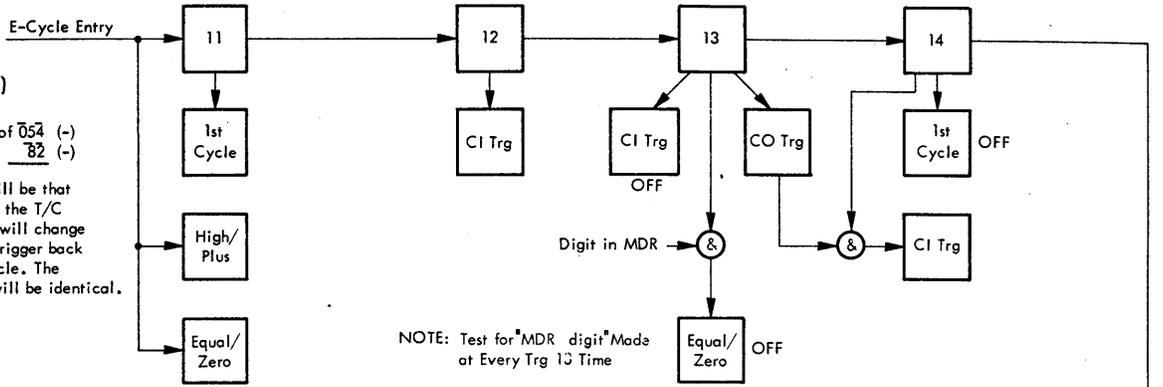
Subtract $\overline{054 (+)}$ $\overline{82 (+)}$ $\overline{054}$ $\overline{918}$
 No. C O $\overline{972}$ Recomp $\overline{028}$ Note Sign (-)

OFF
T/C

(Complement)

NOTE: In the case of $\overline{054 (-)}$ $\overline{82 (-)}$

The only difference will be that trigger 11 will turn on the T/C trigger and trigger 12 will change the status of the T/C trigger back to OFF on the first cycle. The rest of the operation will be identical.



NOTE: Test for "MDR digit" Made at Every Trg 13 Time

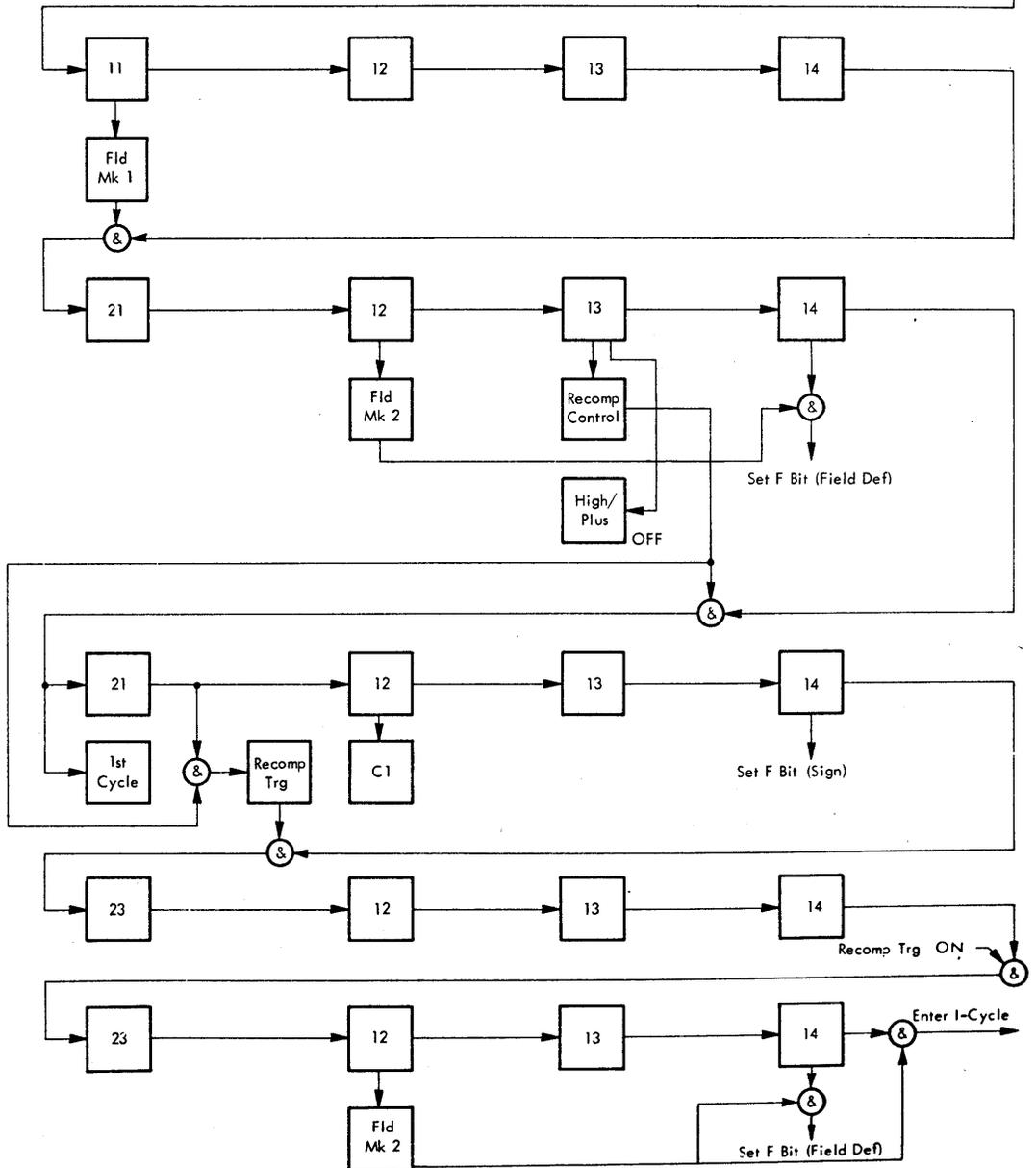


Figure 10-9. Subtract with Recomplement

2. The corresponding digit of the P field is read from memory per OR-2 to MDR.
3. The P field digit is transferred from MDR to the tens position of MAR. The Q field digit stored in the digit/branch register is transferred to the units position of MAR. (NOTE: The Q field digit may be complemented before it is placed in MAR, depending upon the signs of the P and Q fields.) This manufactured address with a 3 placed into the hundreds position of MAR is used to read the one digit difference from the add table, Figure 10-1, in memory to MDR.
4. The one digit difference is then written back into memory from MDR to the correct digit position (per OR-2) replacing the P field digit.

Two types of carries can result from a subtract operation.

1. A carry resulting from the subtraction of two digits is noted by a flag bit with the appropriate one digit difference in the add table. This flag bit turns on the carry out trigger.
2. A carry resulting from the addition of a previous carry to a "nine" output of the τ/c switch is noted by turning on the carry out trigger.

Detection of a carry (carry out trigger on) sets a logical path for the next Q field digit so that it is increased by one before it is used to address the add table in the next subtract cycle.

Recomplement is required for any subtract operation where the signs of the P and Q fields are initially the same and the absolute value of the P field is less than the absolute value of the Q field.

Subtraction proceeds serially, one digit at a time, building up partial differences from the low-order to the high-order digit of the difference field until the operation is terminated by a flag bit, (field mark No. 2), in the high-order position of the P field. The high-order digit of the difference is marked by storing a flag bit. The sign of the difference is marked by the presence or absence of a flag bit in the units position of the difference. (A flag bit indicates minus sign. No flag indicates plus sign.)

The number of digits in the difference is equal to the number of digits in the P field. For a complete difference to be formed, the number of digits in the P field must be greater than or equal to the number of digits in the Q field. If this rule is violated, the overflow indicator is turned ON and the subtraction is performed using only as many Q field digits as there are positions in the P field. (The extra digits in the Q field are not used.) The algebraic difference of the two equal length fields is then obtained.

Resulting Indicator Conditions

1. The high/plus indicator is ON if the difference is positive and is OFF if the difference is negative or zero.
2. The equal zero indicator is ON if the difference is zero and is turned OFF if the difference is not zero.
3. The overflow indicator is turned ON if an overflow occurs; the overflow digit is lost. If the overflow indicator is ON as a result of a previous arithmetic operation, a no overflow condition on this subtract instruction will not turn it OFF.

NOTE: Once the high/plus or equal/zero indicators are turned ON or OFF by an arithmetic or compare operation they will retain that state until the next arithmetic or compare operation is executed (unless the reset key on the console is depressed which turns OFF these indicators). Testing them with a branch indicator or branch no indicator instruction has no effect on their state. The overflow indicator will be turned OFF only by testing it with a branch indicator or branch no indicator instruction (or by depressing the reset key on the console).

Auxiliary Trigger Status

True/Complement (T/C) Trigger. (01.63.20.1).

1. Turned ON by trigger 1 during I-cycle.
2. Turned OFF by the E-cycle entry trigger during trigger 8 time when a subtract or subtract immediate operation code has been set in the OP REG and therefore will be OFF when E-cycle is entered.
3. Turned ON during trigger 11 time on first add cycle if Q field is negative.
4. Status changed during trigger 12 time on first add cycle if P field is negative.
5. Operation will be true if τ/c trigger is ON at the end of trigger 12 time of the first add cycle, and complement if τ/c trigger is OFF. Complement operation requires that the Q field be complemented (10's complement in units position; 9's complement in other positions) for the development of add table addresses.

Other Auxiliary Triggers. Refer to Add - Code 21 for status of other auxiliary triggers.

Sign Analysis

The sign analysis chart, Figure 10-10, shows the operation of the τ/c , high/plus, and equal/zero triggers for a subtract operation. The chart shows the conditions requiring a recomplement operation and the conditions causing an overflow.

		SUBTRACT			
Sign of P Field (Minuend)		+	+	-	-
Sign of Q Field (Subtrahend)		+	-	+	-
Set High/Plus Trg (Store Sign of P Field)		ON +	ON +	OFF -	OFF -
Set T/C Trigger (True or Comp Op)		OFF Comp	ON True	ON True	OFF Comp
If Carry Out		NO Recomp	Set Overflow	Set Overflow	NO Recomp
If No Carry Out		Recomp.	NO Recomp	NO Recomp	Recomp.
Resulting Sign	No Recom	+	+	-	-
	Recomp	-			+
NOTES: If the Equal/Zero Trigger is ON (Zero Difference) when the I-Cycle following the Subtract Operation is Entered, the High/Plus Trigger is turned OFF (- Sign)		T/C turned ON and then OFF During I-Cycle High/Plus and Equal/Zero turned ON in Parallel with Trg 11 at E-Cycle Entry Time			

Figure 10-10. Sign Analysis Chart

E-Timer Trigger Objectives

Trigger 11. (01.60.11.1).

1. Read out memory per OR-1 and store the digit in D/B REG units.
2. Write back OR-1 decremented.
3. Turn ON field mark No. 1 trigger after first cycle when end of Q field is reached.
4. Turn ON T/C trigger during first cycle if Q field is negative.

Trigger 12. (01.60.12.1).

1. Read out of memory per OR-2 and store the digit in MDR.
2. Write back OR-2 bypassed (to save address for writing the difference digit into memory during trigger 14 time).

3. Change the status of the T/C trigger on first cycle if P field is negative.
4. Turn OFF high/plus trigger on first cycle if P field is negative.
5. Turn ON field mark No. 2 trigger after first cycle when end of P field is reached.

For complement operation or recomp:

6. Turn ON the carry in trigger during first cycle to obtain 10's complement in units position.

For recomp:

7. Set D/B REG units from MDR.

Other E-Timer Triggers. Refer to Add-Code 21 for objectives of other E-timer triggers. Other E-timer triggers used are: 13, 14 and 21 (also 23 if complement is required).

Subtract Immediate (Code 12 — SM)

Sequence block diagrams for this operation are shown in Figures 10-8 and 10-9. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

Objective

Subtract the field whose units position is the Q_{11} digit of the subtract immediate instruction from the field at the P address (OR-2), and store the algebraic difference at the P address (OR-2) and successively lower memory locations.

Functions

The subtract immediate operation differs from the Subtract-Code 22 operation only in the method of setting

the subtrahend (Q field) address in OR-1 during the I-cycle. See Subtract - Code 22.

Auxiliary Triggers

See Subtract - Code 22.

E-Timer Triggers

See Subtract - Code 22.

Compare (Code 24 — C)

Sequence block diagrams for this operation are shown in Figures 10-11, 10-12, 10-13, and 10-14. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

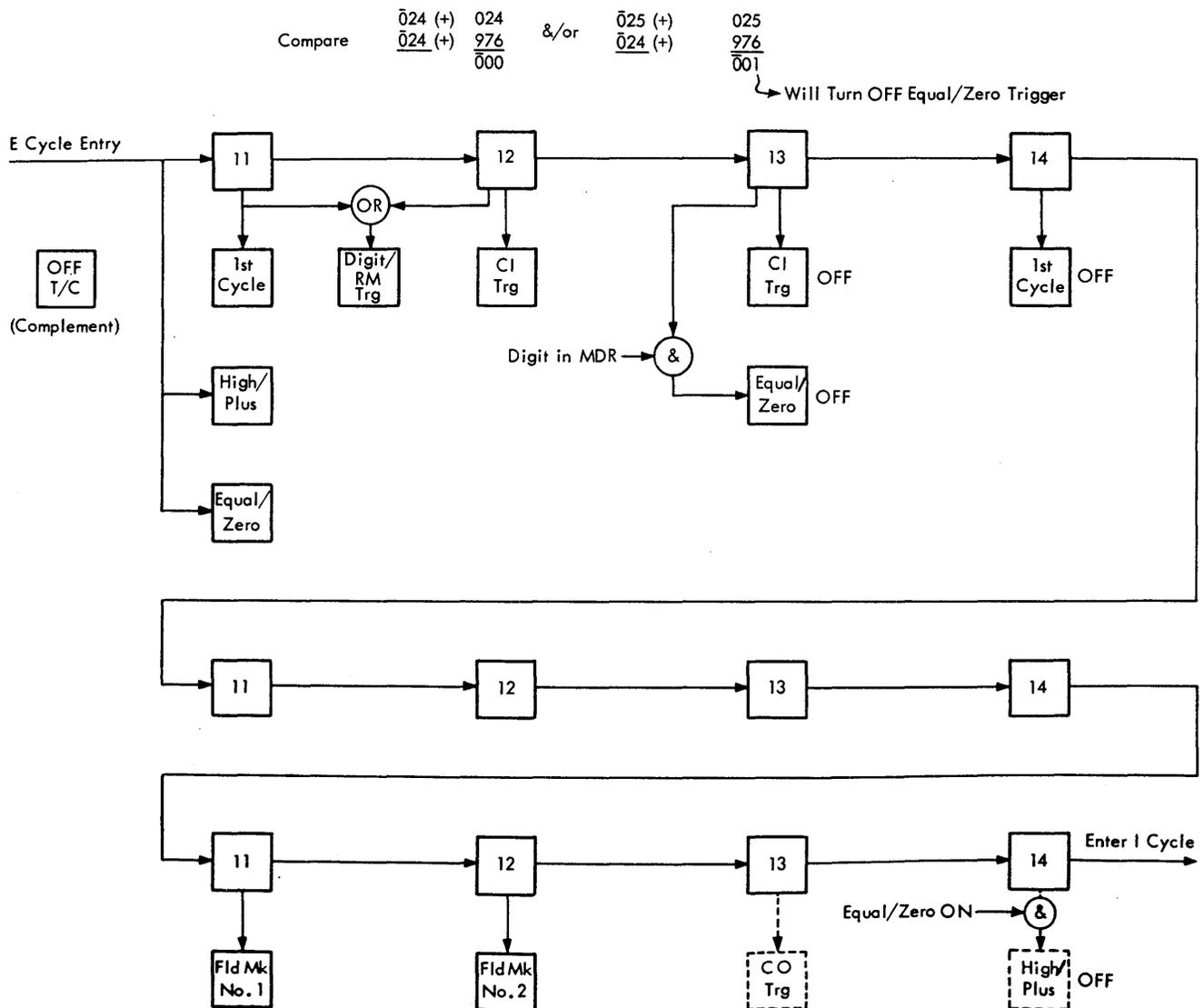


Figure 10-11. Compare-Code 24/14 (Plus Signs)

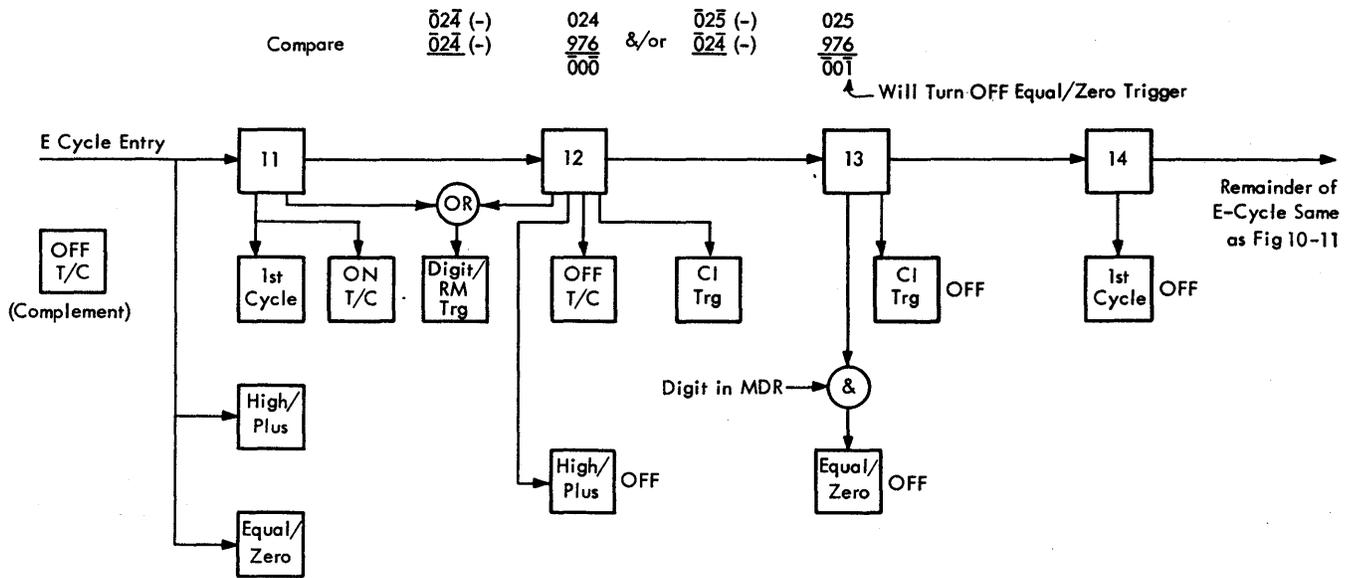


Figure 10-12. Compare—Code 24/14 (Minus Signs)

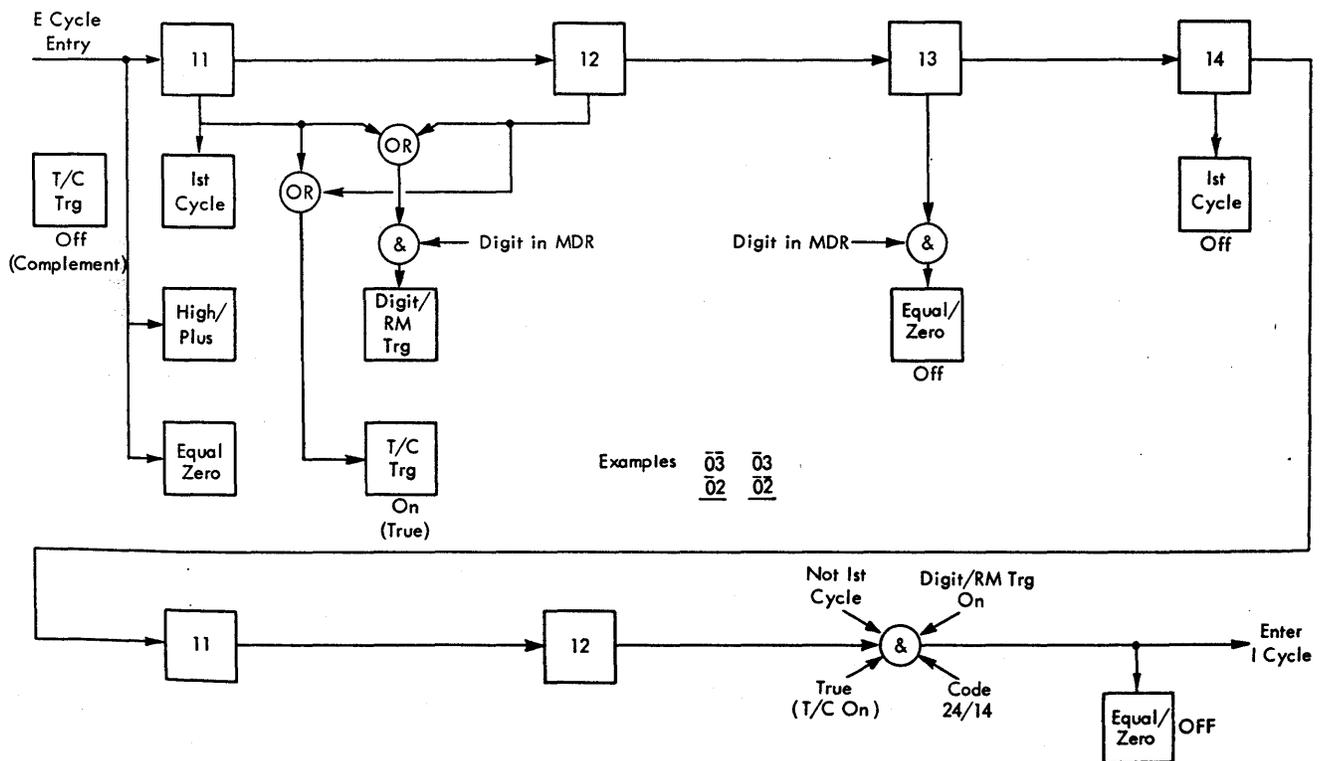


Figure 10-13. Compare—Code 24/14 Significant Digits—(Unlike Signs)

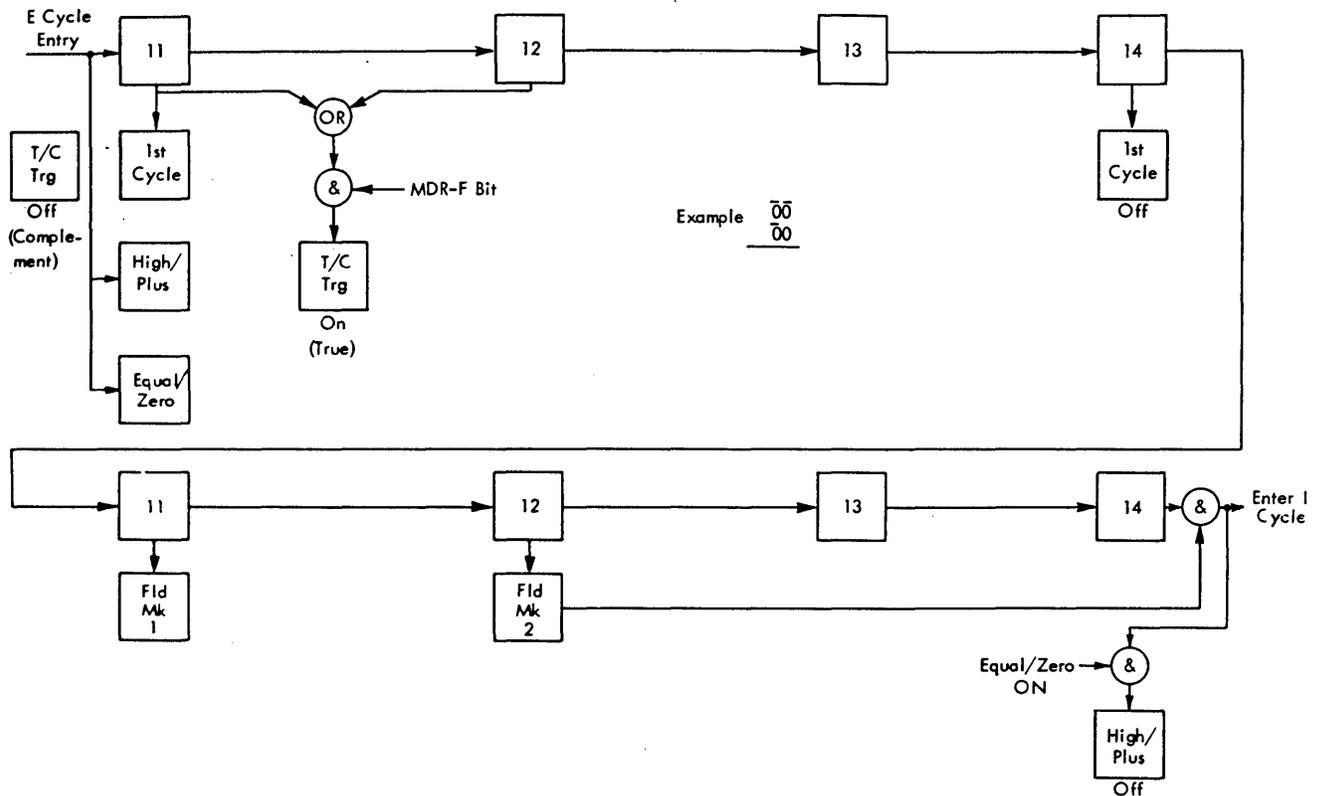


Figure 10-14. Compare—Code 24/14—Zeros—(Unlike Signs)

Objective

Compare the field at the Q address (OR-1) to the field at the P address (OR-2) to determine if the P field is algebraically higher than or algebraically equal to the Q field with signs taken into consideration.

Functions

The high or equal condition of the field specified by OR-2 relative to the field specified by OR-1 is established, considering signs. Both fields remain unchanged after the comparison has been completed.

Four memory cycles are required to compare each position of the two fields as follows:

1. One digit from the Q field is read from memory per OR-1 to the units position of the digit/branch register.
2. The corresponding digit of the P field is read from memory per OR-2 to MDR.
3. The P field digit is transferred from MDR to the tens position of MAR and the digit stored in the digit/branch register is complemented (tens complement in units position; nines complement in other positions) and placed in the units position of MAR. This manufactured address with a 3 placed

into the hundreds position of MAR is used to read the one digit difference from the add table in memory, Figure 10-1, to MDR.

4. The difference digit in MDR is not written back into memory and therefore is lost. The fourth cycle decrements the P address (OR-2) in preparation for the next compare cycle.

The comparison proceeds serially, one digit at a time, from low-order to high-order digits of the compared fields until the operation is terminated by a flag bit (field mark No. 2) in the high-order position of the P field. The comparison is performed internally by subtraction of the Q field from the P field. However, the digits of the difference are lost.

If the signs of the two fields are initially different and one or both of the fields contains a significant digit (not zero) the compare operation is terminated after a digit in one or both fields has been interrogated. The field whose sign is positive is declared high. A comparison of zeros with unlike signs results in an equal comparison (the 1620 considers a plus zero as being equal to a minus zero).

A comparison may be executed on two fields containing a random mixture of alphameric and special

characters or on two numerical fields. A legitimate comparison may not be executed on an alphameric field and a numerical field without first expanding the numerical field to the alphameric format (or vice versa).

For a complete comparison to be performed, the number of digits in the P field must be greater than or equal to the number of digits in the Q field (unless their signs are not alike). If this rule is violated, the overflow indicator is turned ON and the extra digits in the Q field are not used; however, the result of the comparison is correct to the point where comparison is terminated.

Minimum length for either of the fields being compared is two digits.

Resulting Indicator Conditions

1. The high/plus indicator is ON if the P field is higher than the Q field and is OFF if not higher with signs taken into consideration.
2. The equal/zero indicator is ON if the P field is equal to the Q field and is OFF if not equal.
3. The overflow indicator is turned ON if an overflow occurs. If the overflow indicator is ON as the result of a previous arithmetic operation, a no overflow condition of this compare instruction will not turn it OFF.

NOTE: Once the high/plus or equal/zero indicators are turned ON (or OFF) by an arithmetic or compare operation, they will retain that state until the next arithmetic or compare operation is executed (unless the reset key on the console is depressed which turns OFF these indicators). Testing them with a branch indicator or branch no indicator instruction has no effect on their state. The overflow indicator will be turned OFF only by testing it with a branch indicator or branch no indicator instruction (or by depressing the reset key on the console).

Auxiliary Trigger Status

True/Complement (T/C) Trigger. (01.63.20.1).

1. Turned ON by trigger 1 during I-cycle.
2. Turned OFF by the E-cycle entry trigger during trigger 8 time when a compare or compare immediate operation code has been set in the OP REG, and therefore will be OFF when E-cycle is entered.
3. Turned ON during trigger 11 time on first compare cycle if Q field is negative.
4. Status changed during trigger 12 time on first compare cycle if P field is negative.
5. The T/C trigger must be OFF (designating a complement operation) at the end of trigger 12

time if a complete compare operation is to take place. Complement operation requires that the Q field be complemented (10's complement in units position; 9's complement in other positions) for the development of add table addresses.

6. The T/C trigger will be ON (designating a true operation) at the end of trigger 12 time if the P and Q fields have unlike signs.

Decrement Trigger. (01.60.05.1).

1. Turned ON (decrement) during trigger 8 time of the preceding I-cycle.
2. Will remain ON until the next I-cycle is entered.

First Cycle Trigger. (01.63.10.1).

1. Turned ON by trigger 11 when E-cycle is entered.
2. Turned OFF at end of trigger 14 time on first compare cycle.

High/Plus Trigger. (01.60.40.1).

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 12 time on first compare cycle if P field is negative.
3. Status changed during trigger 13 time of the compare cycle in which field mark No. 2 is reached if there is not a carry out.
4. Turned OFF by trigger 14 when I-cycle is entered if the equal/zero trigger is on (comparison of zeros).

Equal/Zero Trigger. (01.60.41.1).

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 13 time if add table digit is other than zero.
3. Turned OFF during trigger 12 time if the P and Q fields have unlike signs, if it is not first cycle, and if a significant digit and a zero or significant digits (not zero) are being compared.

Field Mark No. 1 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first compare cycle (Q field must be a minimum of two digits).
3. Will be turned ON during trigger 11 time on compare cycle in which end of Q field is reached.

Field Mark No. 2 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first compare cycle (P field must be a minimum of two digits).
3. Will be turned ON during trigger 12 time on compare cycle in which end of P field is reached.

Carry Out Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.

- Reset OFF by triggers 11, 21, and 23.
- Turned ON during trigger 13 time on compare cycles where the add table digit in MDR contains an F bit (carry).
- Turned ON during trigger 13 time on compare cycles where the output of the τ/c switch is a "9" and the carry in trigger is on.

Carry In Trigger. (01.63.40.1).

- Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
- Turned ON during trigger 12 time on first compare cycle to obtain 10's complement in units position.
- Reset OFF during trigger 13 time.
- Turned ON by trigger 14 if the carry out trigger is ON.

Digit/RM Trigger. (01.63.50.1)

- Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
- Turned ON during trigger 11 time if MDR contains a digit (not zero).
- Turned ON during trigger 12 time if MDR contains a digit (not zero).

Sign Analysis

The sign analysis chart, Figure 10-15, shows the operation of the τ/c , high/plus, and equal/zero triggers for a compare operation.

E-Timer Trigger Objectives

Trigger 11. (01.60.11.1).

- Read out of memory per OR-1 and store the digit in D/B REG units.
- Write back OR-1 decremented.
- Turn ON field mark No. 1 trigger after first cycle when end of Q field is reached.
- Turn ON τ/c trigger during first cycle if Q field is negative.
- Turn ON digit/RM trigger if MDR contains a significant digit (not zero).

Trigger 12. (01.60.12.1).

- Read out of memory per OR-2 and store the digit in MDR.
- Write back OR-2 bypassed.
- Change the status of the τ/c trigger on first cycle if P field is negative.
- Turn OFF high/plus trigger on first cycle if P field is negative.
- Turn ON field mark No. 2 trigger after first cycle when end of P field is reached.
- Turn ON the carry in trigger during first cycle to obtain 10's complement in units position.
- Turn ON digit/RM trigger if MDR contains a significant digit (not zero).
- Enter I-cycle for next instruction in sequence if it is not the first cycle and a significant digit (not zero) in fields with unlike signs is compared. (I-cycle entry under these conditions turns off the equal/zero trigger.)

	Digits or Zeros		Digits		Zeros	
	+	-	+	-	+	-
OR-2	+	-	+	-	+	-
OR-1	+	-	-	+	-	+
T/C	OFF (Complement)	OFF (Complement)	ON (True)	ON (True)	ON (True)	ON (True)
High/Plus	ON (+) Turn OFF if "P" is Lower than "Q"	OFF (-) Turn ON if "P" is Higher than "Q"	ON (+)	OFF (-)	ON Turned OFF When I-Cycle is Entered	OFF
Equal/Zero	Leave ON if Diff is Zero Turn OFF if Not	Leave ON if Diff is Zero. Turn OFF if Not.	Turn OFF	Turn OFF	ON	ON

Figure 10-15. Sign Analysis Chart

Trigger 13. (01.60.13.1).

1. Develop and set add table address in MAR (see Figure 10-7).
2. Read out of memory per add table address and store the digit in MDR.
3. Turn ON the carry out trigger if MDR contains an F bit (carry).
4. Turn ON the carry out trigger if the output of the T/C switch is a "9" and the carry in trigger is on.
5. Reset D/B REG units. (See trigger 21 Objective 1.)
6. Change status of high/plus trigger if there is no carry out on the cycle in which field mark No. 2 is reached.

Miscellaneous:

7. Turn on the overflow trigger if the P field is shorter than the Q field.
8. Turn OFF equal/zero trigger if add table digit is not a zero.

Trigger 14. (01.60.14.1).

1. Read into MAR from OR-2 and write back OR-2 decremented.
2. Turn ON carry in trigger if carry out trigger is ON.
3. Turn ON trigger 11 if neither field mark No. 1 nor field mark No. 2 has been reached and repeat the four memory cycles (triggers 11, 12, 13 and 14) for the next higher order digits.
4. Turn ON trigger 21 if field mark No. 1 has been reached and field mark No. 2 has not.
5. Enter I-cycle for next instruction in sequence if field mark No. 2 has been reached.

Trigger 21. (01.60.31.1).

NOTE: Trigger 21 replaces trigger 11 for first memory cycle after field mark No. 1 has been reached.

1. Set zero in D/B REG units. (This will provide zeros in place of the Q field digits previously set into D/B REG units by trigger 11.) The D/B REG units was reset during trigger 13 time.

Compare Immediate (Code 14 — CM)

Sequence block diagrams for this operation are shown in Figures 10-11, 10-12, 10-13 and 10-14. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 11 and 12.

Objective

Compare the field whose units position is the Q_{11} digit of the compare immediate instruction to the field at the P address (OR-2) to determine if the field at the P address is higher than or equal to the field in the Q part of the instruction, signs considered.

Functions

The compare immediate operation differs from the Compare - Code 24 operation only in the method of setting the OR-1 address during the I-cycle. See Compare - Code 24.

Auxiliary Triggers

See Compare - Code 24.

E-Timer Triggers

See Compare - Code 24.

Multiply (Code 23 — M)

The sequence block diagram for this operation is shown in Figure 10-16. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 15, 17, and 19.

Objective

Multiply the field at the P address (OR-2) by the field at the Q address (OR-1), and store the algebraic product at location 00099 and successively lower memory locations.

Functions

The location of the multiplier is specified by OR-1 (Q field). The location of the multiplicand is specified by OR-2 (P field). Minimum field length for either the multiplier or the multiplicand is two digits. Both the multiplier and the multiplicand remain unchanged when multiplication is completed.

Multiplication proceeds serially, one digit at a time, building up partial products from the low-order to the high-order digit of the product field. The operation is terminated when the high-order digit of the multiplicand field has been multiplied by the high-order digit of the multiplier field. The high-order position of the product is marked by storing a flag bit. A negative product is marked by a flag bit in the units position; absence of a flag bit indicates a positive product.

The number of digits in the product is limited only by available memory space, and is equal to the sum of the number of digits in the multiplier and in the multiplicand. Twenty positions for the product area in memory (00080-00099) are cleared to zeros at the beginning of the multiply operation. The length of the product may exceed 20 digits if a program is written to clear to zero the required number of locations in excess of 20 before the multiply instruction is executed. If the product exceeds 100 digits, the 101st digit (high-order end of product) will be placed in

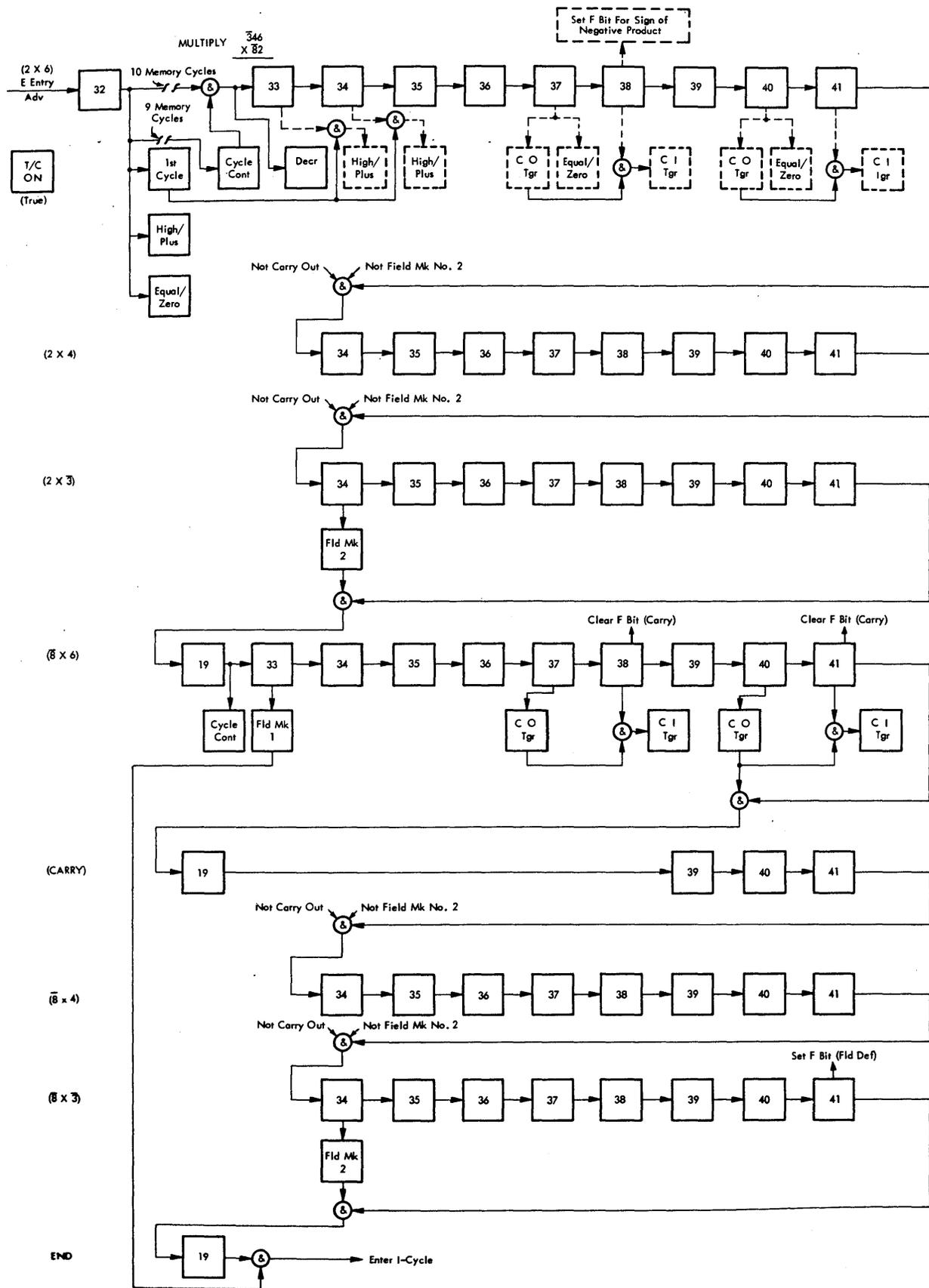


Figure 10-16. Multiply—Code 23/13

location 19999 and the remaining digits in successively lower memory locations.

A minimum of eight memory cycles are required to develop and store in the product area of memory each digit of the product as follows:

1. One digit of the multiplier is read from memory per OR-1 to the multiplier/quotient (M/Q) register where it is stored for use with each digit of the multiplicand.
2. The multiplicand digit is read from memory per OR-2 to MDR.
3. The multiplicand digit is transferred from MDR to the tens position of MAR and the multiplier digit is passed through a doubler circuit to form two digits which are entered into the hundreds and units positions of MAR. (The digit entering the hundreds position is increased by one before being placed in MAR) see Figure 10-17. This manufactured address is used to read the two digit product (whose digits are in reverse order) from the multiply table stored in memory in locations 00100 through 00299, Figure 10-2. The two digit product is stored (with digits in the correct order) in the digit/branch register.
4. The appropriate digit is read from the product area in memory per PR-2 (PR-1 on the first cycle associated with each new multiplier digit) to MDR.
5. The digit in the units position of the digit/branch register is transferred to the units position of MAR and the product area digit in MDR is transferred to the tens position of MAR. This manufactured address with a "3" forced into the hundreds position of MAR is used to read the one digit sum from the add table in memory to MDR (Figure 10-1)

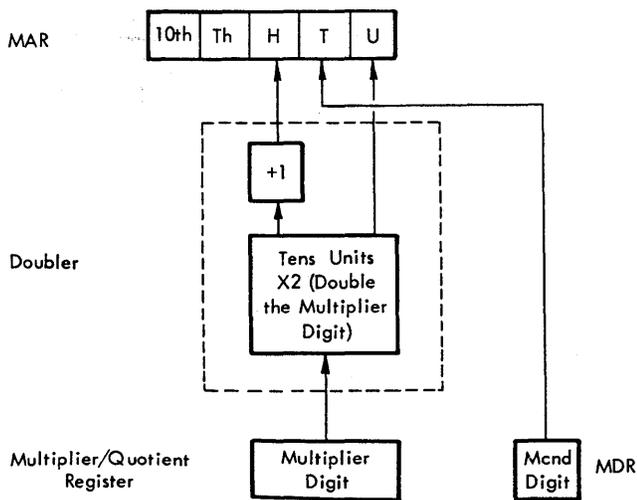


Figure 10-17. Data Flow—Multiply Table Address

6. The one digit sum is then written back into memory from MDR to the appropriate position of the product area per PR-2. PR-2 is written back decremented.
7. The next higher order digit is read from the product area per PR-2 to MDR.
8. The digit in the tens position of the digit/branch register is transferred to the units position of the digit register and from there to the units position of MAR. The product area digit in MDR is transferred to the tens position of MAR. This manufactured address with a "3" forced into the hundreds position of MAR is used to read the one digit sum from the add table to MDR.
9. The one digit sum is then written back into memory from MDR to the appropriate position of the product area per PR-2.
10. Carries, if any, are added to the partial product before the next multiplication cycle begins. See Chart of Trigger Objectives and Sequence, Figure 10-18.

Resulting Indicator Conditions

1. The high/plus indicator is ON if the product is positive and OFF if the product is negative or zero.
2. The equal/zero indicator is ON if the product is zero and is OFF if the product is not zero.

NOTE: Once the high/plus or equal/zero indicators are turned ON or OFF by an arithmetic or compare operation, they will retain that state until the next arithmetic or compare operation is executed (unless the reset key on the console is depressed which turns these indicators OFF). Testing them with a branch indicator or branch no indicator instruction has no effect on their state.

Auxiliary Trigger Status

True/Complement Trigger. (01.63.20.1).

1. Turned ON (true) by trigger 1 of the I-cycle and remains ON throughout the multiply operation to provide a path for transfer of digits from the units position of the digit/branch register to the units position of MAR.

Decrement Trigger. (01.60.05.1).

1. Turned OFF (increment) by the I-cycle trigger.
2. Turned ON (decrement) by trigger 33 at end of product area clearing operation.
3. Remains ON for remainder of multiply operation.

Equal/Zero Trigger. (01.60.41.1).

1. Turned ON by trigger 32.
2. Turned OFF during trigger 37 or trigger 40 time if the add table digit is other than zero.

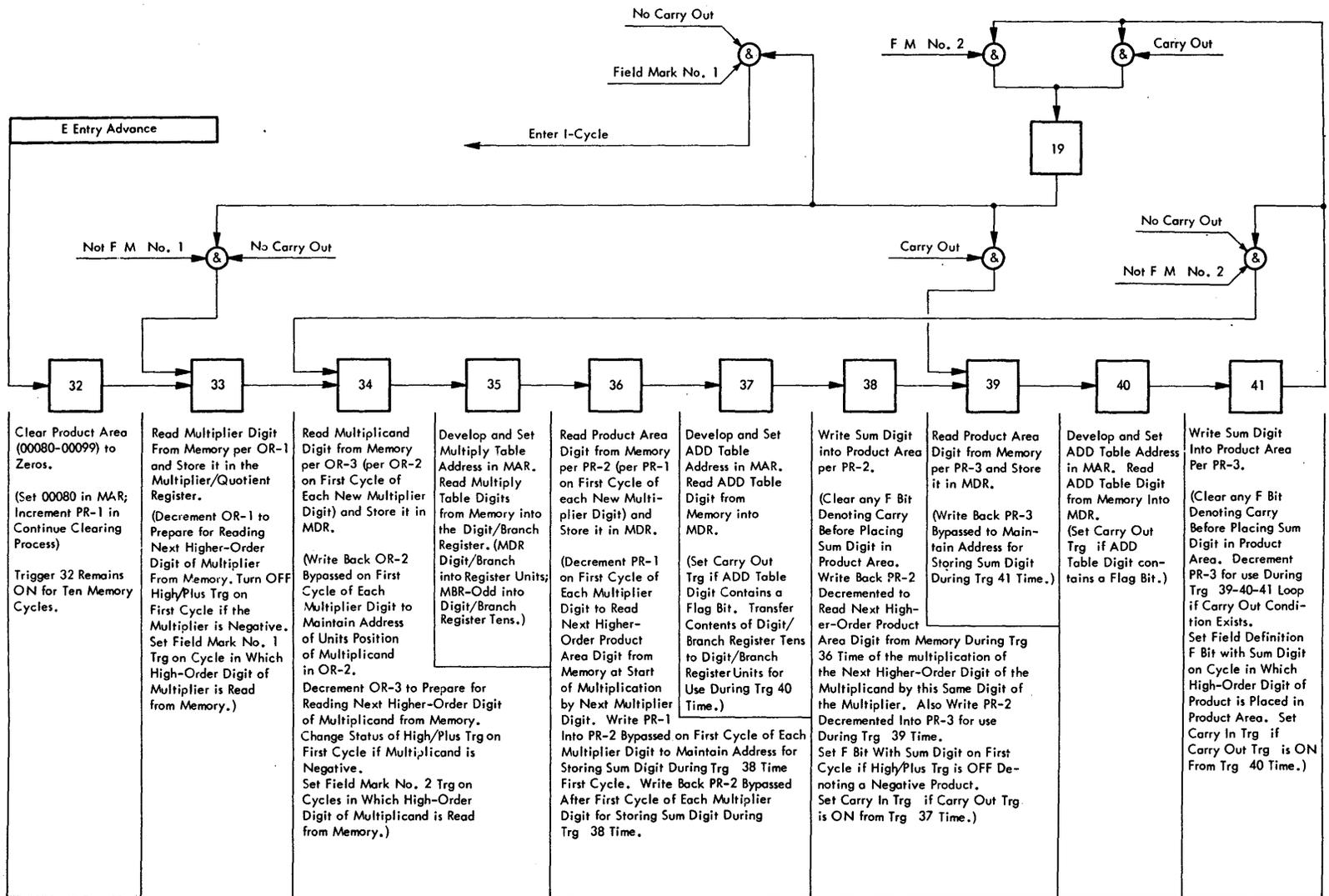


Figure 10-18. Multiply—Flow Diagram

First Cycle Trigger. (01.63.10.1).

1. Turned ON by trigger 32.
2. Turned OFF by trigger 39 on first multiply cycle.

High/Plus Trigger. (01.60.40.1)

1. Turned ON by trigger 32.
2. Turned OFF during trigger 33 time on first multiply cycle if the Q field (OR-1) is negative.
3. Status changed during trigger 34 time on first multiply cycle if the P field (OR-2) is negative.
4. Status at the end of trigger 34 time designates the sign of the product.

Cycle Control Trigger. (01.62.31.1).

1. Turned ON during trigger 32 time when the address in MAR reaches 00096 to terminate the product area clearing operation.
2. Turned OFF by trigger 39.
3. Turned ON by trigger 33 when the next multiplier digit is required.

Carry Out Trigger. (01.63.40.1).

1. Reset OFF by trigger 35 to ensure OFF status for entry into multiply operation.
2. Turned ON during trigger 37 or trigger 40 time on add cycles where the add table digit in MDR contains an F bit (carry).
3. Turned ON during trigger 40 time on add cycles where the output of the r/c switch is a "9" and a carry exists (CI trigger on) from the previous add cycle.
4. Reset OFF during trigger 39 time.

Carry In Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
2. Reset OFF during trigger 35 time.
3. Turned ON by trigger 38 or trigger 41 if the carry out trigger is ON.
4. Reset OFF during trigger 40 time.

Field Mark No. 1 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first multiply cycle (Q field must be a minimum of two digits).
3. Turned ON during trigger 33 time on the multiply cycle in which end of Q field (OR-1) is reached.

Field Mark No. 2 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first multiply cycle (P field must be a minimum of two digits).
3. Turned ON during trigger 34 time on multiply cycles in which end of P field (OR-2) is reached.
4. Turned OFF by trigger 33.

E-Timer Trigger Objectives

Trigger 32. (01.62.30.1).

1. Turn ON first cycle trigger.
2. Turn ON high/plus trigger.
3. Turn ON equal/zero trigger.
4. Set 00080 in MAR on first memory cycle.
5. Read out of memory per 00080, with both the even and odd sense amplifiers blocked (Block Memory SA), to clear product area memory locations 00080 and 00081.
6. Write into PR-1 the MAR address (00080) incremented plus 2.
7. Set C bits into MBR-even and MBR-odd to provide zeros for the product area.
8. Write into memory per MAR address (00080) from MBR.

Trigger 32 remains ON for a total of ten memory cycles during which time the 20 memory locations designated as product area (00080-00099) are set to zero. Memory locations per PR-1 are cleared two per memory cycle by incrementing plus 2 until 00096 and 00097 have been set to zero and 00098 written into PR-1. The cycle control trigger is turned ON during the cycle in which 00096 is in MAR and 00098 is written back into PR-1. On the tenth memory cycle, with the cycle control trigger ON, 00098 is read into MAR from PR-1 and memory locations 00098 and 00099 are set to zero. The MAR address (00098) is written back into PR-1 incremented plus one. PR-1 then contains 00099, which designates the location of the units position of the product area in memory.

Trigger 33. (01.62.33.1).

1. Read out of memory per OR-1 (multiplier) and store the digit in the M/Q register.
2. Write back OR-1 decremented.
3. Turn OFF high/plus trigger during first cycle if Q field (multiplier) is negative.
4. Set field mark No. 1 trigger after first cycle when end of Q field (OR-1) is reached.

Trigger 34. (01.62.34.1).

For units position of multiplicand:

1. Read out of memory per OR-2 and store the digit in MDR.
 2. Write back OR-2 bypassed.
 3. Write from MAR into OR-3 decremented.
- For other than units position of multiplicand:
4. Read out of memory per OR-3 and store the digit in MDR.
 5. Write back OR-3 decremented.

On first multiply cycle:

6. Change status of high/plus trigger if P field (multiplicand) is negative.

After first multiply cycle:

7. Set field mark No. 2 trigger when end of multiplicand field (OR-3) is reached.

Trigger 35. (01.62.35.1).

1. Develop and set multiply table address in MAR (see Figure 10-19.)
2. Read out of memory per multiply table address and store in the digit/branch register the two digits that are associated with that address.

Trigger 36. (01.62.36.1).

For units position of multiplicand:

1. Read out of memory per PR-1 (product area) and store the digit in MDR.
2. Write back PR-1 decremented and write PR-1 bypassed into PR-2.

For other than units position of multiplicand:

3. Read out of memory per PR-2 (product area) and store the digit in MDR.
4. Write back PR-2 bypassed.

Trigger 37. (01.62.37.1).

1. Develop and set add table address in MAR, transferring the digit in the units position of the digit/branch register into the units position of MAR and the digit in MDR into the tens position of MAR.
2. Read out of memory per add table address and store the sum digit in MDR.
3. Transfer the digit in the tens position of the digit/branch register into the units position of the digit/branch register.
4. Turn ON carry out trigger if MDR contains an F bit (add table carry).
5. Turn ON carry out trigger if the output of the τ/c switch is a "9" and a carry exists (CI trigger on) from the previous add cycle.
6. Turn OFF equal/zero trigger if add table digit is not a zero.

Trigger 38. (01.62.38.1).

1. Read out of memory per PR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the PR-2 address is odd or even, to clear the memory location.
2. Write back PR-2 decremented into PR-2 and PR-3.
3. Turn ON carry in trigger if carry out trigger is ON.
4. Set an F bit in MDR if the high/plus trigger is OFF (minus sign) on first multiply cycle.
5. Clear the F bit (carry) after first multiply cycle.
6. Write into memory per PR-2 from MBR.

Trigger 39. (01.62.39.1).

1. Read out of memory per PR-3 (product area) and store the digit in MDR.
2. Write back PR-3 bypassed.

Trigger 40. (01.62.40.1).

1. Develop and set add table address in MAR, transferring the digit in the units position of the digit/branch register into the units position of MAR and the digit in MDR into the tens position of MAR.
2. Read out of memory per add table address and store the sum digit in MDR.
3. Turn ON carry out trigger if MDR contains an F bit (carry).
4. Turn ON carry out trigger if the output of the τ/c switch is a "9" and a carry exists (CI trigger on) from the previous add cycle.
5. Turn OFF equal/zero trigger if the add table digit is not a zero.

Trigger 41. (01.62.41.1).

1. Read out of memory per PR-3 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the PR-3 address is odd or even, to clear the memory location.
2. Write back PR-3 decremented.
3. Turn ON carry in trigger if carry out trigger is ON.
4. Clear the F bit (carry) after first multiply cycle unless an F bit is being set for field definition.
5. Set an F bit in MDR for field definition when field mark No. 1 and field mark No. 2 have been reached and there is no carry out.
6. Write into memory per PR-3 from MBR.
7. Turn ON trigger 34 if there is not a carry out and field mark No. 2 has not been reached.
8. Turn ON trigger 19 if there is a carry out or field mark No. 2 has been reached.

Trigger 19. (01.60.49.1).

1. Turn ON trigger 33 if field mark No. 1 has not been reached, and the carry out trigger is OFF.
2. Turn ON trigger 39 if the carry out trigger is ON.
3. Enter I-cycle for the next instruction if the carry out trigger is OFF and field mark No. 1 has been reached.

Multiply Immediate (Code 13 — MM)

The sequence block diagram for this operation is shown in Figure 10-16. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 15, 17, and 19.

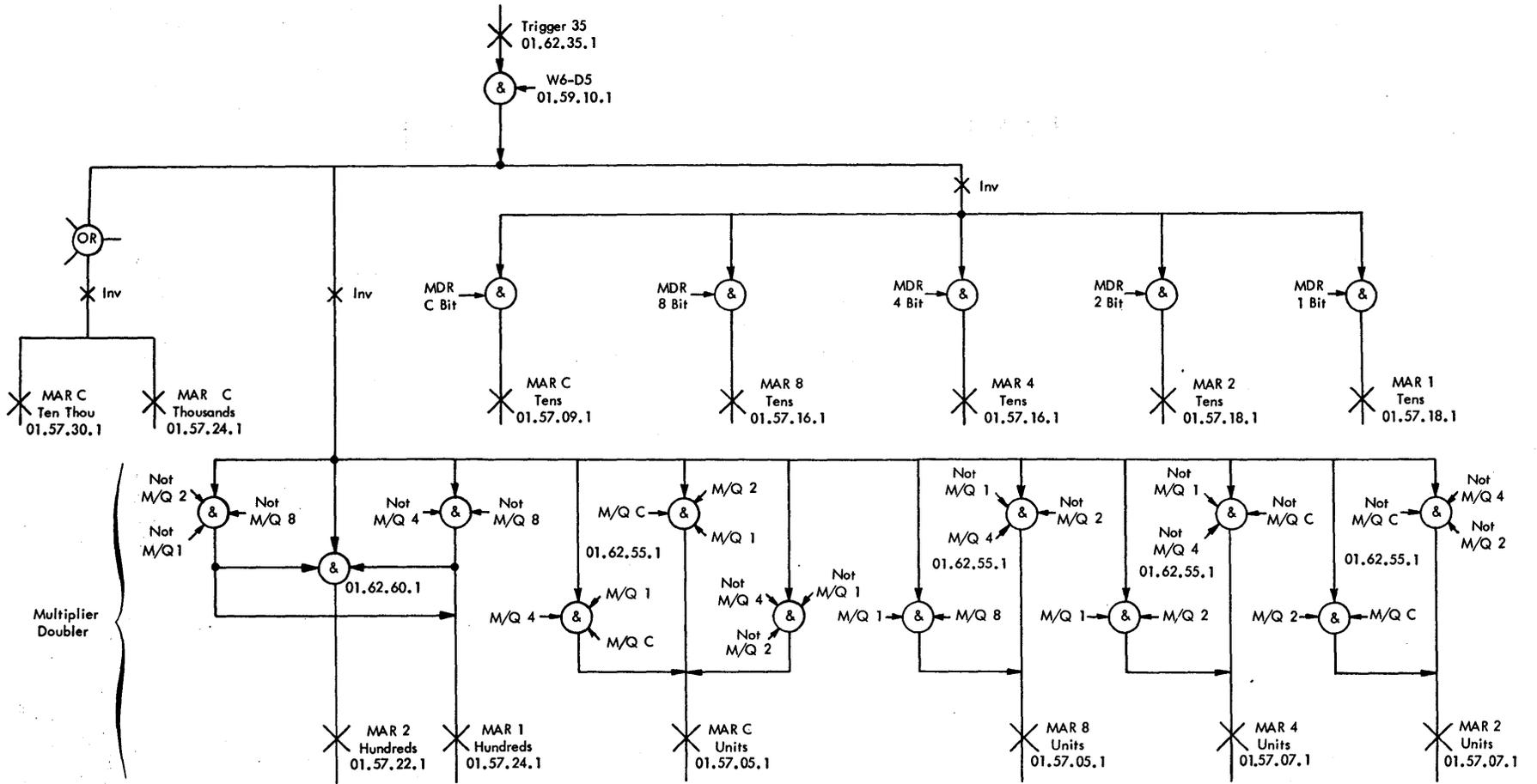


Figure 10-19. Set MAR Multiply Table Address

Objective

Multiply the field at the P address (OR-2) by the field whose units position is the Q₁₁ digit of the multiply immediate instruction, and store the algebraic product at location 00099 and successively lower memory locations.

Functions

The multiply immediate operation differs from the Multiply - Code 23 operation only in the method of setting the multiplier address in OR-1 during the I-cycle. See Multiply - Code 23.

Auxiliary Triggers

See Multiply - Code 23.

E-Timer Triggers

See Multiply - Code 23.

Load Dividend (Code 28 — LD)

The sequence block diagram for this operation is shown in Figure 10-20. The function chart for this operation is shown in the *Instructional System Diagrams*, page 20.

Objectives

1. Transfer the dividend field, including the F bit for field definition, at the Q address (OR-1) to the memory location in the product area designated by the P address (OR-2) and successively lower memory locations.
2. Set an F bit at memory location 00099 if the dividend is negative.

Functions

The address of the dividend field is specified by OR-1. The memory location within the product area into which the units digit of the dividend field is to be placed is specified by OR-2.

The 20 positions in memory designated as product area (00080 - 00099) are cleared to zeros before transmission of the dividend field begins.

Transmission of the Q field (dividend) to the product area proceeds serially from right to left until terminated by the flag bit which marks the high-order position of the Q field. This high-order position flag bit for field definition is duplicated in the product area field. The flag bit which marks the units position of a negative dividend field is cleared before the units digit is placed in memory at the P address.

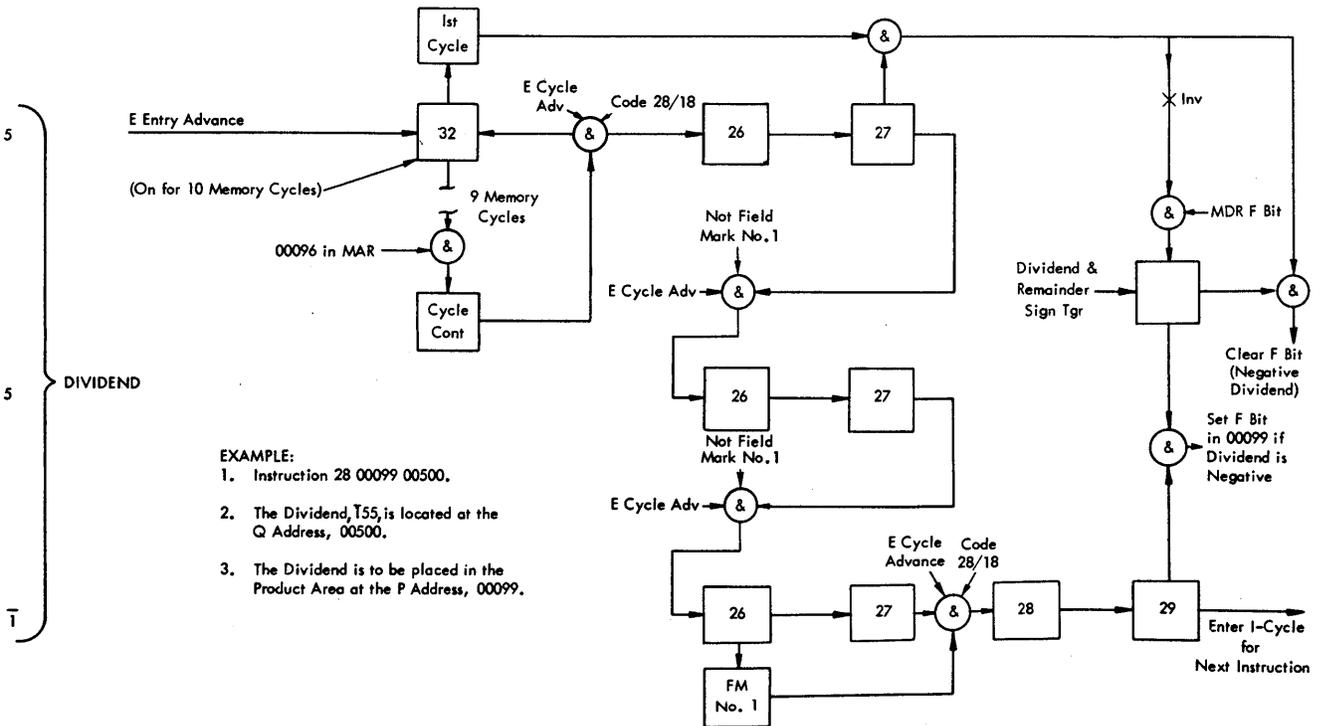


Figure 10-20. Load Dividend—Code 28/18

After transmission of the dividend into the product area is complete, a flag bit is placed in memory location 00099 if the dividend field is negative.

Auxiliary Trigger Status

Decrement Trigger. (01.60.05.1).

1. Turned OFF (increment) by I-cycle entry.
2. Turned ON (decrement) by trigger 26.
3. Remains ON for remainder of load dividend operation.

First Cycle Trigger. (01.63.10.1).

1. Turned on by trigger 32.
2. Turned OFF during trigger 27 time on first transmit cycle.

Cycle Control Trigger. (01.62.31.1).

1. Turned ON during trigger 32 time when the address in MAR reaches 00096 to terminate the product area clearing operation.
2. Turned OFF by trigger 27.

Field Mark No. 1 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 26 time on the transmit cycle in which end of Q field (OR-1) is reached.

Dividend and Remainder Sign Trigger. (01.65.06.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON by trigger 27 during the first transmit cycle if the Q field (dividend) is negative.

E-Timer Trigger Objectives

Trigger 32. (01.62.30.1).

1. Turn ON first cycle trigger.
2. Turn ON high/plus trigger.
3. Set 00080 in MAR on first memory cycle.
4. Read out of memory per 00080, with both the even and odd sense amplifiers blocked (Block Memory SA), to clear product area memory locations 00080 and 00081.
5. Write into PR-1 the MAR address (00080) incremented plus 2.
6. Set C bits into MBR-even and MBR-odd to provide zeros for the product area.
7. Write into memory per MAR address (00080) from MBR.

Trigger 32 remains ON for a total of ten memory cycles during which time the 20 memory locations designated as product area (00080 - 00099) are set to zero. Memory locations per PR-1 are cleared two per memory cycle by incrementing plus 2 until 00096

and 00097 have been set to zero and 00098 written into PR-1. The cycle control trigger is turned ON during the cycle in which 00096 is in MAR and 00098 is written back into PR-1. On the tenth memory cycle, with the cycle control trigger ON, 00098 is read into MAR from PR-1 and memory locations 00098 and 00099 are set to zero. The MAR address (00098) is written back into PR-1 incremented plus one. PR-1 then contains 00099, which designates the location of the units position of the product area in memory.

Trigger 26. (01.60.56.1).

1. Turn ON decrement trigger (decrement).
2. Read out of memory per OR-1 and store the digit in MDR.
3. Write back OR-1 decremented.
4. Turn ON field mark No. 1 trigger when the high-order digit of the Q field is read out of memory.

Trigger 27. (01.60.57.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Write back OR-2 decremented.
4. Transfer MDR to MBR.
5. Write into memory per OR-2 from MBR.
6. Turn the dividend and remainder sign trigger on if the Q field (dividend) is negative.
7. Clear the F bit from the units digit of a negative dividend before placing the digit in memory at the P address.

(Repeat trigger 26 objectives 2 and 3, followed by trigger 27 objectives 1, 2, 3, 4, and 5 for each digit of the field.)

8. Turn ON trigger 28 when all digits of the dividend have been transferred.

Trigger 28. (01.60.58.1).

1. Read out of memory per PR-1 and store the digit in MDR. (PR-1 contains 00099, the address of the units position of the product area, as a result of trigger 32 functions.)
2. Write from MAR into OR-2 bypassed.

Trigger 29. (01.60.59.1).

1. Block reset of MAR.
2. Block reset of MDR (Read Y).
3. Read out of memory per MAR (OR-2) with either the odd or even sense amplifiers blocked (Read Y) depending on whether the MAR (OR-2) address is odd or even, to clear the memory location.
4. Transfer MDR to MBR.

5. Set an F bit in MDR if the dividend and remainder sign trigger is ON.
6. Write into memory per MAR (OR-2) from MBR.
7. End operation and enter I-cycle for the next instruction in sequence.

Load Dividend Immediate (Code 18 — LDM)

The sequence block diagram for this operation is shown in Figure 10-20. The function chart for this operation is shown in the *Instructional System Diagrams*, page 20.

Objectives

1. Transfer the field (dividend) whose units position is the Q_{11} digit of the load dividend immediate instruction to the memory location in the product area designated by the P address (OR-2) and successively lower memory locations.
2. Set an F bit at memory location 00099 if the dividend is negative.

Functions

The load dividend immediate operation differs from the Load Dividend - Code 28 operation only in the method of setting the dividend address in OR-1 during the I-cycle. See Load Dividend - Code 28.

Auxiliary Triggers

See Load Dividend - Code 28.

E-Timer Triggers

See Load Dividend - Code 28.

Divide (Code 29 — D)

The sequence block diagram for this operation is shown in Figures 10-21, 10-22, 10-23, and 10-24. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 23 and 25.

Objectives

1. Divide the field which was placed in the product area during the load dividend operation by the field at the Q address (OR-1).
2. Store the quotient and the remainder with their respective signs in the product area to replace the dividend.

Functions

The location of the divisor is specified by OR-1. The location of the dividend within the product area is de-

termined by the preceding load dividend operation. Minimum field length for either the divisor or dividend is two digits.

The P address specifies the location at which the first subtraction of the divisor is to occur. This P address must be such that the first quotient digit developed is 9 or less. If this condition is not met, or divide by zero is attempted, the overflow indicator is turned ON, the divide operation is terminated, and the computer enters I-cycle for the next instruction in sequence. Subsequent quotient digits are developed by successively subtracting the divisor from the dividend one memory location to the right of the previous successive subtraction.

The first quotient digit is stored in the product area as many positions to the left of the P address as there are digits in the divisor. Each subsequent quotient digit is stored to the right of the last quotient digit. The quotient field is defined by storing a flag bit with the first quotient digit developed.

The low-order digit of the quotient is located as many positions to the left of 00099 as there are digits in the divisor. A flag bit is stored with the low-order digit if the quotient is negative.

The low-order digit of the remainder is located at 00099 and is given the sign of the dividend. The remainder field is equal in length to the divisor and is defined by storing a flag bit with the high-order digit.

The divide operation is terminated when a quotient digit is stored as a result of the divide cycle in which the units position of the divisor is subtracted from location 00099.

See chart of trigger objectives and sequence, Figure 10-25.

Resulting Indicator Conditions

1. The high/plus indicator is ON if the quotient is positive and is OFF if the quotient is negative or zero.
2. The equal/zero indicator is ON if the quotient is zero and is OFF if the quotient is not zero.
3. The overflow indicator is ON if the first quotient digit developed was greater than 9 or if divide by zero was attempted. If the overflow indicator is ON as a result of a previous arithmetic operation, a no overflow condition of this divide instruction will not turn it OFF.

NOTE: Once the high/plus or equal/zero indicators are turned ON or OFF by an arithmetic or compare operation, they will retain that state until the next arithmetic or compare operation is executed (unless the reset key on the console is depressed which turns these indicators OFF). Testing them with a branch indicator

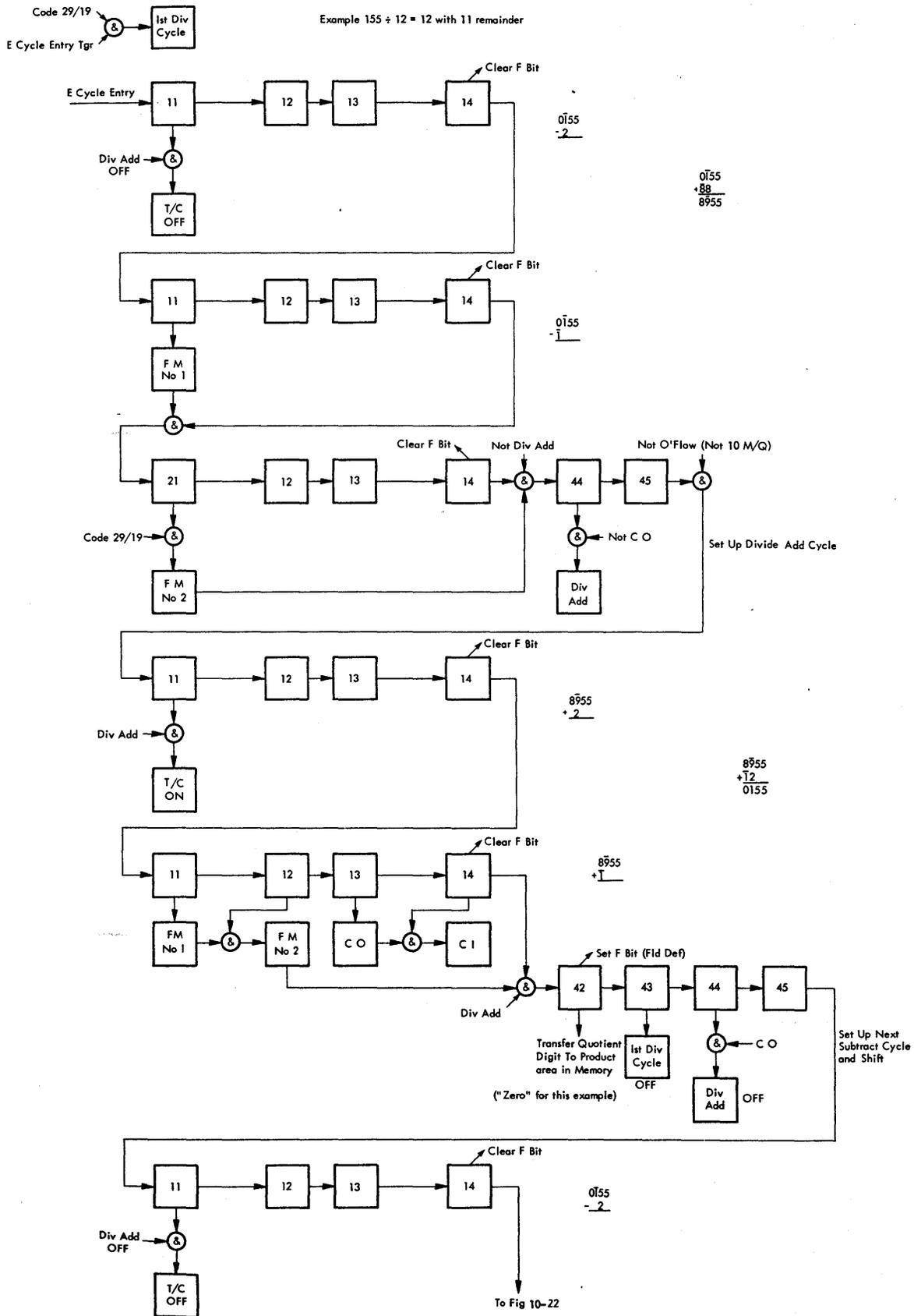


Figure 10-21. Divide—Code 29/19

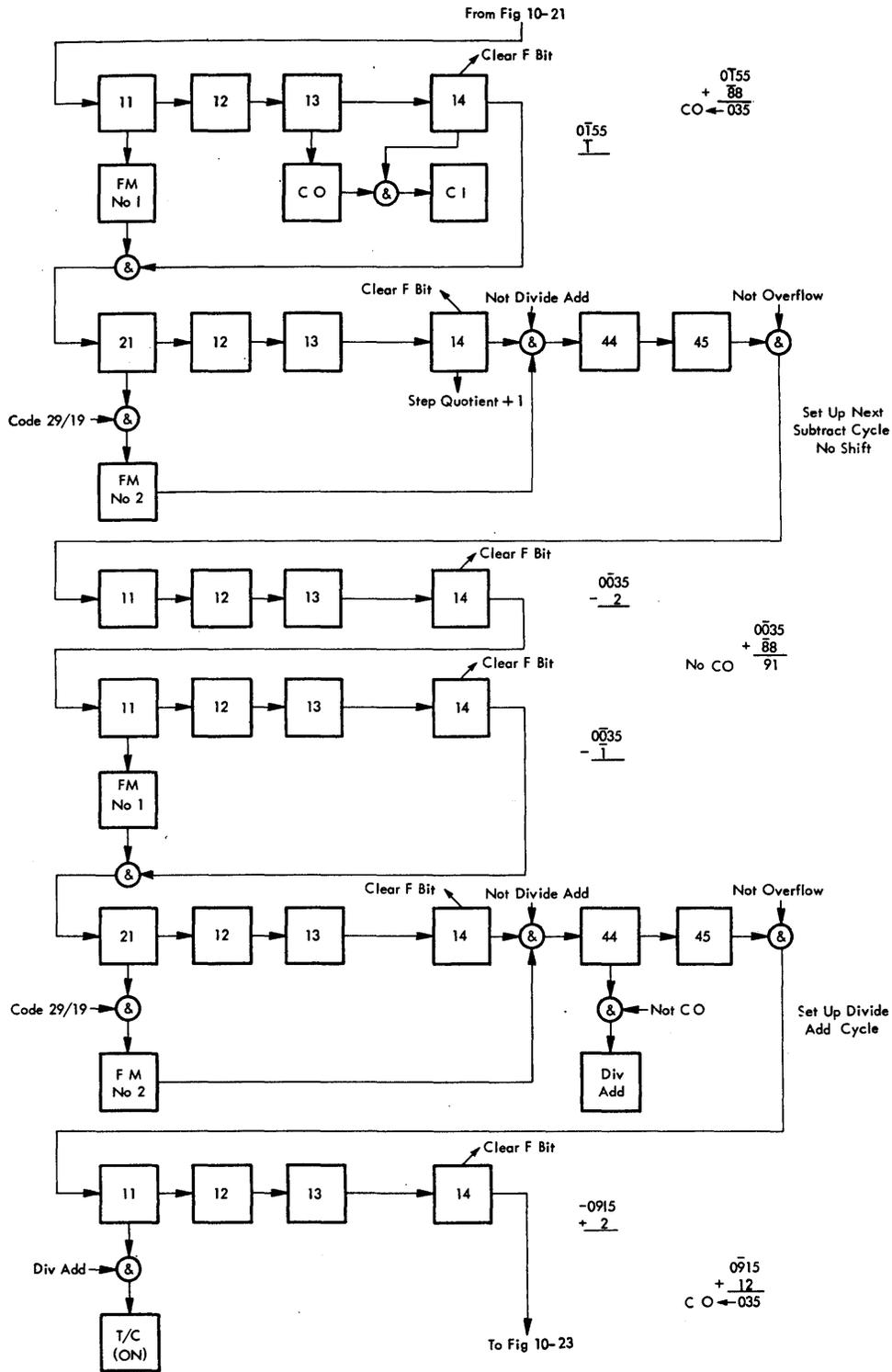


Figure 10-22. Divide-Code 29/19

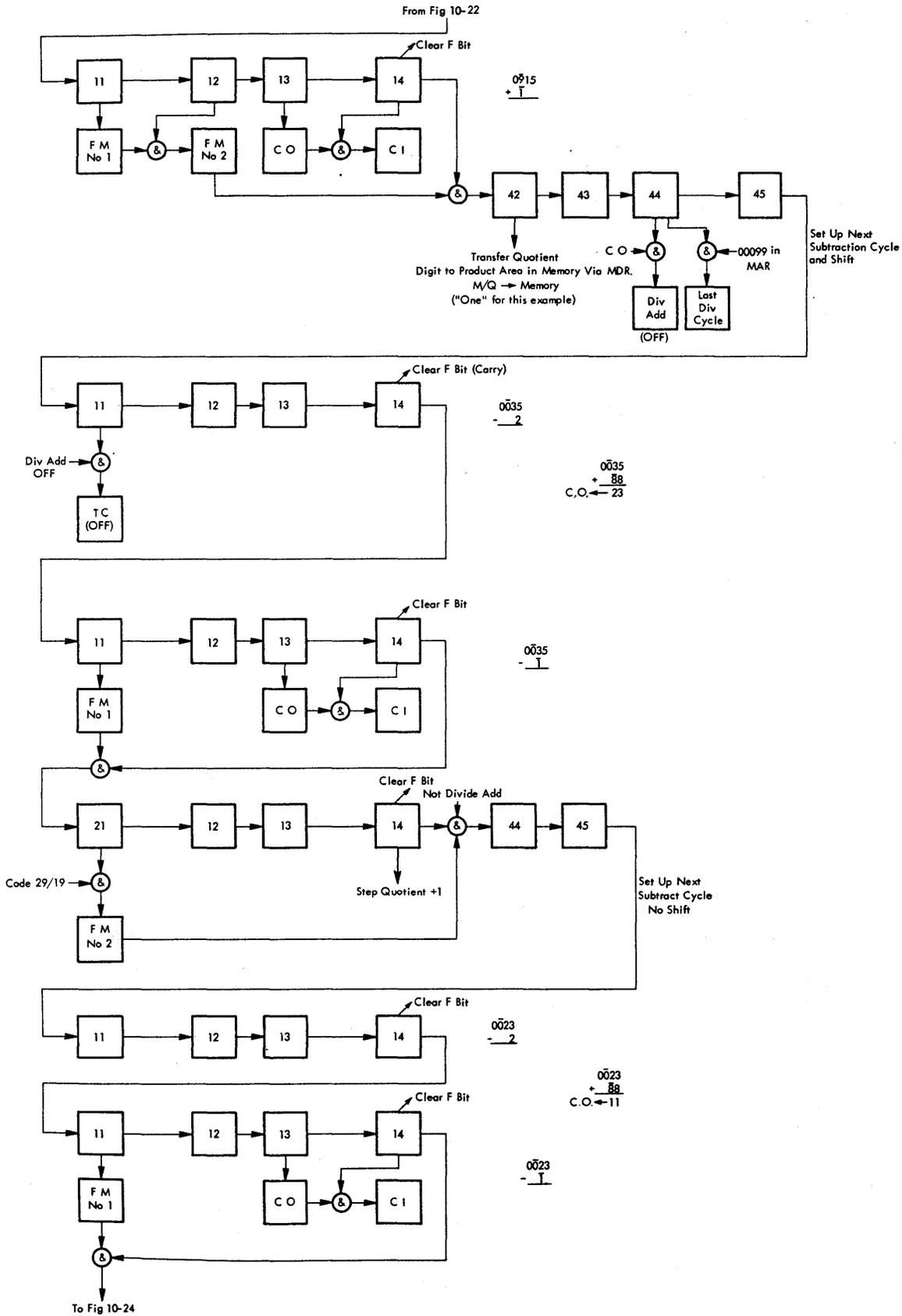


Figure 10-23. Divide—Code 29/19

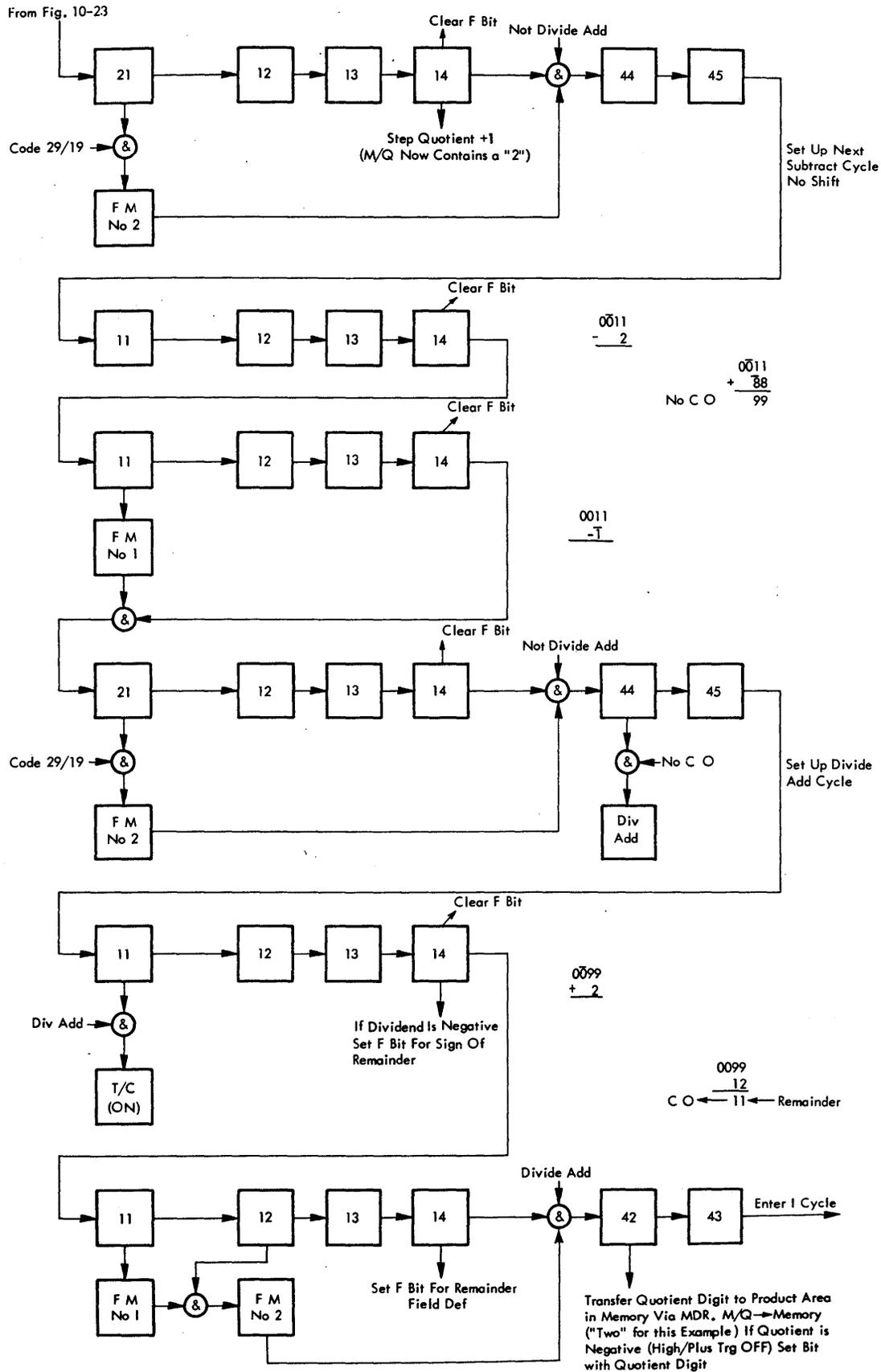


Figure 10-24. Divide—Code 29/19

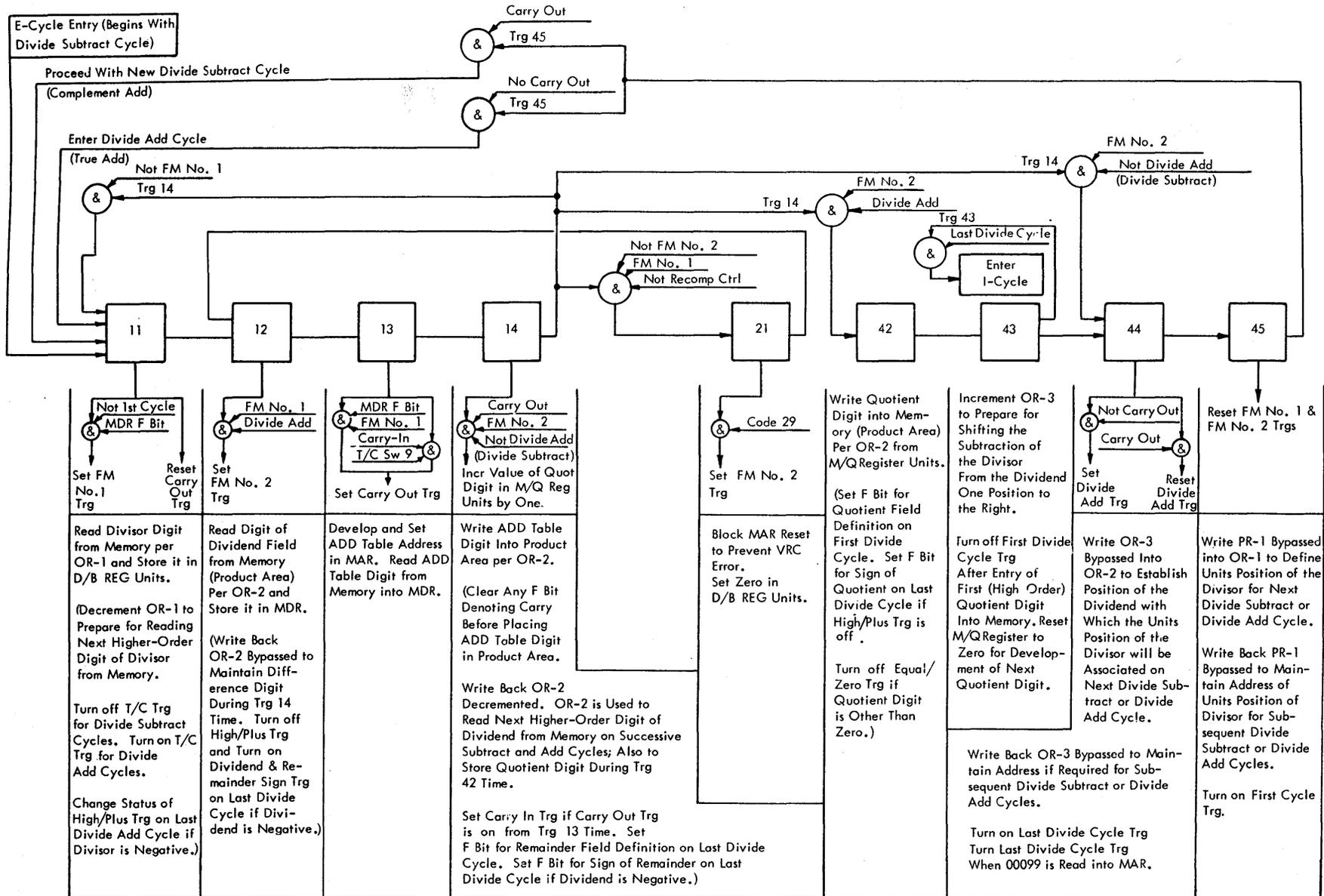


Figure 10-25. Divide-Code 29/19-Flow Diagram

or branch no indicator instruction has no effect on their state. The overflow indicator will be turned OFF only by testing it with a branch indicator or branch no indicator instruction (or by depressing the reset key on the console).

Auxiliary Trigger Status

Divide Add Trigger. (01.65.04.1).

1. Turned OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered. With the divide add trigger OFF, complement addition (subtraction) is performed. One subtraction of the entire divisor from the dividend is termed a "divide subtract cycle". Subtraction of one digit of the divisor from the dividend is termed a "subtract cycle".
2. Turned ON by trigger 44 after a no carry out on a divide subtract cycle. (No carry out occurs on a divide subtract cycle when the portion of the dividend from which the divisor is subtracted is numerically less than the divisor.) With the divide add trigger ON, true addition is performed. One addition of the entire divisor to the dividend is termed a "divide add cycle". Addition of one digit of the divisor to the dividend is termed an "add cycle".

First Divide Cycle Trigger. (01.65.02.1).

1. Turned ON by the E-cycle entry trigger during trigger 8 time of the I-cycle.
2. Turned OFF by trigger 43 after entry of first (high-order) quotient digit into memory. A "divide cycle" consists of one or more divide subtract cycles and one divide add cycle.) See Chart of Divide Operation, Figure 10-26.

Decrement Trigger. (01.60.05.1).

1. Turned ON (decrement) during trigger 8 time of the preceding I-cycle.
2. Turned OFF (increment) by trigger 43 to shift subtraction of the divisor one position to the right. (Remains OFF only during trigger 43 time.)
3. Turned ON by trigger 44.

First Cycle Trigger. (01.63.10.1).

For first divide subtract cycle:

1. Turned ON when E-cycle is entered.
2. Turned OFF at the end of trigger 14 time of the first subtract cycle.

For subsequent divide subtract cycles:

3. Turned ON by trigger 45.
4. Turned OFF at the end of trigger 14 time of the first subtract cycle.

For divide add cycles:

5. Turned ON by trigger 45.
6. Turned OFF at the end of trigger 14 time of the first add cycle.

High/Plus Trigger. (01.60.40.1).

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 12 time on the first subtract cycle of the last divide cycle if the P field (dividend) is negative.
3. Status changed during trigger 11 time on the first add cycle of the last divide cycle if the Q field (divisor) is negative.

Equal/Zero Trigger. (01.60.41.1).

1. Turned ON when E-cycle is entered.
2. Turned OFF during trigger 42 time if the quotient is not zero.

True/Complement (T/C) Trigger. (01.63.20.1).

1. Turned ON by trigger 1 of the I-cycle and therefore will be ON when E-cycle is entered.

Divide subtract cycles:

2. Turned OFF by trigger 11.

Divide add cycles:

3. Turned ON by trigger 11.

Field Mark No. 1 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 11 time of each divide subtract and divide add cycle when end of Q field (divisor) is reached.
3. Reset OFF by trigger 45 between each divide subtract and divide add cycle.

Field Mark No. 2 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Reset OFF by trigger 45 between each divide subtract and divide add cycle.

Divide subtract cycles:

3. Turned ON by trigger 21.

Divide add cycles:

4. Turned ON by trigger 12 on the add cycle in which field mark No. 1 is reached.

Carry Out Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore will be off when E-cycle is entered.
2. Reset OFF by triggers 11 and 21.
3. Turned ON during trigger 13 time on add and subtract cycles where the output of the T/C switch is a "9" and the carry in trigger is ON.
4. Turned ON during trigger 13 time if the add table digit contains an F bit (carry).

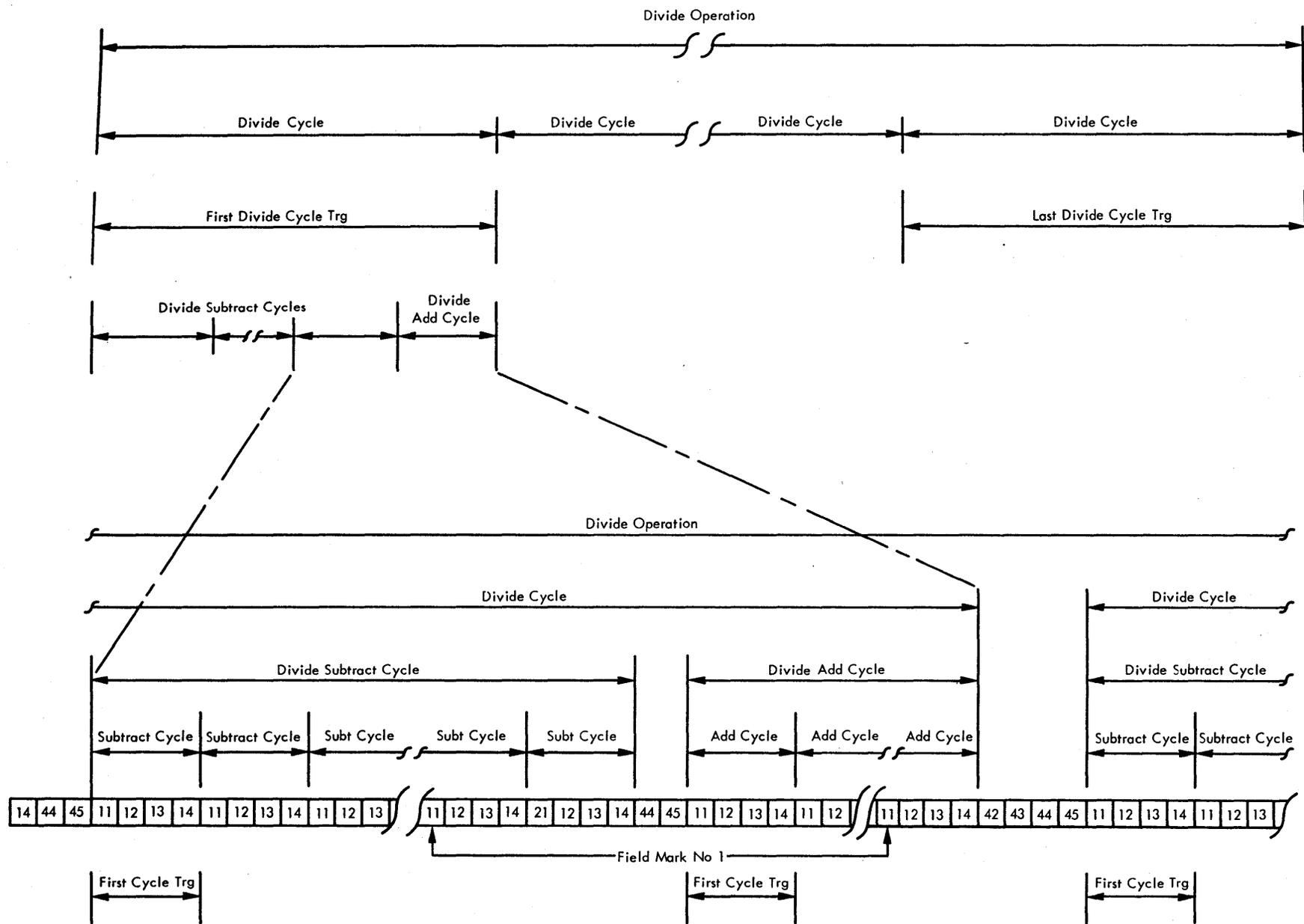


Figure 10-26. Divide Operation

Carry In Trigger. (01.63.40.1).

1. Reset OFF by the I-cycle trigger and therefore is off when E-cycle is entered.
2. Turned ON during trigger 12 time of the first subtract cycle of each divide subtract cycle to obtain 10's complement in units position.
3. Reset OFF during trigger 13 time.
4. Turned ON by trigger 14 if the carry out trigger is ON.

Last Divide Cycle Trigger. (01.65.02.1).

1. Turned OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 44 time when 00099 is read from OR-3 into MAR.

Dividend and Remainder Sign Trigger. (01.65.06.1).

1. Turned OFF by I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 12 time of the first subtract cycle in the last divide cycle if the dividend is negative. (The sign of the dividend was stored in memory location 00099 during the load dividend operation.)

Quotient Counter Latches. (01.65.07.1).

1. Reset OFF by trigger 12 for use during trigger 13 time.
2. Turned ON by trigger 13, depending upon the content of the M/Q register, to set up logical paths for increasing by one the quotient digit stored in the M/Q register. The increase by one is accomplished during trigger 14 time.

E-Timer Trigger Objectives

Trigger 11. (01.60.11.1).

1. Read out of memory per OR-1 and store the digit in D/B REG units.
2. Write back OR-1 decremented.

Divide subtract cycles:

3. Turn OFF T/C trigger.
4. Turn ON field mark No. 1 trigger after the first subtract cycle of each divide subtract cycle when end of Q field (divisor) is reached.

Divide add cycles:

5. Turn ON T/C trigger.
6. Turn ON field mark No. 1 trigger after the first add cycle of each divide add cycle when end of Q field (divisor) is reached.
7. Change status of high/plus trigger during the first add cycle of the last divide cycle if the Q field (divisor) is negative.

Trigger 12. (01.60.12.1).

1. Read out of memory per OR-2 and store the digit in MDR.

2. Write back OR-2 bypassed.

Divide subtract cycles:

3. Turn ON the carry in trigger during the first subtract cycle of each divide subtract cycle to obtain 10's complement in units position.

Divide add cycles:

4. Turn ON field mark No. 2 trigger if field mark No. 1 trigger is ON.
5. Turn OFF high/plus trigger during the first subtract cycle of the last divide cycle if the dividend is negative. (Sign of the dividend was stored in 00099 during the load dividend operation.)
6. Turn ON dividend and remainder sign trigger during the first subtract cycle of the last divide cycle, if the dividend is negative, for use in setting sign of the remainder.
7. Reset off the quotient counter latches.

Trigger 13. (01.60.13.1).

1. Develop and set add table address in MAR.
2. Read out of memory per add table address and store the difference or sum digit in MDR.
3. Turn ON the carry out trigger if MDR contains an F bit (carry).
4. Reset D/B REG units. (See Trigger 21 Objective 3.)
5. Turn on the quotient counter latches depending upon the content of the M/Q register.

Trigger 14. (01.60.14.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even to clear the memory location.
3. Write back OR-2 decremented.
4. Turn ON carry in trigger if carry out trigger is ON.
5. Clear the F bit if the add table digit contains a carry.
6. Set an F bit in MDR for sign of the remainder during the first add cycle of the last divide cycle if the dividend and remainder sign trigger is ON. (See Trigger 12 Objective 6.)
7. Set an F bit in MBR for field definition of the remainder during the add cycle of the last divide cycle when field mark No. 2 trigger is ON. (See Trigger 12 Objective 4.)
8. Transfer MDR to MBR.
9. Write into memory per OR-2 from MBR.
10. Turn trigger 11 ON if field mark No. 1 has not been reached to repeat the four memory cycles (triggers 11, 12, 13, and 14) for the next higher order digits.
11. Increase the digit value of the M/Q register by one (step quotient) for each divide subtract cycle in which a carry out occurs. (A carry out results

when the portion of the dividend from which the divisor is subtracted is numerically larger than the divisor.) See Figure 10-27.

12. Turn ON trigger 21 during the divide subtract cycle in which field mark No. 1 is reached.
13. Turn ON trigger 44 at the end of the subtract cycle in which trigger 21 turns ON field mark No. 2 trigger.

Divide add cycles:

14. Turn ON trigger 42 during the divide add cycle in which field mark No. 1 is reached. (See Trigger 12 Objective 4.)

Trigger 21. (01.60.31.1).

Replaces trigger 11 for first memory cycle after field mark No. 1 has been reached during divide subtract cycles.

1. Block MAR reset to prevent VRC error.
2. Turn ON field mark No. 2 trigger.
3. Set zero in D/B REG units. This will provide a zero in place of the Q field (divisor) digits previously set into D/B REG units by trigger 11. (The D/B REG units was reset during trigger 13 time.)
4. Turn ON trigger 12.

Trigger 42. (01.65.05.1).

1. Read out of memory per OR-2 with either the odd or the even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
2. Allow the reset of MDR.
3. Transfer quotient digit from the M/Q register to MDR.
4. Set an F bit in MDR for field definition of the quotient on first divide cycle.
5. Set an F bit in MDR on the last divide cycle if the high/plus trigger is OFF (minus sign).
6. Transfer MDR to MBR (Read Y).
7. Write into memory per OR-2 from MBR.
8. Turn OFF equal/zero trigger if the quotient digit is not zero.
9. Turn ON trigger 43.

Trigger 43. (01.65.05.1).

1. Turn OFF decrement trigger (increment).
2. Read OR-3 into MAR.
3. Write from MAR into OR-3 incremented.
4. Turn OFF first divide cycle trigger after entry of first (high-order) quotient digit into memory.
5. Reset the M/Q register to zero.
6. Turn ON trigger 44 if the last divide cycle trigger is OFF.
7. End operation and enter I-cycle for the next instruction in sequence if the last divide cycle trigger is ON.

Trigger 44. (01.65.03.1).

1. Read OR-3 into MAR.
2. Write from MAR into OR-2 and OR-3 bypassed.
3. Turn ON last divide cycle trigger when 00099 is read into MAR.
4. Turn ON decrement trigger (decrement).
5. Turn ON divide add trigger if the carry out trigger is OFF.
6. Turn OFF divide add trigger if the carry out trigger is ON.
7. Turn ON trigger 45.

Trigger 45. (01.65.03.1).

1. Read PR-1 into MAR.
2. Write from MAR into PR-1 and OR-1 bypassed.
3. Turn ON first cycle trigger.
4. Turn ON overflow trigger and enter I-cycle for the next instruction in sequence if a quotient digit of ten was developed. (C, 8, and 2 bits in M/Q register.)
5. Turn ON trigger 11 if an overflow condition was not developed (C, 8, and 2 bits in M/Q register) during the previous trigger 14 time.

Divide Immediate (Code 19 — DM)

The sequence block diagram for this operation is shown in Figures 10-21, 10-22, 10-23, and 10-24. The function chart for this operation is shown in the *Instructional System Diagrams*, pages 23 and 25.

Objectives

1. Divide the field which was placed in the product area during the load dividend operation by the field whose units position is the Q_{11} digit of the divide immediate instruction.
2. Store the quotient and the remainder with their respective signs in the product area, replacing the dividend.

Functions

The divide immediate operation differs from the Divide - Code 29 operation only in the method of setting the divisor address in OR-1 during the I-cycle. See Divide - Code 29.

Auxiliary Triggers

See Divide - Code 29.

E-Timer Triggers

See Divide - Code 29.

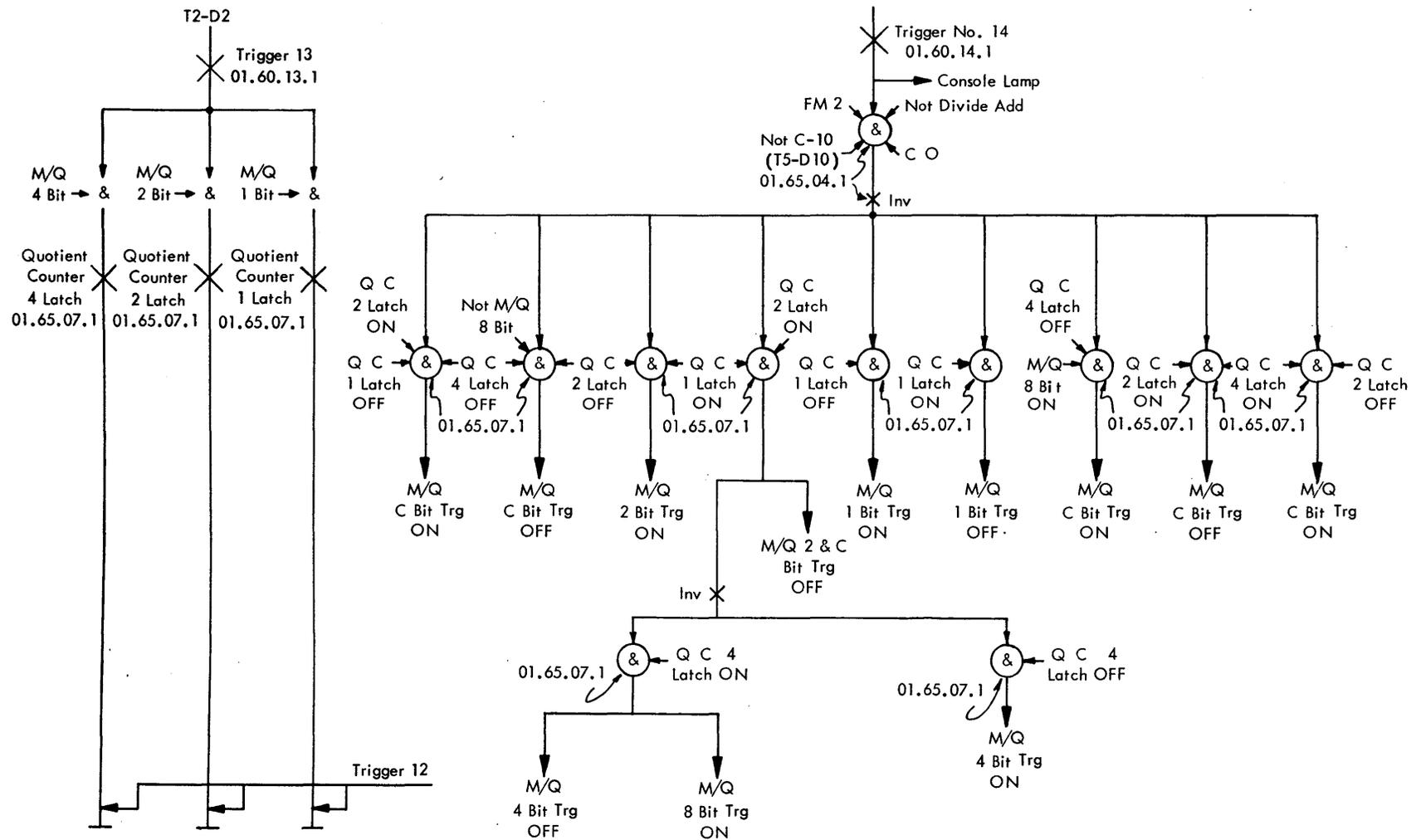


Figure 10-27. Quotient Counter

Section 11 Branch Operations

Branch instructions are used in a program to permit alteration of the sequential execution of the program. These instructions may be classified into two categories as follows:

1. Unconditional branch instructions:
 - a. Branch
 - b. Branch and transmit
 - c. Branch back
2. Conditional branch instructions:
 - a. Branch on digit
 - b. Branch no flag
 - c. Branch no record mark
 - d. Branch indicator
 - e. Branch no indicator

Unconditional branch instructions always alter the sequential execution of a program when they are executed. Whether branching occurs as a result of conditional branch instructions is dependent upon the status of a sense switch on the console or an indicator within the computer.

The P address in every branch instruction must be even because it designates the address of another instruction.

Function charts for 1620 operation codes are located in the *IBM Customer Engineering Instructional System Diagrams, 1620 Data Processing System* (Form No. 227-5631).

Branch (Code 49 — B)

The sequence block diagram for this operation is shown in Figure 11-1. The function chart for this operation is shown in the *Instructional System Diagrams*, page 26.

Objective

Branch (unconditionally) to the instruction at the P address (OR-2).

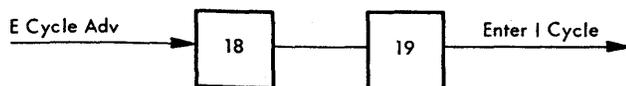


Figure 11-1. Branch—Code 49

Functions

The content of OR-2 is placed in IR-1. The address of the next instruction to be executed is then specified by IR-1.

The Q address of this instruction is placed in OR-1 but is not used in the execution of this operation.

Auxiliary Triggers

None are used.

E-Timer Trigger Objectives

Trigger 18. (01.60.48.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).

(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch instruction (now stored in IR-1).

Branch and Transmit (Code 27 — BT)

The sequence block diagram for this operation is shown in Figure 11-2. The function chart for this operation is shown in the *Instructional System Diagrams*, page 27.

Objective

1. Save the address of the next instruction in sequence. Execution of the next branch back instruction directs the computer to resume the program beginning with the instruction at the saved address.
2. Transmit the field at the Q address (OR-1) to the memory location designated by the P address minus one and successively lower memory locations.
3. Branch to the instruction at the P address.

Functions

The address of the next instruction in sequence is saved by storing the IR-1 address in IR-2.

The P address (OR-2) is transferred into IR-1 so that when the computer enters the next I-cycle it will execute the instruction at the P address.

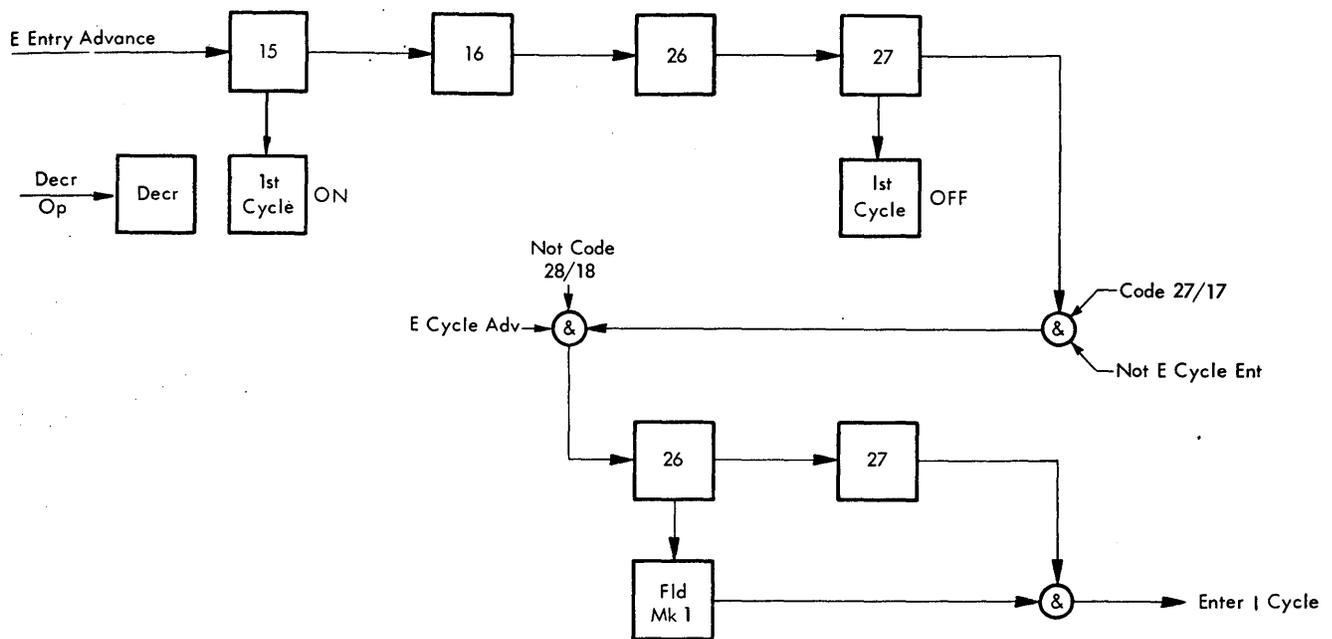


Figure 11-2. Branch and Transmit—Code 27/17

Transmission of the Q field proceeds serially, one digit at a time, from low-order to high-order digit of the transmitted field until the operation is terminated by a flag bit (field mark No. 1) in the high-order position of the transmitted field. The flag bits in the high-order and units position of the transmitted field are duplicated in the field at P minus one. The digits in the field at P minus one prior to transmission are obliterated, including their flag bits, if any. When transmission is completed, enter I-cycle for the instruction at the P address now stored in IR-1.

Auxiliary Trigger Status

First Cycle Trigger. (01.63.10.1).

1. Turned ON by trigger 15.
2. Turned OFF by trigger 27 on first transmit cycle.

Decrement Trigger. (01.60.05.1).

1. Turned ON (decrement) during trigger 8 time of the preceding I-cycle.
2. Remains ON until the next I-cycle is entered.

Field Mark No. 1 Trigger. (01.63.30.1).

1. Reset OFF by the I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Cannot be turned ON during first transmit cycle. (Q Field must be a minimum of two digits.)
3. Turned ON during trigger 26 time of the transmit cycle in which the high-order digit of the Q field is read out of memory.

E-Timer Trigger Objectives

Trigger 15. (01.60.45.1).

1. Read IR-1 into MAR.
2. Write from MAR into IR-2 bypassed.

Trigger 16. (01.60.45.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.
3. Write from MAR into OR-2 decremented (P address minus one).

Trigger 26. (01.60.56.1).

1. Read out of memory per OR-1 and store the digit in MDR.
2. Write back OR-1 decremented.

Trigger 27. (01.60.57.1).

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Write back OR-2 decremented.
4. Transfer MDR to MBR.
5. Write into memory per OR-2 from MBR. (Repeat trigger 26 and trigger 27 Objectives 1, 2, 3, 4, and 5 for each digit of the field.)
6. End operation when all digits of the field have been transferred and enter I-cycle for the instruction at the P address of the branch and transmit instruction (now stored in IR-1).

Branch and Transmit Immediate (Code 17 — BTM)

The sequence block diagram for this operation is shown in Figure 11-2. The function chart for this operation is shown in the *Instructional System Diagrams*, page 27.

Objectives

1. Save the address of the next instruction in sequence for use with the next branch back instruction.
2. Transmit the field whose units position is the Q_{11} digit of the branch and transmit (immediate) instruction to the memory location designated by the P address minus one and successively lower memory locations.
3. Branch to the instruction at the P address.

Functions

The branch and transmit immediate operation differs from the Branch and Transmit — Code 27 operation only in the method of setting the OR-1 address during the I-cycle. See Branch and Transmit — Code 27.

Auxiliary Triggers

See Branch and Transmit — Code 27.

E-Timer Triggers

See Branch and Transmit — Code 27.

Branch Back (Code 42 — BB)

The sequence block diagram for this operation is shown in Figure 11-3. The function chart for this operation is shown in the *Instructional System Diagrams*, page 28.

Objectives

1. Interrogate the status of the save control trigger.
2. If the save control trigger is ON, proceed to the instruction at the address designated by PR-1. (Address was saved by storing it in PR-1 due to a previous depression of the save key.)
3. If the save control trigger is OFF, proceed to the instruction at the address designated by IR-2.

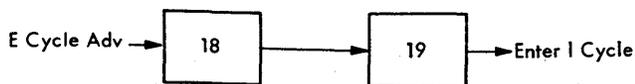


Figure 11-3. Branch Back—Code 42

(Address was saved by storing it in IR-2 when the last branch and transmit or branch and transmit immediate instruction was executed.)

Functions

The address in PR-1 or IR-2, depending upon the status of the save control trigger, is transferred to IR-1.

Auxiliary Trigger Status

Save Control Trigger. (01.06.15.1).

1. Reset OFF by manual reset.
2. Turned ON by depressing the save key on the console.
3. Turned OFF when status is interrogated by a branch back operation.

E-Timer Trigger Objectives

Trigger 18. (01.60.48.1).

1. Read either PR-1 or IR-2 into MAR, depending upon the status of the save control trigger.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).

(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the saved address (now stored in IR-1).

Branch On Digit (Code 43 — BD)

The sequence block diagram for this operation is shown in Figure 11-4. The function chart for this operation is shown in the *Instructional System Diagrams*, page 29.

Objectives

1. Interrogate the single digit located at the Q address (OR-1).
2. If the digit is not a zero, branch to the instruction at the P address (OR-2).
3. If the digit is a zero, proceed to the next instruction in sequence (address in IR-1).

Functions

The digit at the Q address per OR-1 is read out of memory to MDR and is interrogated by the digit trigger. The digit/RM trigger will be turned ON by a digit other than zero.

If the digit is zero (digit/RM trigger OFF), the computer proceeds to the next instruction in sequence.

If the digit is not a zero (digit/RM trigger ON), the address in OR-2 is transferred into IR-1 and I-cycle is entered for the instruction at the P address (now in IR-1).

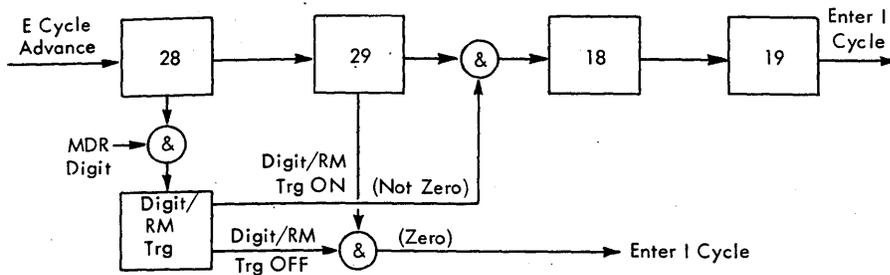


Figure 11-4. Branch On Digit—Code 43

Auxiliary Trigger Status

Digit/RM Trigger. (01.63.30.1)

1. Turned OFF by I-cycle trigger and therefore is off when E-cycle is entered.
2. Turned ON by a digit other than zero in MDR during trigger 28 time.

E-Timer Trigger Objectives

Triggers 28 and 29 are used. Triggers 18 and 19 are also used if the digit is not zero.

Trigger 28. (01.60.58.1).

1. Read out of memory per OR-1 and store the the digit in MDR.
2. Turn ON the digit/RM trigger if the digit in MDR is not a zero.

Trigger 29. (01.60.59.1).

1. Interrogate status of digit/RM trigger. (If ON, proceed to trigger 18. If OFF, end operation and enter I-cycle for the next instruction in sequence.)

Trigger 18. (01.60.48.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).

(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch on digit instruction (now stored in IR-1).

Branch No Flag (Code 44 — BNF)

The sequence block diagram for this operation is shown in Figure 11-5. The function chart for this operation is shown in the *Instructional System Diagrams*, page 30.

Objectives

1. Interrogate the single memory position designated by the Q address (OR-1) for the presence of a flag bit.

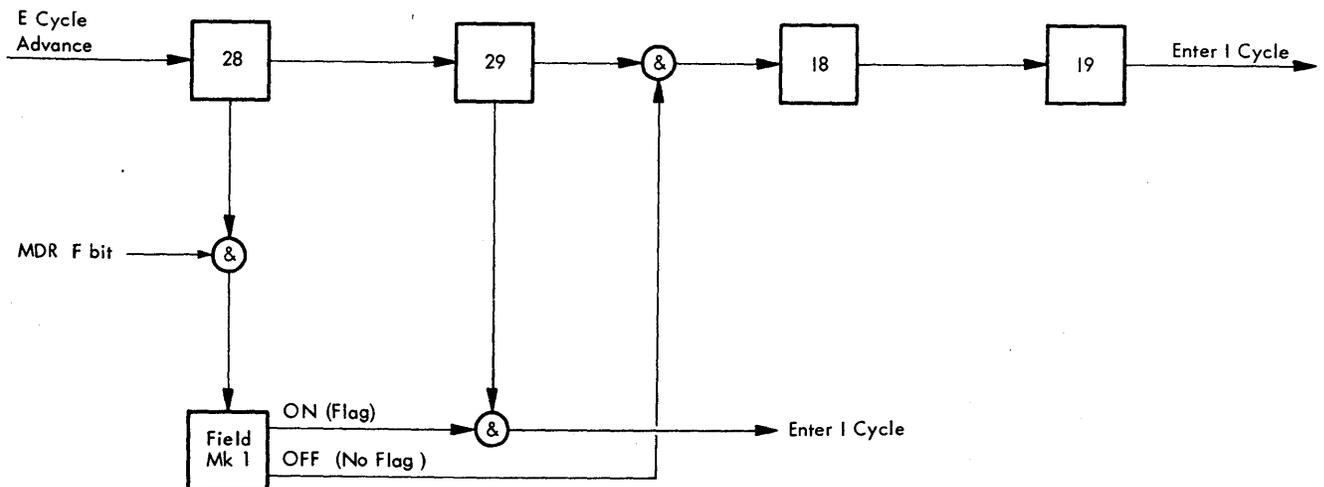


Figure 11-5. Branch No Flag—Code 44

2. If a flag bit is not present, branch to the instruction at the P address (OR-2).
3. If a flag bit is present, proceed to the next instruction in sequence (address in IR-1).

Functions

The memory location designated by OR-1 is read out to MDR and is interrogated for the presence of a flag bit. The field mark No. 1 trigger will be turned ON if a flag bit is present.

If a flag bit is present (field mark No. 1 ON), the computer proceeds to the next instruction in sequence.

If a flag bit is not present (field mark No. 1 OFF), the address in OR-2 is transferred into IR-1 and I-cycle is entered for the instruction at the P address (now in IR-1).

Auxiliary Trigger Status

Field Mark No. 1. (01.63.30.1).

1. Turned OFF by I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 28 time by the presence of an F bit in MDR.

E-Timer Trigger Objectives

Triggers 28 and 29 are used. Triggers 18 and 19 are also used if a flag bit is not present.

Trigger 28. (01.60.58.1).

1. Read out of memory per OR-1 and store the digit in MDR.
2. Turn field mark No. 1 trigger ON if an F bit is present in MDR.

Trigger 29. (01.60.59.1).

1. Interrogate the status of field mark No. 1 trigger.

(If OFF, proceed to trigger 18. If ON, end operation and enter I-cycle for the next instruction in sequence.)

Trigger 18. (01.60.48.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).

(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch no flag instruction (now stored in IR-1).

Branch No Record Mark (Code 45 — BNR)

The sequence block diagram for this operation is shown in Figure 11-6. The function chart for this operation is shown in the *Instructional System Diagrams*, page 31.

Objectives

1. Interrogate the single memory position designated by the Q address (OR-1) for the presence of a record mark character.
2. If a record mark character is not present, branch to the instruction at the P address (OR-2).
3. If a record mark character is present, proceed to the next instruction in sequence (address in IR-1).

Functions

The memory location designated by OR-1 is read out to MDR and is interrogated for the presence of a record mark (C-8-2). The digit/RM trigger will be turned ON if a record mark is present.

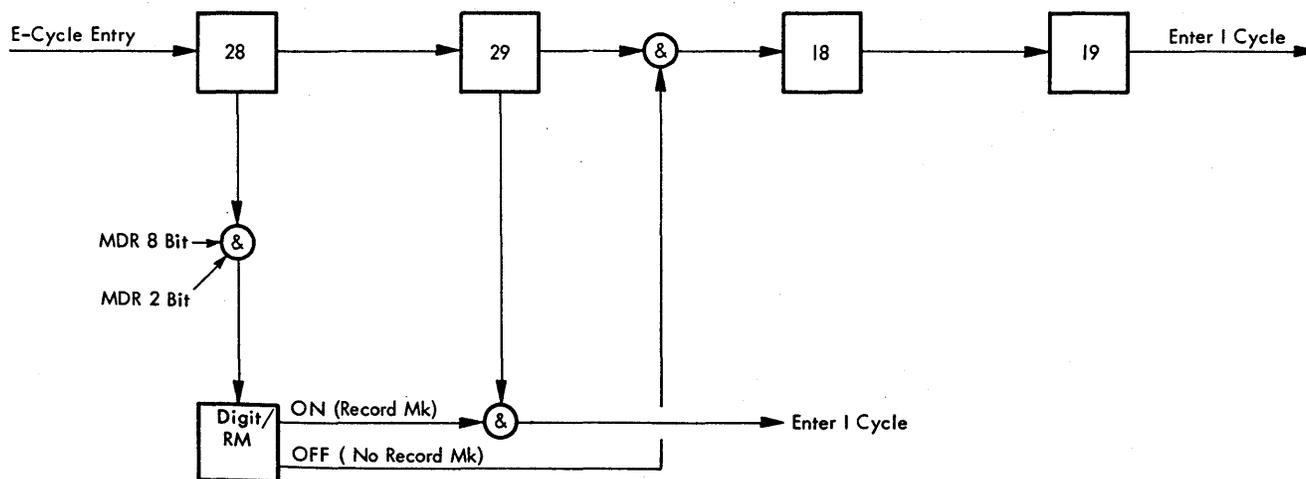


Figure 11-6. Branch No Record Mark—Code 45

If a record mark is present (digit/RM trigger ON), the computer proceeds to the next instruction in sequence.

If a record mark is present (digit/RM trigger ON), the address in OR-2 is transferred into IR-1 and I-cycle is entered for the instruction at the P address (now in IR-1).

Auxiliary Trigger Status

Digit/RM. (01.63.50.1).

1. Turned OFF by I-cycle trigger and therefore will be OFF when E-cycle is entered.
2. Turned ON during trigger 28 time by the presence of a record mark (C, 8, 2) in MDR.

E-Timer Trigger Objectives

Triggers 28 and 29 are used. Triggers 18 and 19 are also used if a record mark is not present.

Trigger 28. (01.60.58.1)

1. Read out of memory per OR-1 and store the digit in MDR.
2. Turn on the digit/RM trigger if a record mark is present in MDR.

Trigger 29. (01.60.59.1).

1. Interrogate the status of the digit/RM trigger. (If OFF, proceed to trigger 18. If ON, end operation and enter I-cycle for the next instruction in sequence.)

Trigger 18. (01.60.48.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).

(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch no record mark instruction (now stored in IR-1).

Branch Indicator (Code 46 — BI)

The sequence block diagram for this operation is shown in Figure 11-7. Function charts for this operation are shown in Figures 11-9, 11-10, 11-11 and in the *Instructional System Diagrams*, page 32.

Objectives

1. Interrogate the status of the indicator or program switch designated by the Q_8 and Q_9 digits of the instruction.
2. If the indicator or program switch is ON, branch to the instruction at the P address (OR-2).
3. If the indicator or program switch is OFF, proceed to the next instruction in sequence (address in IR-1).

Functions

During trigger 6 time of the I-cycle, the Q_8 and Q_9 digits of the instruction are placed in the digit/branch register where they are decoded to determine which indicator or program switch is to be interrogated. Examination of the status of the particular indicator or program switch is conducted during trigger 7 time to determine whether it is ON or OFF. The

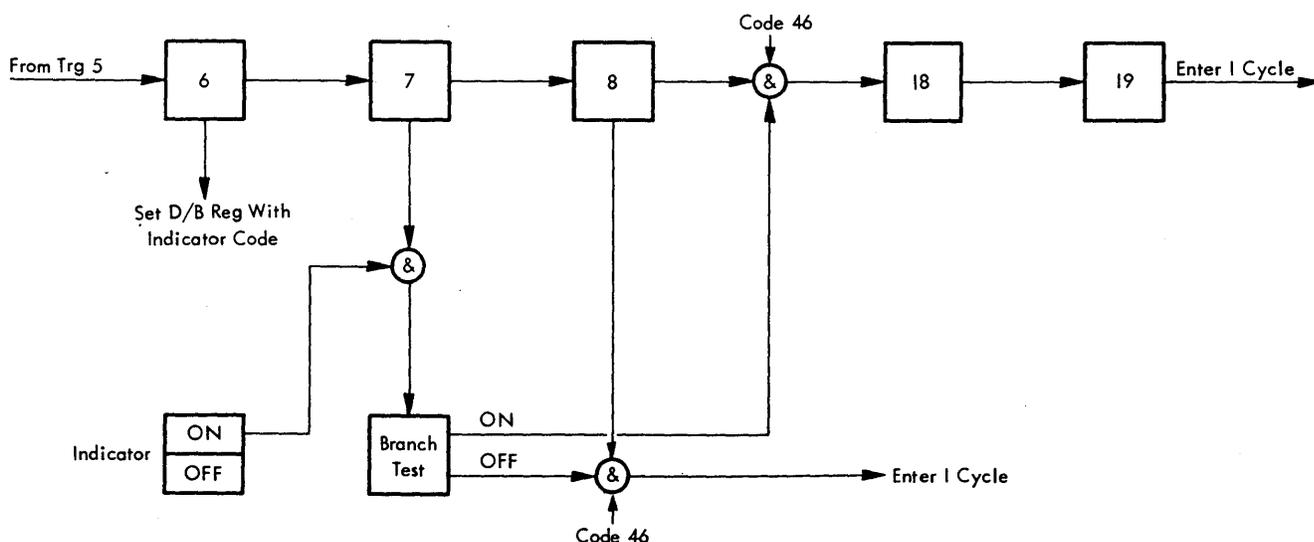


Figure 11-7. Branch Indicator—Code 46

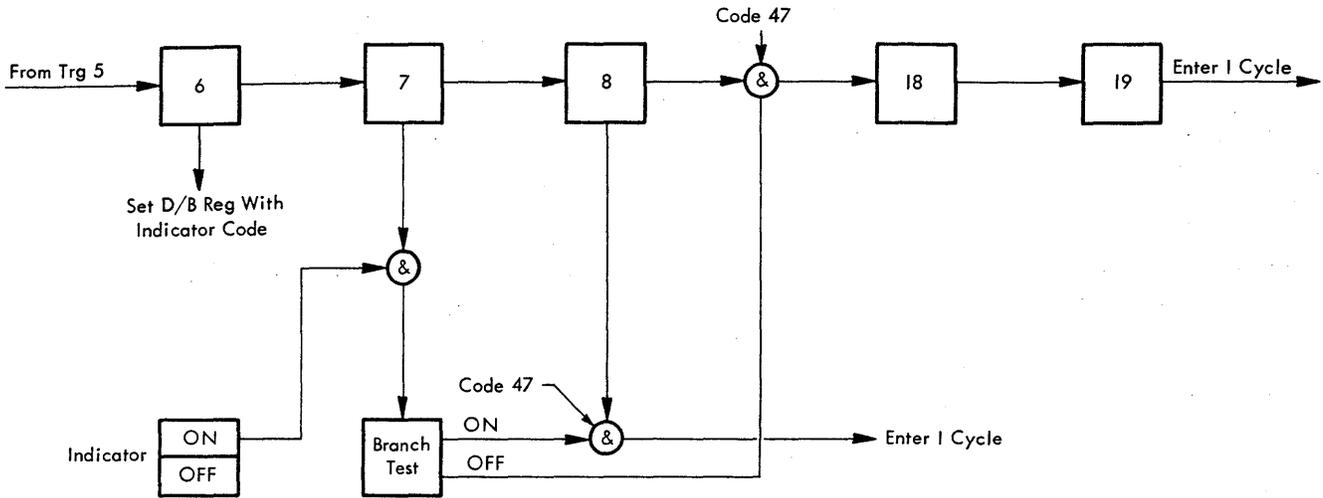
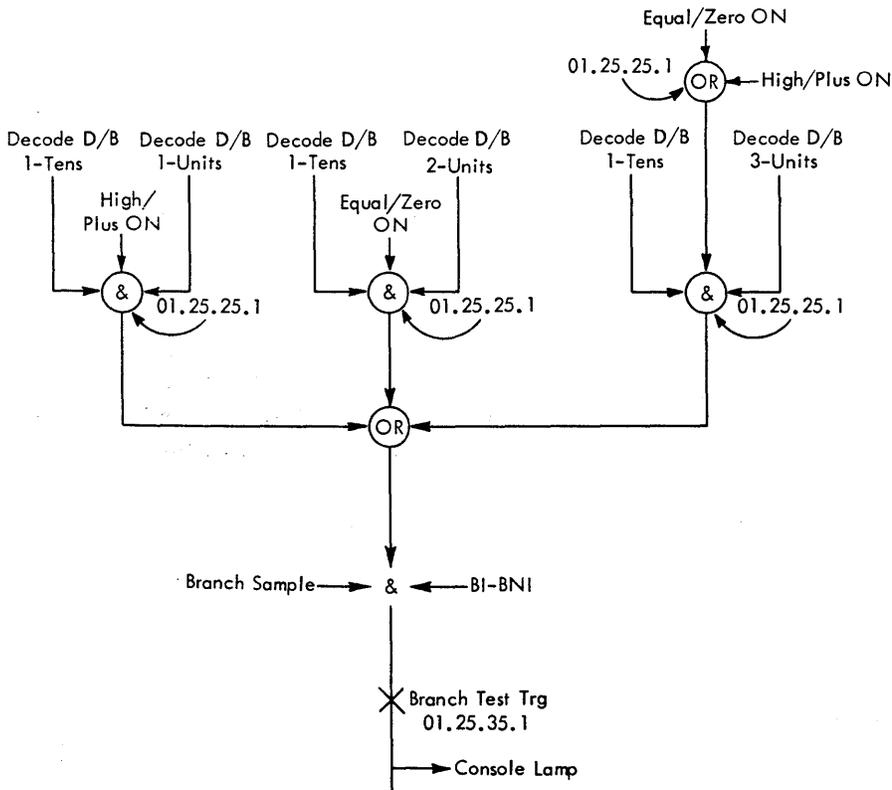
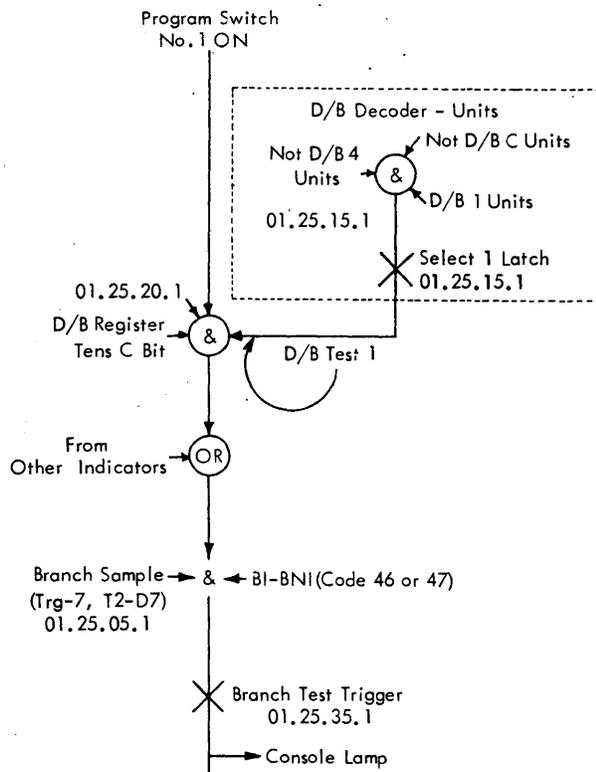


Figure 11-8. Branch No Indicator—Code 47



Code 46: Branch Can Occur if Either High/Plus OR Equal/Zero Trigger is ON.
 Code 47: Branch Can Occur Only if Both High/Plus AND Equal/Zero Triggers are OFF.

Figure 11-9. Branch Indicator—Branch No Indicator—Indicator Codes 11, 12, 13.



NOTE: For Program Switches 2, 3, & 4 the D/B Decoder Units Will Decode a 2, 3, or 4. Remainder of Function is Identical.

Figure 11-10. Branch Indicator—Branch No Indicator—Program Switch Indicator Code 01

branch test trigger will be turned ON if the indicator or program switch is on.

If the branch test trigger is ON, the address in OR-2 is transferred into IR-1 and I-cycle is entered for the instruction at the P address (now in IR-1).

If the branch test trigger is OFF, the computer proceeds to the next instruction in sequence.

The codes which appear in the Q_8 and Q_9 digits of this instruction for interrogation of program switches and indicators are assigned as follows:

- 01 — Program switch 1
- 02 — Program switch 2
- 03 — Program switch 3
- 04 — Program switch 4
- *06 — Read check indicator
- 07 — Write check indicator
- 08 — MAR check indicator (for CE use only)
- 09 — Last card indicator
- 11 — High/plus indicator

*Will cause 19 to be on.

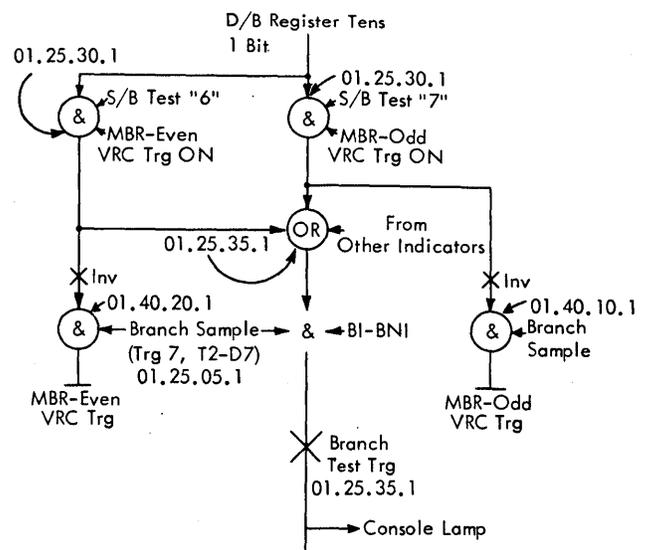


Figure 11-11. Branch Indicator—Branch No Indicator—MBR VRC Trgs Indicator Codes 16-17

- 12 — Equal/zero indicator
- 13 — High/plus or equal/zero indicator
- 14 — Overflow indicator
- *16 — MBR-even check indicator
- *17 — MBR-odd check indicator
- 19 — ANY data check indicator line

All indicators, except the high/plus, equal/zero, and the "any" line, are turned OFF, if ON, when interrogated by this instruction. Interrogation of a program switch has no effect upon its state, since the program switches are set manually at the console.

Interrogation of the high/plus indicator or the equal/zero indicator has no effect upon its state; they are automatically turned ON only at the start of arithmetic and compare operations.

Interrogation of "any" data check indicator line has no effect upon its state. The "any" data check indicator line is ON when one or more of the four data check indicators (read, write, MBR-even, and MBR-odd) is ON, and OFF when all four of the data check indicators are OFF.

Auxiliary Trigger Status

Branch Test Trigger. (01.25.35.1)

1. Turned OFF during I-cycle by trigger 2.
2. Turned ON during trigger 7 time of the I-cycle if the indicator or program switch that is designated for interrogation is ON.

E-Timer Trigger Objectives

Triggers 18 and 19 are used if a branch occurs. Only I-cycle triggers are used if no branch occurs.

Trigger 18. (01.60.48.1)

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).
(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch indicator instruction (now stored in IR-1).

Branch No Indicator (Code 47 — BNI)

The sequence block diagram for this operation is shown in Figure 11-8. Function charts for this operation are shown in Figures 11-9, 11-10, 11-11 and in the *Instructional System Diagrams*, page 32.

Objectives

1. Interrogate the status of the indicator or program switch designated by the Q_8 and Q_9 digits of the instruction.
2. If the indicator or program switch is OFF, branch to the instruction at the P address (OR-2).
3. If the indicator or program switch is ON, proceed to the next instruction in sequence (address in IR-1).

Functions

During trigger 6 time of the I-cycle, the Q_8 and Q_9 digits of the instruction are placed in the digit/branch register where they are decoded to determine which indicator or program switch is to be interrogated. Examination of the status of the particular indicator or program switch is conducted during trigger 7 time to determine whether it is OFF or ON. The branch test trigger will be turned ON if the indicator or program switch is ON.

If the branch test trigger is OFF, the address in OR-2 is transferred into IR-1 and I-cycle is entered for the instruction at the P address (now in IR-1).

The branch test trigger is ON, the computer proceeds to the next instruction in sequence.

The codes which appear in the Q_8 and Q_9 digits of this instruction for interrogation of program switches and indicators are assigned as follows:

- 01 — Program switch 1
- 02 — Program switch 2
- 03 — Program switch 3

- 04 — Program switch 4
- * 06 — Read check indicator
- * 07 — Write check indicator
- 08 — MAR check indicator (for CE use only)
- 09 — Last card indicator
- 11 — High/plus indicator
- 12 — Equal/zero indicator
- 13 — High/plus and equal/zero indicators
- 14 — Overflow indicator
- * 16 — MBR-even check indicator
- * 17 — MBR-odd check indicator
- 19 — ANY data check indicator line

All indicators, except the high/plus, equal/zero, and the ANY line, if ON are turned OFF when interrogated by this instruction. Interrogation of a program switch has no effect upon its state, since the program switches are set manually at the console.

Interrogation of the high/plus indicator or the equal/zero indicator has no effect upon its state; they are automatically reset OFF only at the start of arithmetic and compare operations.

Interrogation of the ANY data check indicator line has no effect upon its state. The ANY data check indicator line is ON when one or more of the four data check indicators (read, write, MBR-even, and MBR-odd) is ON, and OFF when all four of the data check indicators are OFF.

Auxiliary Trigger Status

Branch Test Trigger. (01.25.35.1)

1. Turned OFF during I-cycle by trigger 2.
2. Turned ON during trigger 7 time of the I-cycle if the indicator or program switch that is designated for interrogation is ON.

E-Timer Trigger Objectives

Triggers 18 and 19 are used if a branch occurs. Only I-cycle triggers are used if no branch occurs.

Trigger 18. (01.60.48.1).

1. Read OR-2 into MAR.
2. Write from MAR into IR-1 bypassed.

Trigger 19. (01.60.49.1).
(Dummy E Timer)

1. Block MAR reset to prevent VRC error.
2. End operation and enter I-cycle for the instruction at the P address of the branch no indicator instruction (now stored in IR-1).

* Will cause 19 to be on.

Section 12 Miscellaneous Operations

Set Flag (Code 32 — SF)

The sequence block diagram for this operation is shown in Figure 12-1. The function chart for this operation is shown in the *Instructional System Diagrams*, page 33.

Objective

Place a flag bit in the memory location designated by the P address (OR-2).

Functions

Two memory cycles are required as follows:

1. The content of the location in which a flag bit is to be stored is read from memory per OR-2 and stored in MDR.
2. Reset of MDR is blocked (Read Y). The content of the same memory location is again read out. The Read Y condition blocks odd or even sense amplifiers per OR-2 address to prevent entry into MBR, and clear the memory location.

The MDR F bit trigger is turned on (if off). The contents of MDR are transferred to MBR as a function of Read Y. If a C bit is present in MDR it will be cleared by the C bit corrector between MDR and MBR. If no C bit is present in MDR, it will be added by the C bit corrector between MDR and MBR. The contents of MBR are written into memory per OR-2.

The Q address (OR-1) is not used in this operation.

Auxiliary Triggers

None are used.

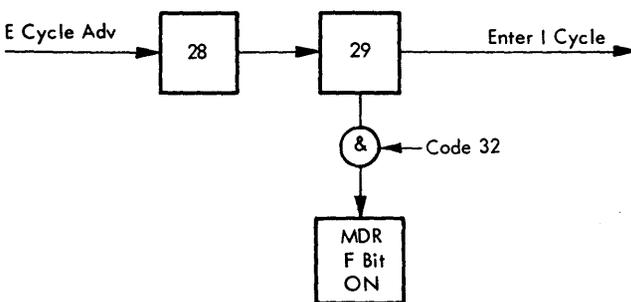


Figure 12-1. Set Flag—Code 32

E-Timer Trigger Objectives

Trigger 28. (01.60.58.1)

1. Read out of memory per OR-2 and store the digit in MDR.

Trigger 29. (01.60.59.1)

1. Block reset of MDR (Read Y).
2. Read out of memory per OR-2 with either the odd or even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
3. Set an F bit in MDR.
4. Transfer MDR to MBR setting or clearing a C bit in MBR to maintain odd parity.
5. Write into memory per OR-2 from MBR.
6. End operation and enter I-cycle for the next instruction in sequence.

Clear Flag (Code 33 — CF)

The sequence block diagram for this operation is shown in Figure 12-2. The function chart for this operation is shown in the *Instructional System Diagrams*, page 33.

Objective

Remove the flag bit, if present, from the memory location designated by the P address (OR-2).

Functions

Two memory cycles are required as follows:

1. The content of the location from which a flag bit is to be removed is read from memory per OR-2 and stored in MDR.

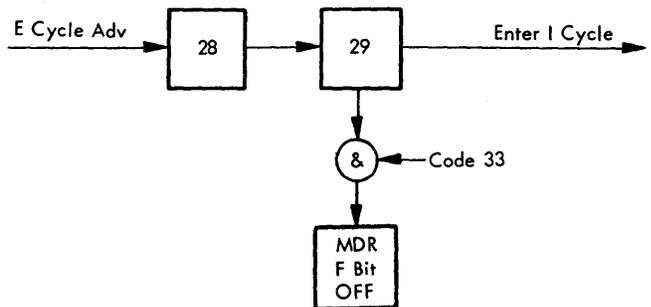


Figure 12-2. Clear Flag—Code 33

- Reset of MDR is blocked (Read Y). The content of the same memory location is again read out. The Read Y condition blocks the odd or even sense amplifiers per OR-2 address to prevent entry into MBR and clear the memory location. The MDR F bit trigger is turned off (if on). The contents of MDR are transferred to MBR as a function of Read Y. If a C bit is present in MDR it will be cleared by the C bit corrector between MDR and MBR. The contents of MBR are written into memory per OR-2.

The Q address (OR-1) is not used in this operation.

Auxiliary Triggers

None are used.

E-Timer Trigger Objectives

Trigger 28. (01.60.58.1)

- Read out of memory per OR-2 and store the digit in MDR.

Trigger 29. (01.60.59.1)

- Block reset of MDR (Read Y).
- Read out of memory per OR-2 with either the odd or even sense amplifiers blocked (Read Y), depending on whether the OR-2 address is odd or even, to clear the memory location.
- Turn off the MDR F bit trigger.
- Transfer MDR to MBR setting or clearing a C bit in MBR to maintain odd parity.
- Write into memory per OR-2 from MBR.
- End operation and enter I-cycle for the next instruction in sequence.

Halt (Code 48 — H)

The sequence block diagram for this operation is shown in Figure 12-3. The function chart for this operation is shown in the *Instructional System Diagrams*, page 9.

Objective

Stop the machine upon completion of the I-cycle of this instruction.

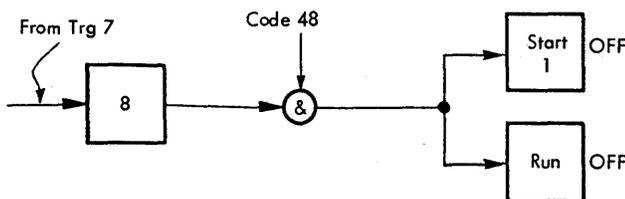


Figure 12-3. Halt—Code 48

Function

The machine stops at the end of I-cycle. The manual status light is turned ON when the run trigger is turned OFF. The automatic status light is turned off when the start I trigger is turned off. P and Q addresses are not used in this operation.

Auxiliary Triggers

None are used.

E-Timer Triggers

None are used. This operation is completed by trigger 8. Refer to I-cycle function chart shown in the *Instructional System Diagrams*, page 9.

Trigger 8. (01.15.18.1)

- Turn OFF start I trigger.
- Turn OFF run trigger.

No Operation (Code 41 — NOP)

The sequence block diagram for this operation is shown in Figure 12-4. The function chart for this operation is shown in the *Instructional System Diagrams*, page 9.

Objective

Complete the I-cycle of this instruction and proceed to the next instruction in sequence (address in IR-1).

Functions

See Objective. The P and Q addresses are not used in this operation.

Auxiliary Triggers

None are used.

E-Timer Triggers

None are used. This operation is completed by trigger 8. Refer to I-cycle function chart shown in the *Instructional System Diagrams*, page 9.

Trigger 8. (01.15.18.1)

- Enter I-cycle for next instruction in sequence.

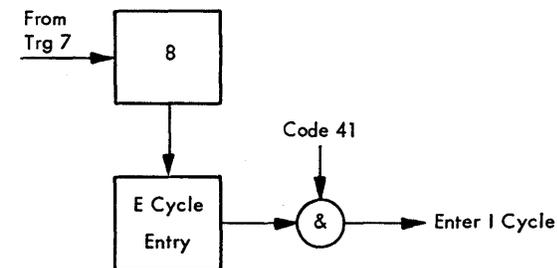


Figure 12-4. No Operation—Code 41

Typewriter

The console typewriter for the IBM 1620 Computer is a modified IBM B1 electric typewriter. For a description of typewriter features and mechanical operations, reference should be made to the *CE Manual of Instruction for the B1 Electric Typewriter (Modified for DP Equipment)*, (Form 223-6653).

The typewriter for the 1620 Computer has modifications not described in the above mentioned manual. These modifications are described as follows:

1. The selector common contact assembly consists of two normally open contacts. Figure 13-1.
2. The modified keyboard arrangement is shown in Figure 13-2.

IBM 1621 Paper Tape Reader

Purpose

The IBM 1621 Paper Tape Reader reads 8-channel paper tape at a rate of 150 characters per second. The 1621 uses chad tape only, that is, tape with the holes completely punched out. Tape with the holes not completely punched out is called chadless tape. Three modes for reading paper tape are provided; reel feed, strip feed, and center roll feed. Functionally, the reader

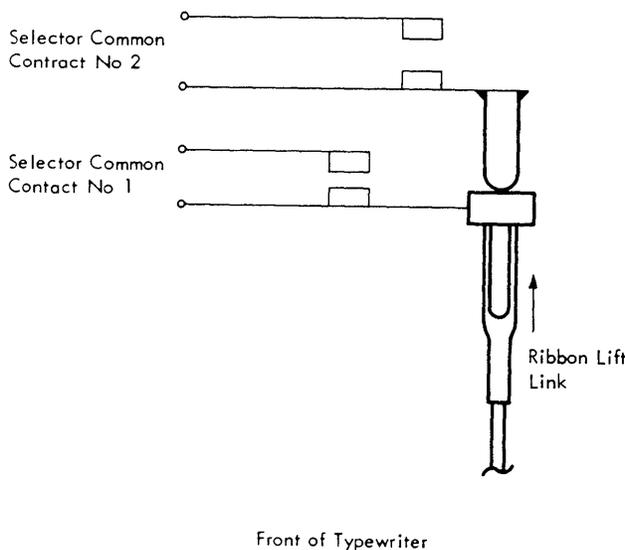


Figure 13-1. Typewriter Selector Common Contacts

has mechanisms for (1) reading, (2) driving the tape past the read head, and (3) turning the tape reels.

READING MECHANISM

The head assembly contains eight photocells to sense the light that passes through holes punched in paper tape as the tape moves over the read head. The photocell outputs are amplified and supplied to the 1620 as S levels.

TAPE DRIVING MECHANISM

A pinwheel, located in the read head assembly, meshes with feedholes in the tape and drives the tape from right to left under control of a clutch-and-brake.

REEL DRIVING MECHANISM

Paper tape is supplied to, and removed from, the head by independently controlled tape reels. Reel rotation is controlled by the action of buffer arms placed in the tape path.

Component Locations

MAIN PLATE, FRONT (FIGURE 13-3).

Mounted on the front of the main plate are the following components, all in the tape path.

The supply reel (on the right), buffer idler rolls, supply buffer arm, read head idler, tape tension bar, tape runout bar, read photocells, read head idler, buffer idlers, take-up buffer arm, take-up reel, and center roll feed motor reversing switch arm.



Figure 13-2. Typewriter Keyboard

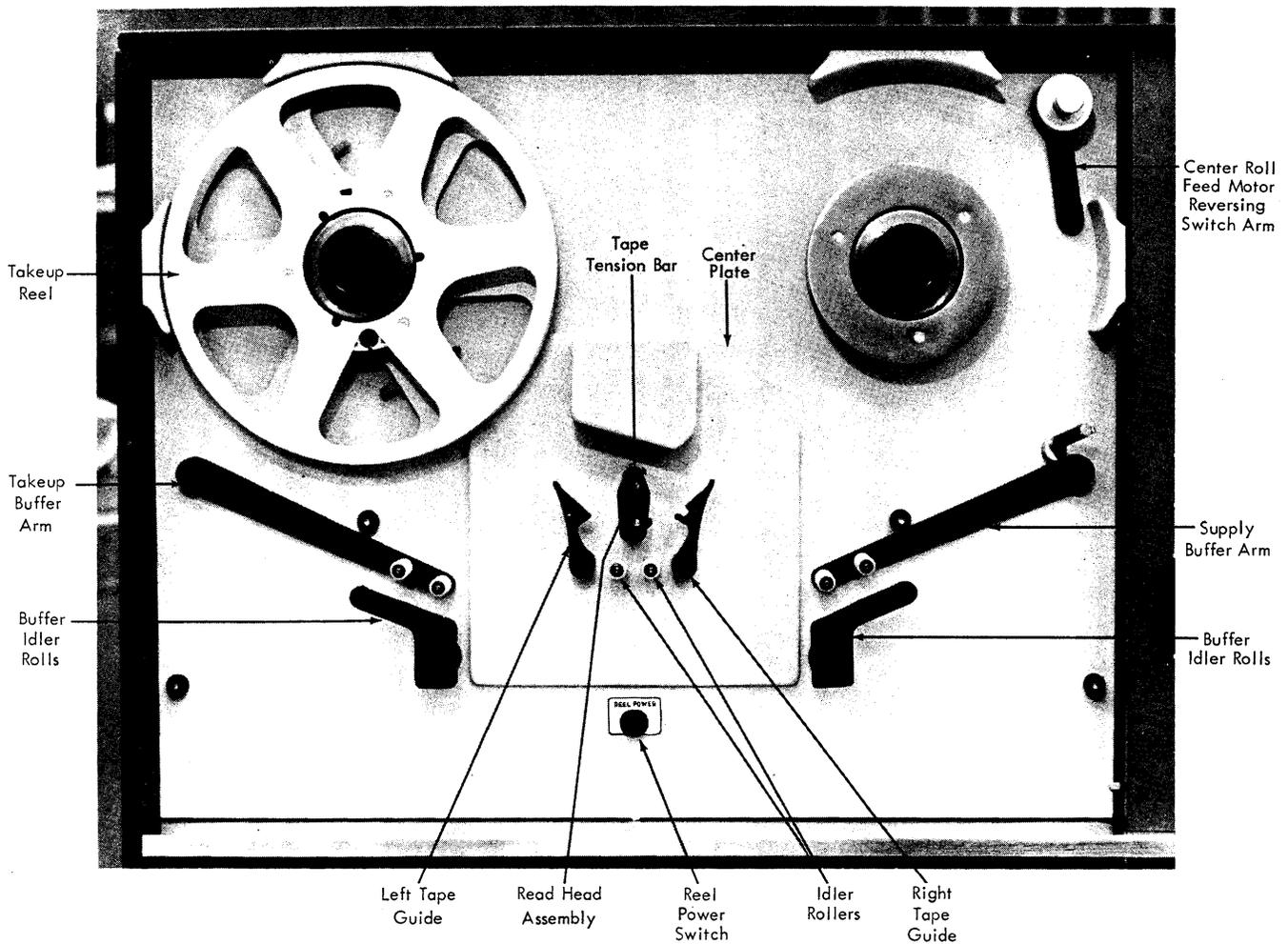


Figure 13-3. Main Plate, Front

The read head assembly is a part of the center plate, which supports most of the tape-driving mechanism. Below the center plate is the reel power switch.

MAIN PLATE, REAR (FIGURE 13-4).

Mounted on the back of the main plate are the following (reading clockwise from lower left corner):

1. SMS card gate
2. Supply buffer arm cam
3. Supply reel pulley
4. Center roll feed
Motor reversing switch
5. Supply reel motor
6. Take-up reel motor
7. Relays R1-R2-R3
8. Read lamp potentiometer
9. Brake potentiometer
10. Clutch potentiometer
11. Timing disk lamp potentiometer
12. Take-up buffer arm cam

13. Main drive motor
14. Center plate assembly
15. Clutch shaft

REEL-STRIP MODE SWITCH

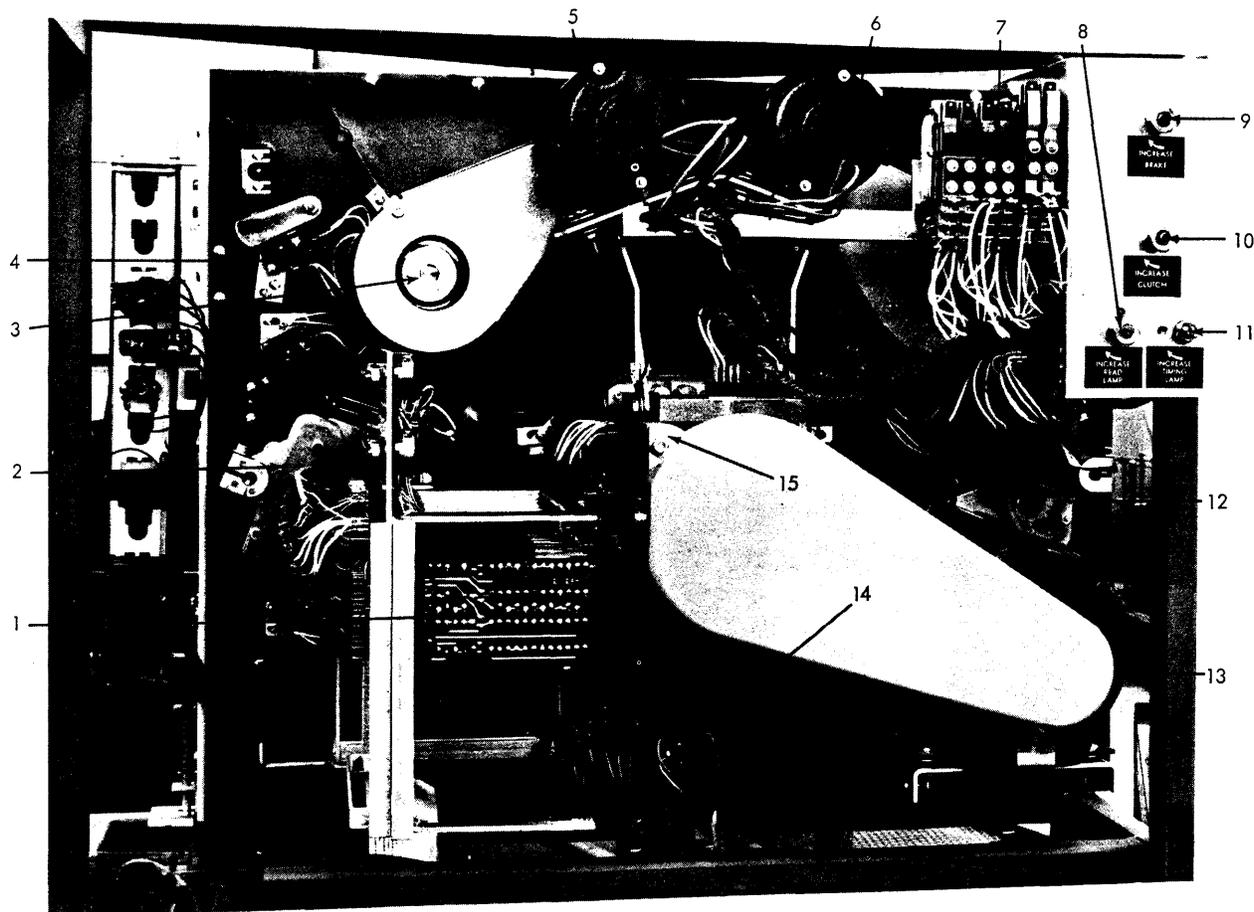
A reel-strip mode switch is mounted on the paper tape reader and allows the reader to be placed in either reel mode or strip mode.

Miscellaneous Components. System Diagram 02.83.00.1 shows the location of relays, edge connectors, and terminals. For a brief definition of the major components and their function, refer to the Glossary.

Mechanical Components

CLUTCH

The continuously running main drive motor transmits motion through a drive belt to a pulley fixed to the rear of the clutch shaft, which turns the clutch rotor. When the clutch is energized, the armature is



- | | |
|---|------------------------------------|
| 1. SMS Card Gate | 8. Read Lamp Potentiometer |
| 2. Supply Buffer-Arm Cam | 9. Brake Potentiometer |
| 3. Supply Reel Pulley | 10. Clutch Potentiometer |
| 4. Center Roll Feed
Motor Reversing Switch | 11. Timing Disk Lamp Potentiometer |
| 5. Supply Reel Motor | 12. Takeup Buffer - Arm Cam |
| 6. Takeup Reel Motor | 13. Main Drive Motor |
| 7. Relays R1-R2-R3 | 14. Center Plate Assembly |
| | 15. Clutch Shaft |

Figure 13-4. Main Plate, Rear

attracted to the rotor and turns with it. The armature, in turn, drives the pinwheel shaft, which moves tape past the read head (Figure 13-5). The clutch armature can slide, axially, along the splined hub that is fixed to the pinwheel shaft. The splined hub provides the rotational force when the armature is magnetically attracted to, and turning with, the continuously turning rotor fixed to the clutch shaft.

BRAKE

Simultaneously with clutch de-energization, the electromagnetic brake is actuated. This brake is similar in operation to the clutch, except that the brake-rotor is stationary, while the armature turns with the pin-

wheel shaft. When the brake is actuated, the rotating armature is attracted and sealed to the stationary rotor, thus stopping the pinwheel shaft.

NOTE: The brake and clutch are never energized simultaneously.

READ HEAD ASSEMBLY

Figure 13-6 shows the read head assembly, with the tape guides open. The read head assembly is a two-section shell that encloses the pinwheel. Mounted in the top-left of the assembly are eight silicon photo-duodiods (read photocells). These photocells are energized, during reading, by collimated light from an incandescent lamp located above the read head.

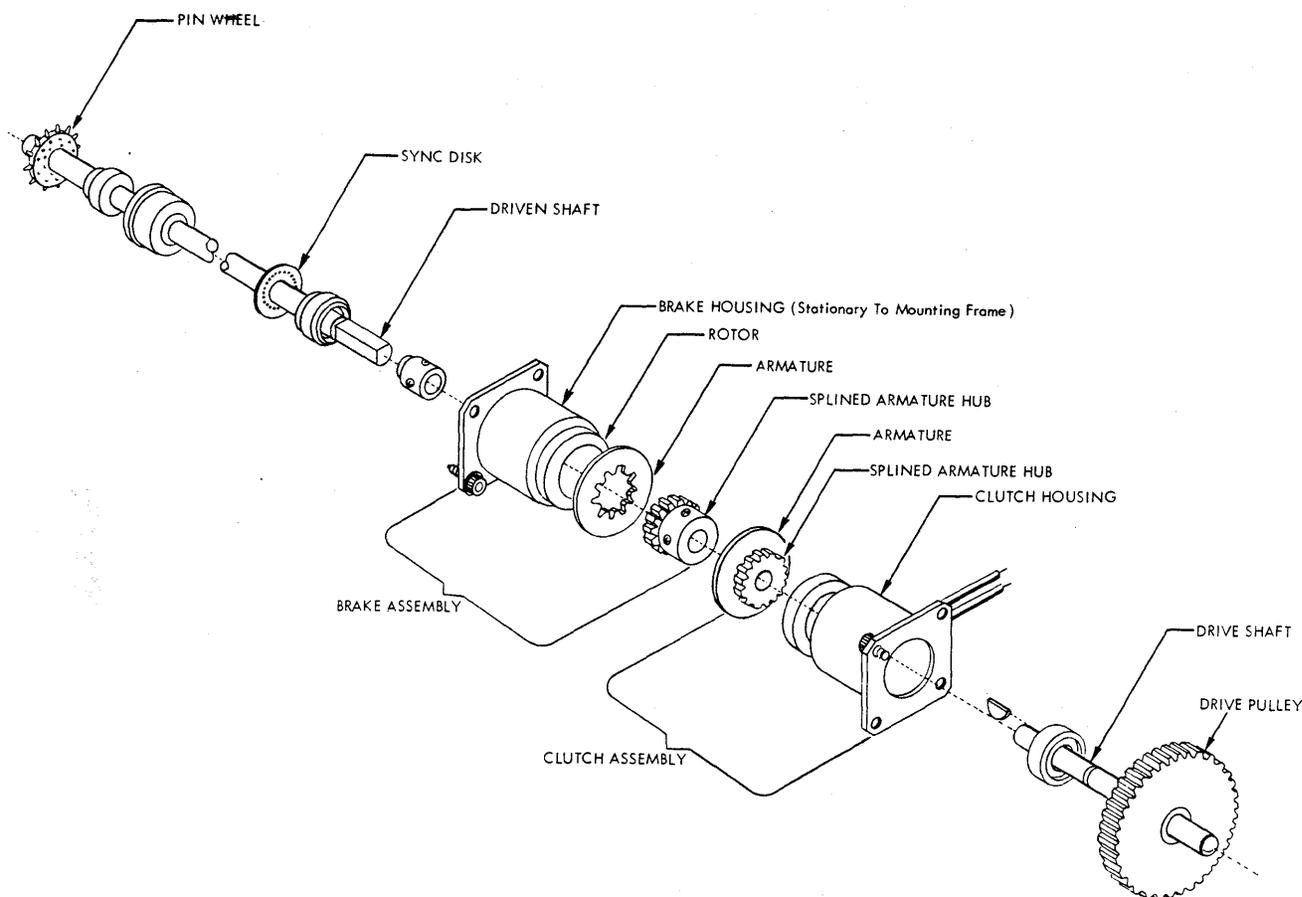


Figure 13-5. Drive Shaft Assembly

Adjacent to the read head assembly, but not a part of it, are the tape tension bar and the tape runout bar. These bars operate microswitches located on the back of the center plate, and through electrical circuits control the operation of the tape reader when tape runs out, breaks, or is not properly loaded at the read head.

The tape guide assembly has two functions: (1) to hold the tape in mesh with the pinwheel and in contact with the read head; (2) to actuate the tape guide microswitch when the guides are closed.

The tape guide assembly is divided into two sections: the tape guides in front of the center plate, and the stop arms to the rear of the center plate. Since an adjustable clevis connects the two arms, both guides and stop arms move as a unit.

TIMING DISK (SYNC DISK)

Associated with the reading mechanism is the timing disk, which is pinned to the pinwheel shaft. The disk contains 24 holes around its periphery, these holes correspond to the 24 pins on the pinwheel and permit

adjustment of timing pulses generated during reading. Figure 13-5 shows the drive shaft assembly with timing disk (sync disk).

Reel Operation

PREPARING FOR REEL OPERATION (02.83.70.1)

To put the reader in reel operation (machine power on):

1. Set the reel-strip mode switch for reel operation.
2. Mount the tape reels on their hubs.
3. Rotate each buffer arm inward until it detents (Figure 13-7).
4. Thread the tape as shown in Figure 13-7.

NOTE: a. The three-hole side of the tape must be toward the front. Tape level or channel 1 is to the front.

b. There must be no slack in the tape, at the read head, when the guides are closed.

5. Close the tape guide.
6. Release each buffer arm from its detent positions, allowing the buffer-arm rollers to rest on the tape.

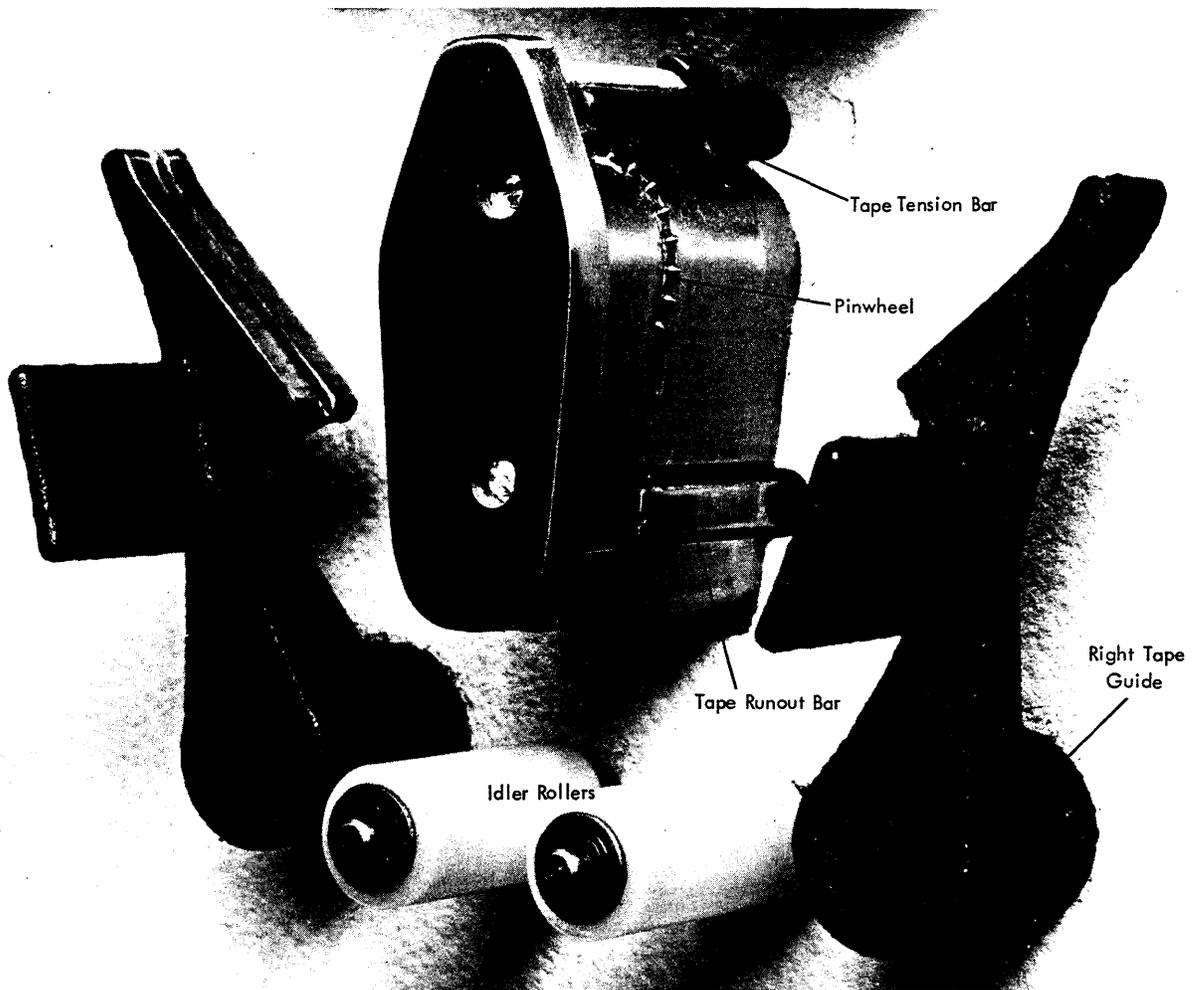


Figure 13-6. Read Head Assembly

7. Depress the reel-power switch. The reader is ready for operation. After the reel-power switch is depressed, both buffer arms assume their neutral position and the reels are placed under the control of their respective buffer arms. Operation of both buffer arms is similar and may be understood by studying the supply reel operation.

REEL POWER

After the supply buffer arm is released from its detent position and lowered onto the tape, the depression of the reel power switch starts a chain of events as follows:

1. Relays 1 and 3 are picked providing the supply reel limit switches are closed.
2. Relay 2 is picked providing the take-up reel limit switches are closed.
3. The buffer arm having been released from the detented position, allows the rollers to rest on the tape until tape slack is exhausted or the lower buffer arm contact is closed by the buffer arm cam.

Closure of the lower buffer arm contact (Figure 13-8) drives the supply motor to take up the slack in the tape. The motor turns until the lower contact is opened — the supply buffer arm is now in its neutral position.

4. Functions of relays R1, R2, R3 (02.83.70.1)

R1 — Controls the supply motor and reel ready condition.

1AL: Holds R1 energized after the reel power switch is released.

1BL: Directs -48v to the supply reel buffer arm contacts.

R2 — Controls the take-up motor and reel ready condition.

2AL: Holds R2 energized after the reel power switch is released.

2BL: Directs -48v to the take-up reel buffer arm contacts.

2AU: Produces a take-up reel-ready signal.

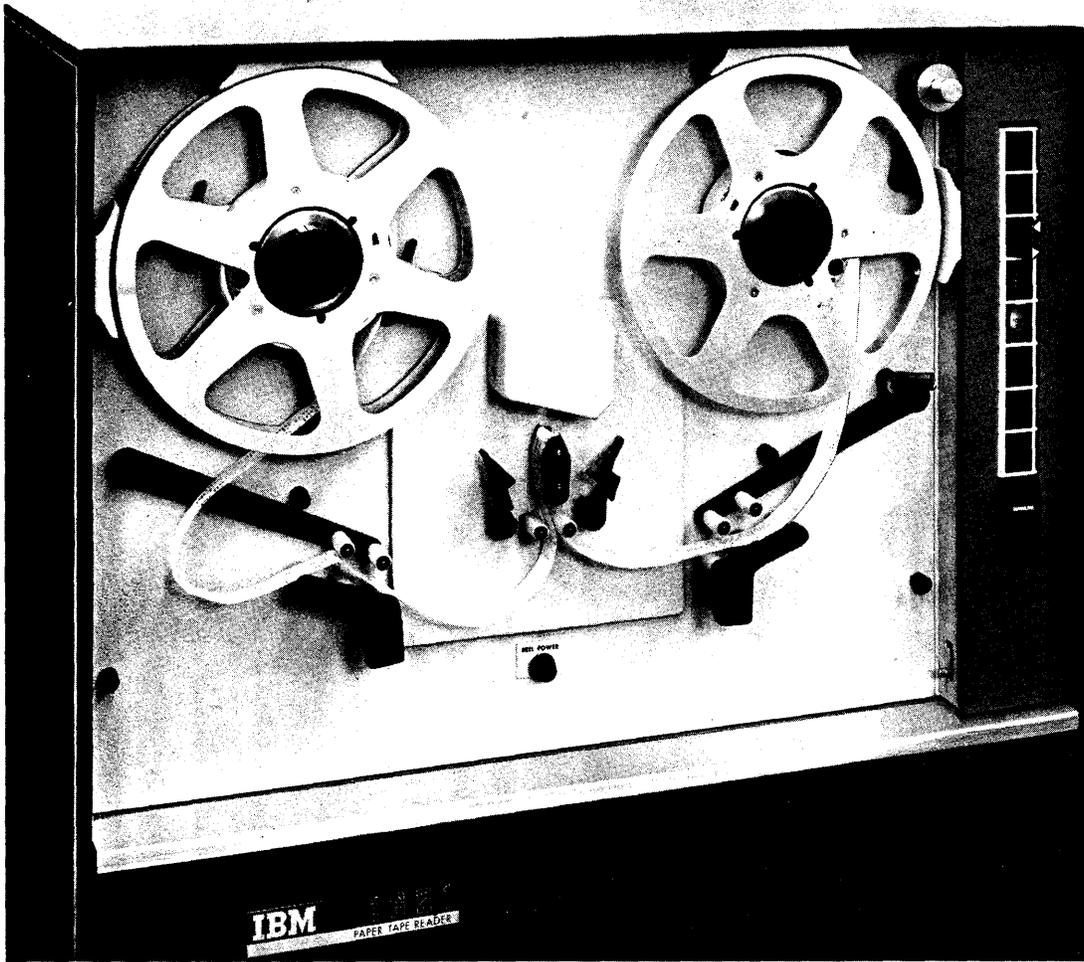


Figure 13-7. Reel Mode

R3 – Retains the reels-ready condition when tape runs out.

3AL: Holds R3 energized after the supply reel lower limit switch opens on tape runout.

3AU – 3BU: Produces a supply reel ready signal.

Strip Mode Operation

The strip mode operation provides a means for reading a strip of paper tape when the short length of the tape prevents its being placed on either the supply or take-up reels. The buffer arm controls must be bypassed and the take-up and supply reel ready lines must be brought to the ready condition (+S).

1. The reel strip mode switch is placed in the strip position. This gives a +S "Supply Reel Ready" and +S "Take-Up Reel Ready" without regard to the status of R2 or R3. The circuit to the supply

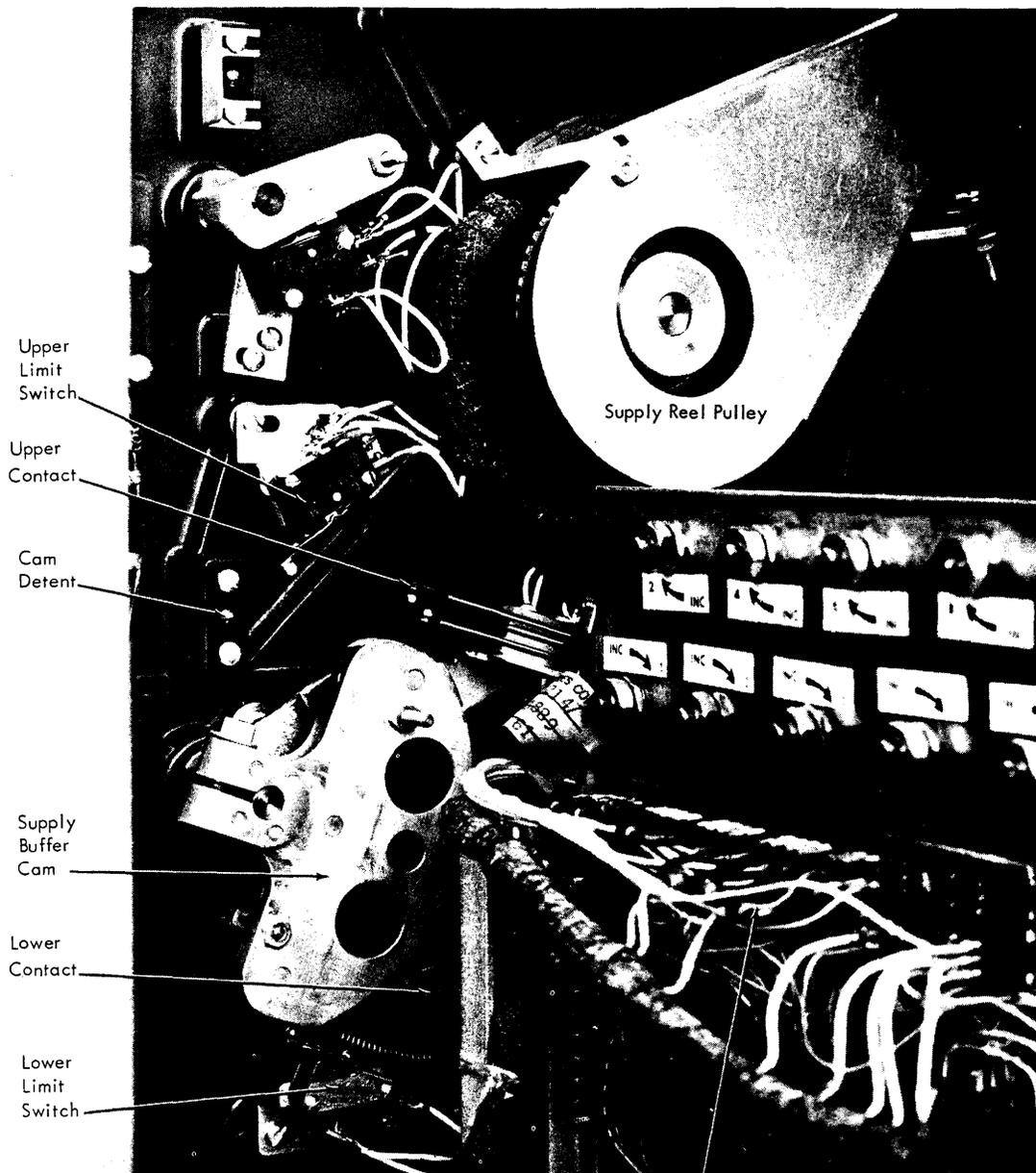
and take-up reel motor armatures is opened. Buffer arm position has no control.

2. The strip of tape is threaded onto the read head in the same manner as reel or CRF (center roll feed) mode. The tape break, tape load, and tape runout contacts are made. The paper tape reader is now in a ready status.

CRF Mode (Center Roll Feed)

The only difference between CRF mode and reel mode is the manner in which the supply reel is used.

Tape is loaded from the CRF table over the top of the supply reel from right to left. The center roll feed motor reversing switch arm is placed against the tape to hold it onto the supply reel hub. The supply reel motor now runs in reverse compared to reel mode. All other load conditions are identical with reel mode. See Figure 13-9.



SMS Card Gate

Figure 13-8. Supply Buffer Arm Contacts & Limit Switches

Reading Tape

As the pinwheel moves tape past the read head, the supply buffer arm rotates clockwise (upward) until the upper buffer arm contact is closed. This rotates the supply motor and reel clockwise, and delivers tape to the buffer arm. The buffer arm then lowers until the upper buffer arm contact opens, whereupon the motor stops. As long as tape is being read, the upper buffer arm contact continuously closes and opens to control the supply reel rotation.

When the pinwheel stops, the supply buffer arm returns to its neutral position, wherein the cam is rest-

ing between the upper and lower contacts and neither is made. When neither contact is made, the armature of the reel motor is shorted: this shorting results in a dynamic braking effect that helps stop the reel.

Take-up Reel

The operation of the take-up reel and buffer arm is almost a mirror-image of supply reel operation. As tape feeds to the take-up buffer arm, the arm lowers, causing the buffer arm cam to close the lower contact. The take-up reel then rotates counterclockwise, pulling the tape, and raising the buffer arm to its neutral position.

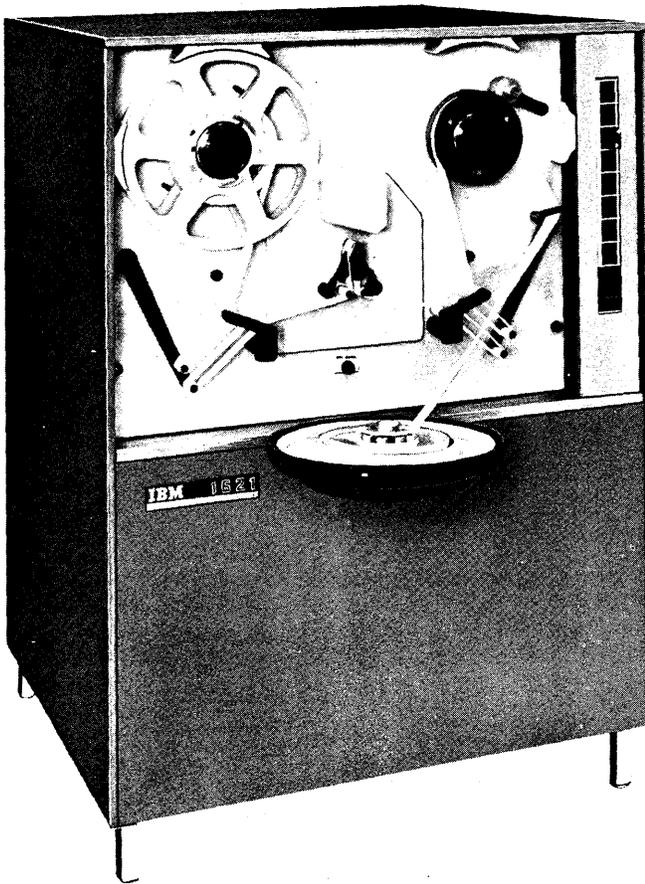


Figure 13-9. Center Roll Feed Mode

As with the supply operation, oscillation of the buffer-arm continues as long as the tape is being moved past the read head.

Limit Switches

The upper buffer arm limit switches prevent reel operation when the buffer arms are detented. See Figure 13-8.

The lower limit switches control reel operation when the tape breaks or runs out. When tape runs out, the supply reel buffer arm drops opening the lower limit switch. However, the hold for R3 bypasses the limit switch and therefore R3 will not drop, thus maintaining supply reel ready. R1 drops, opening the circuit to the supply reel drive motor. Control of the runout is maintained by the runout contact.

Reader Ready Lines

The operating condition of the paper tape reader is indicated by the following lines (02.83.60.1):

1. "+S take-up reel ready."

2. "+S tape break contacts."
3. "+S tape runout contacts."
4. "+S tape load contacts."
5. "+S supply reel ready."

The above 5 lines AND'ed together provide the "-S reader ready" line to the computer.

Incandescent Lamps

The read lamp, mounted above the read head, has a masked lens that exposes a beam of light on the eight read photocells.

The timing disk lamp is a miniature 12-volt lamp mounted behind the timing disk and photocell. No lens is required.

Each lamp is in series with a potentiometer for minimizing the voltage across the lamp and extending lamp-life.

Electronics

PAPER TAPE READ CIRCUIT (FIGURE 13-10)

The paper tape read circuit amplifies and shapes the nearly sinusoidal output of the read photocell, producing S levels at the circuit inverter output. When the tape blocks light from the photocell, the output is at a -S level (-12v). As a hole in the moving tape exposes the photocell to light, the resistance of the cell decreases. When point A becomes sufficiently negative, T₁ becomes forward biased and conducts, making point B negative. (Since the cell has been dark, T₂ has been conducting heavily, with point D positive.) As T₁ conducts and point B goes negative, T₂ conducts less and point C follows point B excluding a small drop through T₂. With point C negative, point D is also negative, forward biasing T₃, and causing it to conduct: the output rises to the +S level (ground).

As the tape blocks light from the photocell, the current through it is reduced to a maximum of 60 μ a, raising the voltage at point A to approximately +1.5 volts. T₁ continues conducting; point B is also at +1.5 volts. T₂ now has a greater forward bias, conducts more heavily, and makes point C positive (approximately +1.7 volts). This plus level is reflected at point D, cutting T₃ off, and dropping the output to the -S level (approximately -5.6 to -12 volts).

The DC voltage requirements are:

Turn-on: -2.37 to -12.0 volts.

Turn-off: -0.11 to +12.0 volts.

Current requirements are:

Minimum turn-on current:

165 μ a with input at -2.37 volts.

Maximum turn-off current:

60 μ a with input at -0.11 volts.

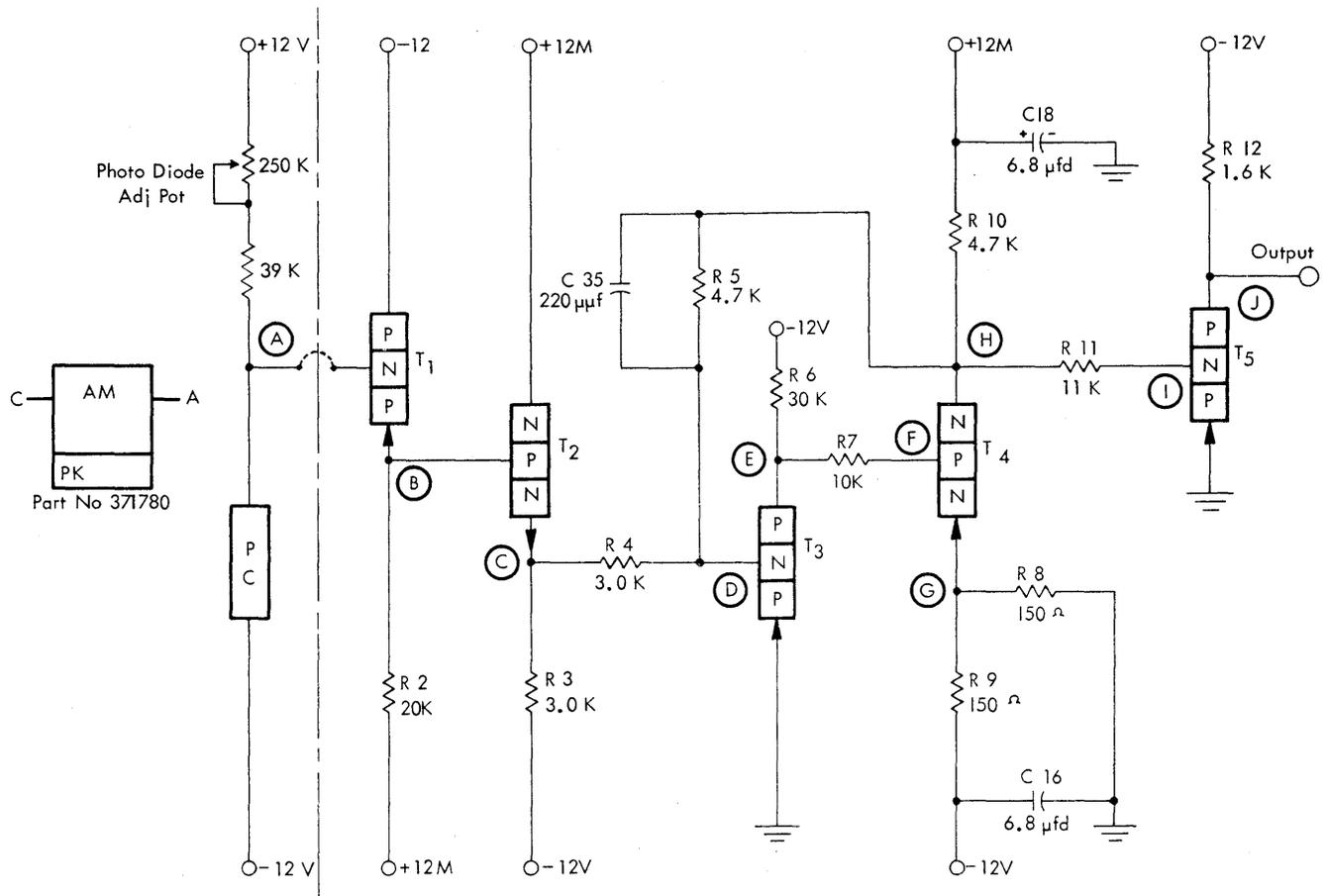


Figure 13-11. Timing Disk Photocell Amplifier

5. The output is at +S level (0 volts).

As the timing-disk moves to block light from reaching the photocell, the resistance of the cell increases, making points A and B go positive. T_2 conducts and point C follows point B in its positive voltage swing. When the input (point A) is nearly +6 volts, point D is sufficiently positive to turn off T_3 . Point E becomes negative, reverse biasing T_4 , and turning it off. Since T_3 and T_4 are off, the voltage at point H (+11.2 volts) is determined by the divider:

+12v to R10 to R5 to R4 to point C (+ 10 volts).

T_5 is reverse biased and turned off, making the output at a -S level (-12v).

PHOTO DIODE ADJUSTING POTENTIOMETERS

The photo diode adjusting potentiometers are used to compensate for inherent differences between photo diodes. Changing the potentiometer setting shifts the entire photocell amplifier output waveform up or down in relation to a base level. Adjusting the potentiometer does not increase or decrease the potential difference between the positive and negative peaks.

BRAKE AND CLUTCH: SELECTOR AND CONTROL CIRCUIT (FIGURE 13-12)

A single input to the selector and control circuit operates either the clutch or brake. Because a +S input energizes the clutch, and a -S input energizes the brake, the hazard of simultaneously operating the clutch and brake is reduced.

Input at -S (-12 volts). A -S input at point D energizes the brake, but not the clutch.

Initial Conditions: T_2 does not conduct because its base and emitter are both at -12 volts. Because of the R15-R13 divider, the base of T_1 is at -6 volts.

Operation: Since point A (T_2 emitter) is at -12 volts, T_1 is forward biased and conducts. Point A is now at -6 volts, reverse biasing T_2 and preventing it from conducting.

When T_1 conducts, the base of T_3 is negative; T_3 is forward biased and conducts, which makes the base of T_5 positive. T_5 is reverse biased and does not conduct; preventing the clutch from being energized.

Meanwhile T_2 is held off, making the base of T_4 positive. Being reverse biased, T_4 cannot conduct. The

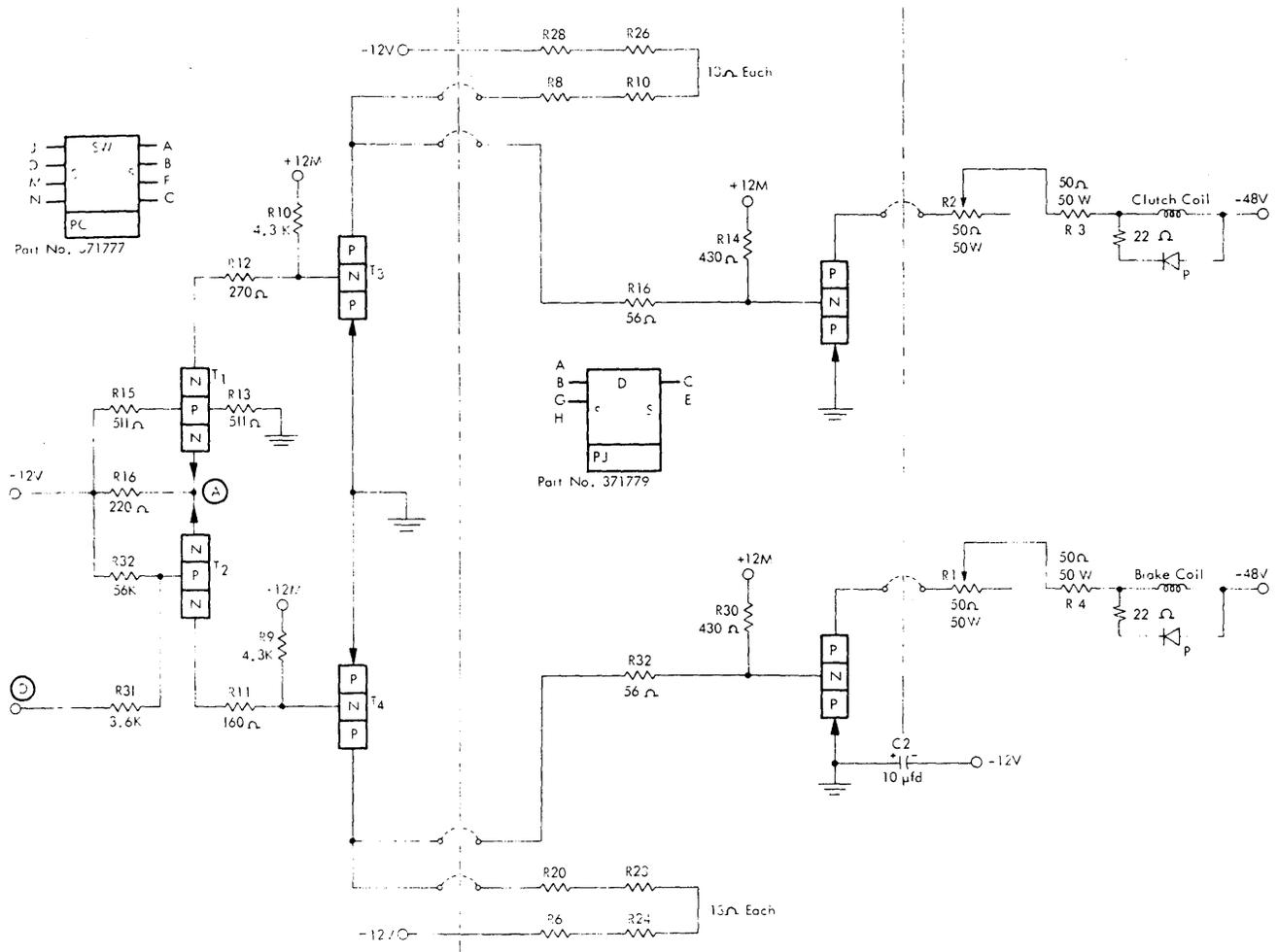


Figure 13-12. Brake and Clutch-Selector and Control Circuit

base of T_6 is negative and it conducts, causing the brake to be energized.

Input at +S (0 volts). With 0 volts at point D, T_2 is forward biased and conducts, making the base of T_4 negative. T_4 is forward biased and conducts. Point A goes from -12 volts to -5 volts, reverse biasing T_1 and cutting it off. When T_4 conducts, the base of T_3 is positive and T_6 turns off, de-energizing the brake.

With T_1 cut off, T_3 is reverse biased and cut off, making the base of T_5 negative. As T_5 conducts, the clutch is energized.

IBM 1624 Paper Tape Punch

Drive Mechanism

The punch is powered by a 1725 RPM, 1/40 HP motor. The clutch pulley is belt-driven from the motor (Fig-

ure 13-13). When the clutch is engaged, the clutch pulley transmits motion to the horizontal cam shaft that has the five electrical cams and the index on the left end of the shaft. By means of spiral bevel gears, the vertical shaft is driven from the horizontal shaft. The six mechanical cams (numbered top to bottom) are mounted on the vertical shaft. Cams 2 and 3 are the punch drive cams; cams 1 and 4 restore the punch drive mechanism; cam 5 operates the latch lock mechanism; cam 6 operates the tape feed; the roller stud between cams 5 and 6 operates the lever latch and magnet armature restoration.

The punch clutch permits operation of the tape punching mechanism under control of the clutch magnet (Figure 13-14). At zero on the index, the horizontal cam shaft is held in a detented position by the clutch detent arm and the clutch armature. The clutch spring at this point is held open or partially uncoiled from

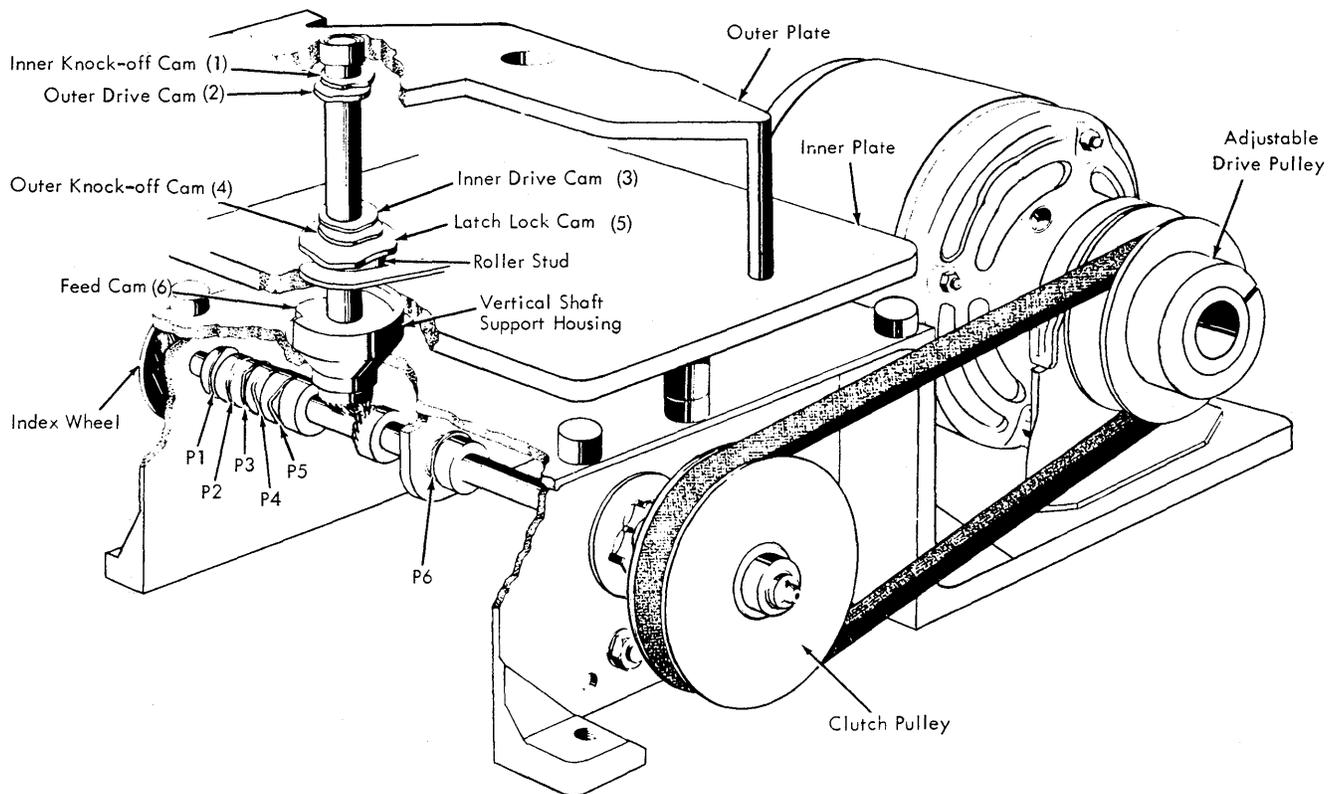


Figure 13-13. Tape Punch Drive and Cams

its normal position by the pressure of the armature against the step on the clutch sleeve. This uncoiled condition of the spring allows free rotation of the drive pulley.

When the clutch magnet receives an impulse, the armature is attracted and the clutch sleeve is free to rotate under pressure of the clutch spring. This rotation causes the spring to grip tightly around the collar on the drive pulley and the clutch collar. Motion is then transmitted from the drive pulley to the clutch collar through the clutch spring.

When the clutch latches at the end of a cycle, the spring again is uncoiled by the action of the armature engaging the step on the sleeve and overthrow of the horizontal shaft. This allows the detent arm to drop in behind the step on the clutch detent to keep the cam shaft from drifting backwards.

The punch clutch latches at 0° on the index.

Punch Operation

The punch drive cams consist of two pairs of complementary cams — 1 and 4, 2 and 3. They furnish a reciprocating motion for the two punch drive arms. The drive arms are pivoted about the drive-arm pivot shaft on their right ends. The punch levers (one for

each punch) are mounted on the punch bail shaft fastened between the two drive arms. When the punch bail shaft is moved by the reciprocating action of the drive arms, one of the two ends of the punch levers must also move (Figure 13-15). If the right end of the lever is free to move, the right end rather than the left end will move because the springs from the levers to the punch lever stop and spring support will tend to hold the left end stationary. However, if the right end should be held stationary by the step on the punch lever latch (the latch having been released because of the attraction of its punch magnet armature), the left end will be forced to move against its spring tension. Because the punch is attached to the left end of the punch lever, it will be driven forward each time the left end of the punch lever is driven forward (Figure 13-16). In this way the tape is punched in different combinations by energizing various punch magnets.

The fourth punch up from the bottom of the die (small punch) is the tape feed punch. This punch is driven through the tape each cycle to punch the hole for the pins on the pinwheel assembly for tape feeding. The punch lever for this position has no punch lever

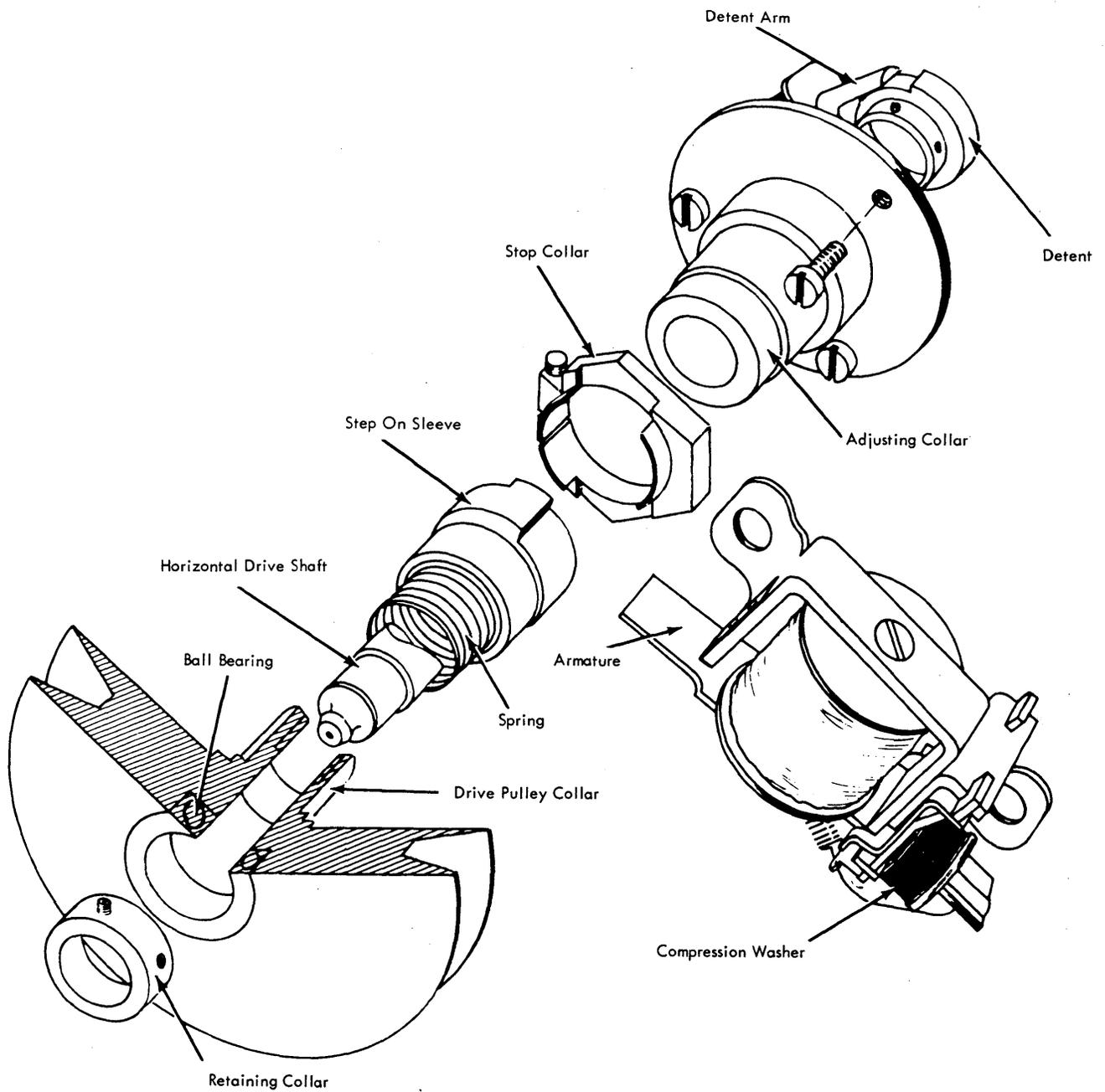


Figure 13-14. Punch Clutch

latch, but rather, a metal stop arm fastened to the latch comb and stop assembly (Figure 13-17). This stop arm acts to hold the right end of the feed punch lever in a fixed position, thus allowing the punch to perforate the tape once each cycle. This hole is for feeding purposes only.

The punches are slotted to allow insertion of the punch levers. They are held in the punch guide to eliminate any wobbling that would cause binding as

the punches are driven into the die. The motion of the punch levers is such that the punches are driven far enough into the die to cause a clean hole to be cut. The section of the punch guide adjacent to the die strips the paper from the punches as they return to normal and acts as a second guide comb. All punches have beveled ends to aid in cutting a clean hole in the tape. A chad chute fastened to the punch die directs the chads, or punchings, into the chad box.

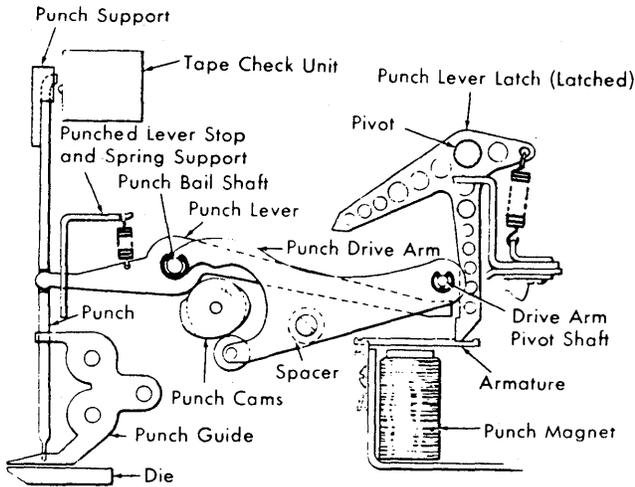


Figure 13-15. Punching Mechanism (at Rest)

Latch Lock

The latch lock mechanism is operated by cam 5. Its purpose is to lock the punch lever latches during the punching portion of the cycle. After the lever latches have been tripped for each punch code, the latch lock is cammed into the path of the left end of the latches (Figure 13-18). The tripped latches are locked in the tripped position. The latch lock also locks the untripped latches to prevent late or extra latches from being tripped after about 20° of the punch cycle. Just before the punches restore, the latch lock is cammed back to normal allowing the latches to be relatched on their armatures.

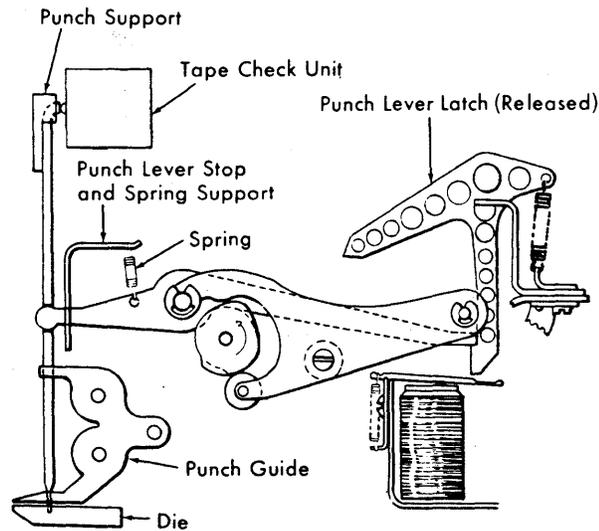


Figure 13-16. Punching Mechanism (Operated)

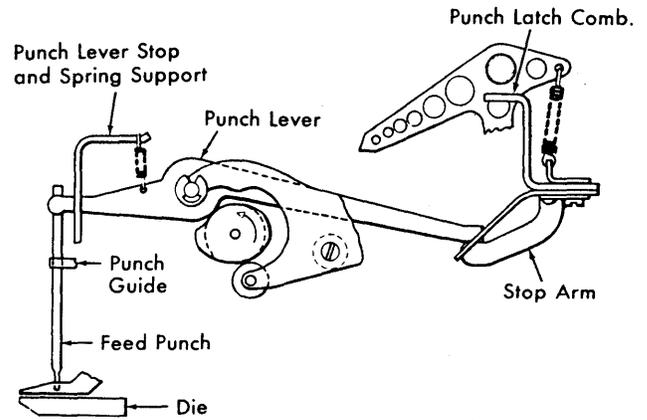


Figure 13-17. Feed Punch

Punch Lever Latch and Magnet Armature Restoration

The restoration of the punch lever latches is done by a roller stud mounted between cams 5 and 6 (Figure 13-19). After the punching operation has been completed, this roller stud contacts the latch restoring lever, causing it to pivot and operate against the punch lever latches. This action against the latches causes them to pivot counterclockwise, and to restore past the latching point on the armatures.

The restoration of the inner (bottom) lever latch causes restoration of all the punch magnet armatures. As the inner level latch is restored to the right, the right edge of the latch comes in contact with the armature knockoff lever, causing it to rotate clockwise and to knock the armatures away from the cores. There are two styles of magnet armatures and coils; a right and a left. They are not interchangeable.

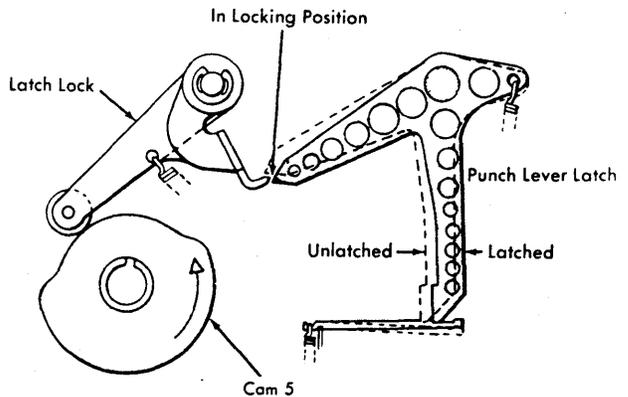


Figure 13-18. Latch Lock

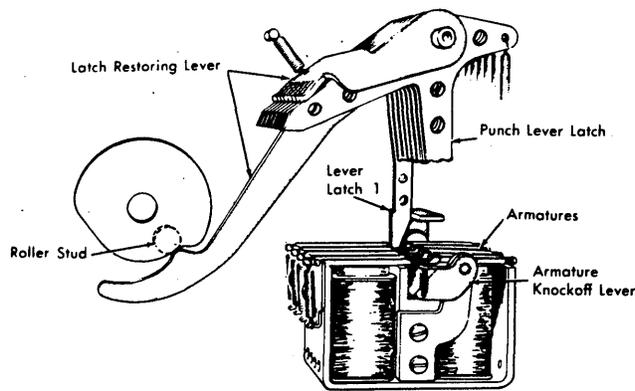


Figure 13-19. Punch Lever Latch and Magnet Armature Restoration.

Tape Feed

Cam 6 is used to feed tape after punching one column for each code in the tape under normal operation. After the punches have been withdrawn from the tape, cam 6 raises the cam follower on the tape feed arm (Figure 13-20). The arm pivots counterclockwise and pulls the tape feed pawl into engagement with the feed ratchet and moves the ratchet pinwheel one tooth. After each advance of the feed ratchet, a detent roller holds the detent wheel and feed ratchet in position for the next punch cycle. An eccentric feed pawl stop stud limits the overthrow action of the tape feed arm and feed pawl when feeding under power. As the cam follower falls from the high dwell of the cam, the feed pawl restores to its rest position under spring tension. The tape pinwheel may be positioned in either direction by hand.

Under error conditions, the no-feed magnet is energized and its armature pivots to prevent the feed pawl from engaging the feed ratchet. This prevents feed-

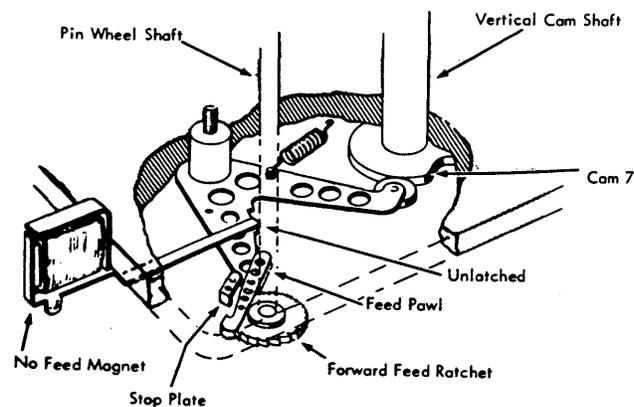


Figure 13-20. Tape Feed

ing the tape (Figure 13-21). The pinwheel assembly consists of the shaft, the manual tape feed knob, pinwheel, ratchet wheel and detent wheel (Figure 13-22). As the pinwheel is rotated, the tape is advanced, column for column.

Code Check Unit

The code check unit contains eight steel balls, eight sliding plungers, and 32 contact wires. Extending through each plunger are two sets of contact wires that provide tension of the plungers against the ball (as well as being the operating points of electrical transfer contacts).

Each punch (except the feed punch) has an extension with a turned-over end that contacts a ball and moves it into the unit whenever a punch is driven toward the tape (Figure 13-23). The ball pushes the plunger and it, in turn, transfers the contact wires. The punches that are not actuated do not transfer their contact wires. An electrical path is set up to check for an odd code. If an odd number of plungers is moved, the machine continues normally. However, if an even number of plungers is moved, an error condition exists and the write check trigger is turned on.

Tape-Feed-Pressure Arm and Adjustable Tape Guide

The tape-feed-pressure arm is used to hold the tape against the pinwheel and to guide the tape as it is

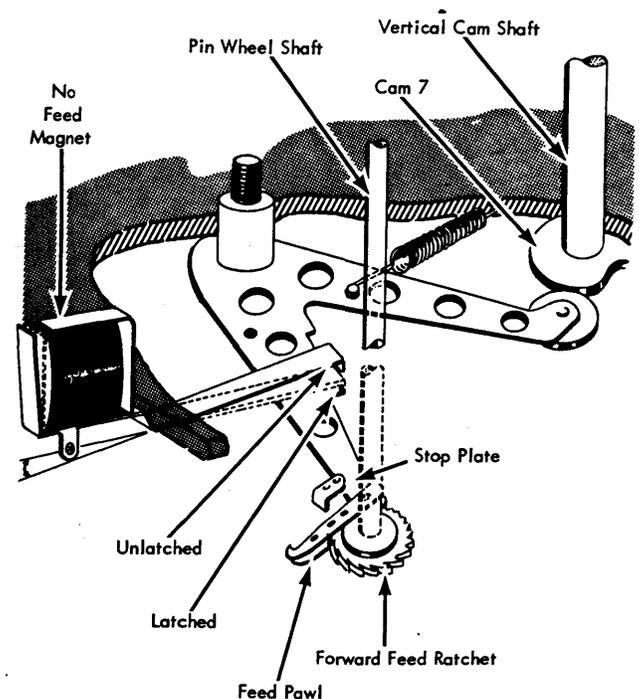


Figure 13-21. No-Feed Magnet

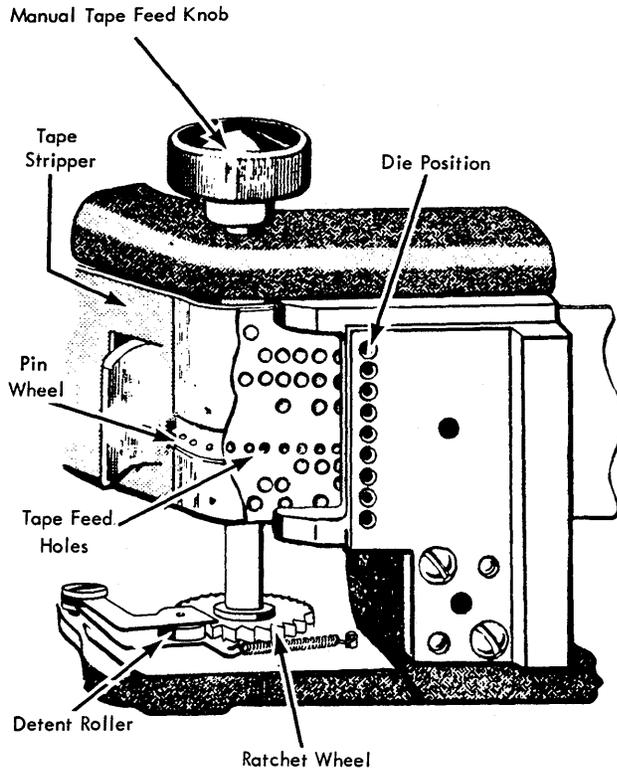


Figure 13-22. Pin Wheel Assembly

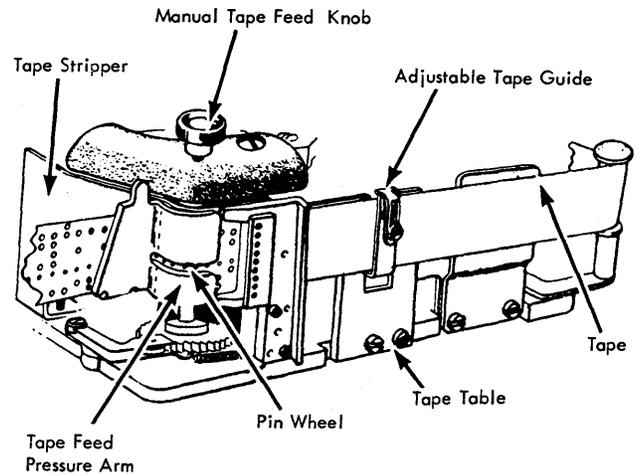


Figure 13-24. Tape Feed Pressure Arm and Adjustable Tape Guide

being advanced (Figure 13-24). As the tape is fed, a tape stripper and guide on the back of the tape-feed-pressure arm causes the tape to leave the surface of the pinwheel and advance to the rewind plate.

An adjustable tape guide is located ahead of the die position and assists in proper feeding of the tape. When the tape-feed-pressure arm is opened, or when there is no tape passing in back of the adjustable tape guide, the tape tension contact (Figure 13-25) closes. Closure of the contact turns on the no-feed trigger and the punch-no-feed lamp.

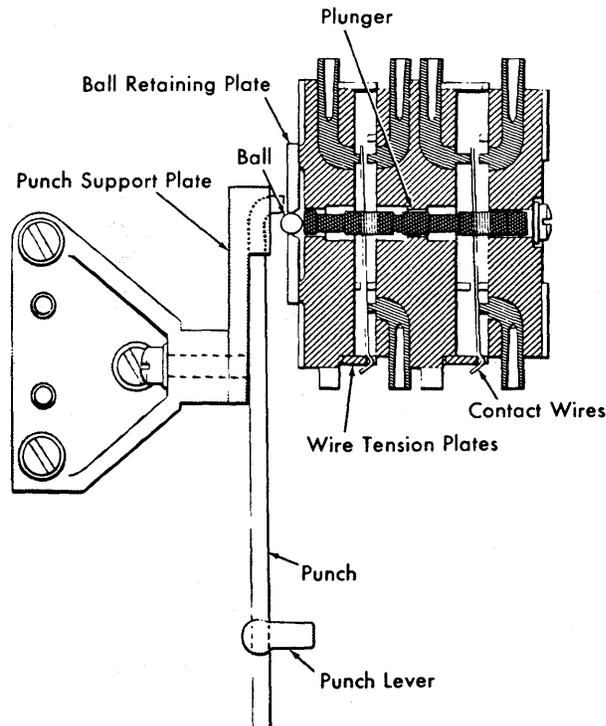


Figure 13-23. Code Check Unit

Punch-Magnet-Yoke Assembly

The punch-magnet-yoke assembly consists of eight punch magnets and armatures. As the character is read, the corresponding code is determined and the proper punch magnets are energized. Energization of the punch magnets allows the lever latches to pivot and move under the punch levers for those positions. At the same time, the action of the punch drive arms punches the holes for a code in the tape.

Sequence of Operation

When the tape punch clutch is latched, the driver pulley is the only rotating member of the punch mechanism. The punch lever latches are latched on their respective armatures, the tape feed pawl is held away from the feed ratchet, the latch lock and the lever latch restoring and armature knockoff mechanisms are at rest. The punch drive arms are at rest. The punch drive arms are toward the rear.

As soon as the tape punch clutch magnet is energized, the action of the clutch spring and the clutch sleeve immediately transmits the drive pulley motion

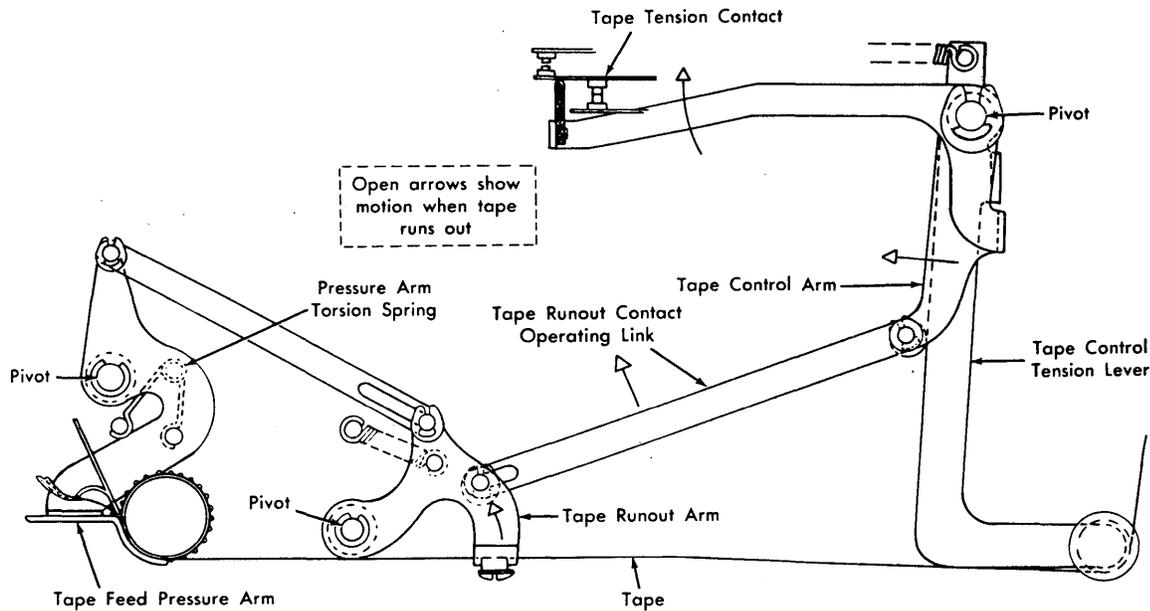


Figure 13-25. Tape Tension and Runout Linkage

to the cam shafts. At the same time that the clutch is energized, the punch magnets are energized for the positions to be punched, releasing the lever latches and allowing them to move under the right end of their corresponding punch levers. At about 20°, the latch lock is positioned so that it holds all lever latches in their latched or unlatched positions. Following the locking of the punch lever latches, the punch drive arms are operated by cams 2 and 3. The punch levers that are held stationary on the latching end because their corresponding lever latches have been released, are driven on the punch end by the punch drive arms. This causes punching in the tape for those positions (Figure 13-26). Those punch levers that are not held stationary by their corresponding lever latches are driven on the latching end rather than the punching end. No holes are punched in the tape for those positions. Each punch, in moving forward, transfers two pairs of contact wires in the code check unit when the punch depresses its corresponding ball and plunger. Punches and punch levers are then restored to normal by the action of cams 1 and 4 in restoring the punch drive arms. The latch lock is cammed away from the lever latches at the same time that the punches are being restored.

Immediately after the punches and latch lock mechanism have been restored to normal, cam 6 starts to operate the tape feed arm. This action engages the tape feed pawl in the feed ratchet and rotates the ratchet one tooth. This spaces the tape to the next

column and places it in position for the next punching operation. While the tape feed mechanism is in operation, the stud between cams 5 and 6 acts against the latch restoring lever to cause the punch lever latches to pivot counterclockwise and restore past their latching point on the punch magnet armatures. During this latch-restoring operation, the inner punch lever latch acts upon the armature knockoff lever to knock any sticking armatures away from the magnet cores so that they will be in position to allow the lever latches to latch properly.

Tape Supply Disk and Rewind Reel Assembly

The tape supply disk is located on top of the unit. A blank roll of tape is inserted and, after the tape has been threaded properly through the punching station, it is fed to the rewind plate assembly. This assembly is mounted on a bracket to the left of the punching unit and is driven by a spring belt from the horizontal cam shaft. The rewind assembly spring belt drive allows slippage and prevents the tape from winding with excessive tension or possible tape breakage.

Tape Feed Switch

Operation of the tape feed switch energizes the punch clutch and punch magnets 1 through 7 (tape feed code). The tape feed code will continue to be punched as long as the key is held operated.

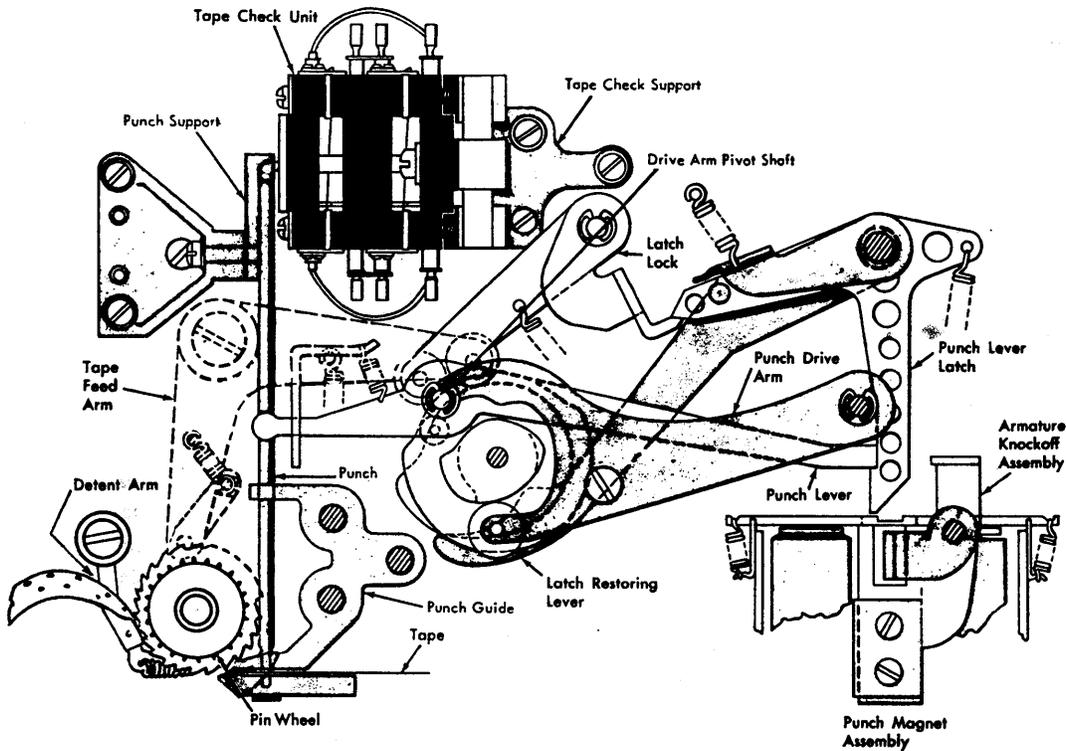


Figure 13-26. Sequence of Operation

If the no-feed trigger is ON due to an output error, depression of the feed switch will turn OFF the no-feed trigger as well as punch the tape feed code.

A function chart of the tape feed switch and its associated circuitry is included in the *Instructional System Diagrams*, page 42.

AUXILIARY TRIGGERS

Tape Feed Trigger. (01.84.10.1).

1. Turned ON by depression of tape feed switch to

energize the punch clutch and all punch magnets except EOL.

2. Turned OFF at PCB-3 time when tape feed switch is released.

Tape Feed Interlock Trigger. (01.84.10.1).

1. Turned ON by tape feed trigger to prevent a response from cycling the computer when PCB-3 makes.

General

The input devices for the IBM 1620 Data Processing System are the IBM 1621 Paper Tape Reader, the IBM 1622 Card Read Punch, and the console typewriter. These devices are used to enter (read) instructions and data into memory.

The output devices are the IBM 1624 Paper Tape Punch, the IBM 1622 Card Read Punch, and the console typewriter. These devices are used to record (write) data from memory.

The Q_s and Q_w digits of read and write instructions are used to specify the desired input-output device with two-digit addresses assigned as follows:

- 01 – Typewriter
- 02 – Paper tape punch
- 03 – Paper tape reader
- 04 – Card Punch
- 05 – Card Reader

Only one device may be selected at a time, and the device remains selected until the execution of the read or write instruction is terminated. If an invalid address is specified for an input or output device, or if the selected device is not in a ready condition to operate, the computer suspends operation in the automatic mode. (When the typewriter has been selected as an input device, the computer appears to suspend operation, but actually is awaiting the manual entry of information from the keyboard.)

Each character of data read from an input device or written on an output device is checked for odd parity. The read check indicator is turned ON if any character of an input record has incorrect parity. The write check indicator is turned ON if any character of an output record has incorrect parity.

The read and write functions for the console typewriter, paper tape punch, and paper tape reader are given in this manual. A description of the IBM 1622 Card Read Punch including the card read and write functions is given in the *IBM Customer Engineering Manual of Instruction, 1620 Data Processing System, Additional Features* (Form 227-5513).

Function charts for 1620 operation codes are located in the *IBM Customer Engineering Instructional System Diagrams, 1620 Data Processing System* (Form 227-5631).

Typewriter

The console typewriter can be used to enter (read) instructions and data into memory or to print out (write) data from memory. A record of input data is obtained as a byproduct of an input operation. The rate of entry of information from the typewriter keyboard depends upon the speed of the operator; the rate of automatic print out of data is 10 characters per second. The typewriter cannot be used off-line because the keyboard is locked when not in use to enter data into memory.

Depression of INSERT on the console activates the typewriter keyboard to permit the direct entry of instructions into memory, starting at location 00000. Each depression of a typewriter key enters that character into memory one location higher than the previous character. The keyboard remains activated until the RELEASE key on the console or the R-S key on the typewriter is depressed.

Flags for sign and field definition in numerical data are indicated on the typewriter by depressing the flag key, before depressing the digit key to which the flag refers.

During an input operation, characters are presented to computer input translator circuitry from the keyboard for conversion to the six-bit digit code (see Appendix B). If a character with incorrect parity is presented to the translator, the read check indicator is turned ON; however, the computer remains in the automatic mode and continues to accept data from the keyboard until RELEASE or the R-S key is depressed. The computer then enters the manual mode. The read check indicator can be interrogated by a branch indicator or branch no indicator instruction and will be turned OFF, if ON, by the execution of that instruction.

During an output operation, characters from memory in the six-digit code are presented to computer output translator circuitry for conversion to typewriter code (See Appendix B). If an invalid character is presented to the typewriter from the translator, an invalid character symbol \mathcal{K} is printed. If a character with incorrect parity is presented to the typewriter, the write check indicator is turned ON and the character is centerscored. The write check indicator is turned on if a typebar fails to mechanically respond to a character in the typewriter encoding (bit) relays. The setting of the I/O check switch on the console to

STOP or PROGRAM determines whether the computer stops upon completion of writing the record in which the parity error was detected or continues with the program. The write check indicator can be interrogated by a branch indicator or branch no indicator instruction and is turned OFF, if ON, by the execution of that instruction.

Space, carriage return, and tabulate are typewriter functions that may be inserted in a program by means of Control-Code 34 instructions.

Paper Tape

Instructions and/or data are read or written (punched) as holes in one-inch-wide paper tape which contains eight channels. Information is recorded at a density of ten characters to the inch. Figure 14-1 is a schematic of a section of paper tape illustrating the eight channels and the coded punching for alphabetic and numerical characters.

The four lower channels of the tape (excluding the feed hole) are used to record numerical characters 1 through 9. For example, a hole in the "1" channel represents a numerical 1; a hole in the "2" channel a numerical 2; and holes in the "1" and "2" channels of the same tape column a numerical 3. A zero is recorded as a hole in the "0" channel. A flag (F) bit in numerical data for sign or field definition is recorded as a hole in the "X" channel of the tape column which contains the numerical digit to which the flag refers.

The "X" and the "0" channels are used in combination with the numerical channels to record alphabetic characters, similar to zone punching in IBM cards. The 12-zone alphabetic characters (A-I) require both the "X" and "0" channel holes in combination with the numerical digit punching, while alphabetic characters J-R use the "X" channel hole in combination with the numerical digit punching, and the alphabetic char-

acters S-Z use the "0" channel hole in combination with the numerical digit punching.

Special characters are recorded by the use of combinations of numerical channels not required for numerical characters, combinations of the "X" and/or "0" channels with numerical channels not required for numerical and alphabetic characters, and the "X" channel alone (See Appendix B).

As a check that each character is recorded correctly, each column of the tape is punched with an odd number of holes. A check, "C" channel, hole must be added in any column where the basic character code (X, 0, 8, 4, 2, 1) consists of an even number of holes. (In numerical data if the "X" channel hole representing a flag bit for sign or field definition causes the number of holes in a column to be an odd number, a "C" channel hole will not be added.)

The "C" channel hole alone represents a space when an alphameric read or write instruction is being executed.

The paper tape code for all numerical, alphabetic, and special characters is illustrated in Figure 14-2.

The EL (end-of-line) channel of the tape is associated with termination of data transmission to or from memory. When the paper tape reader senses a column of the tape which contains a hole in the EL channel, a record mark is placed in memory and the read operation is terminated. When a record mark is sensed from memory during a write instruction with the paper tape punch as the selected device, a hole is punched in the EL channel of the tape column following that occupied by the last character transmitted from memory, and the write instruction is terminated.

Tape feed codes (holes in channels 1 through 7 of each column) are punched repetitively when TAPE FEED is held depressed. They are used to cause the tape to feed in the paper tape reader during insertion and run-out.

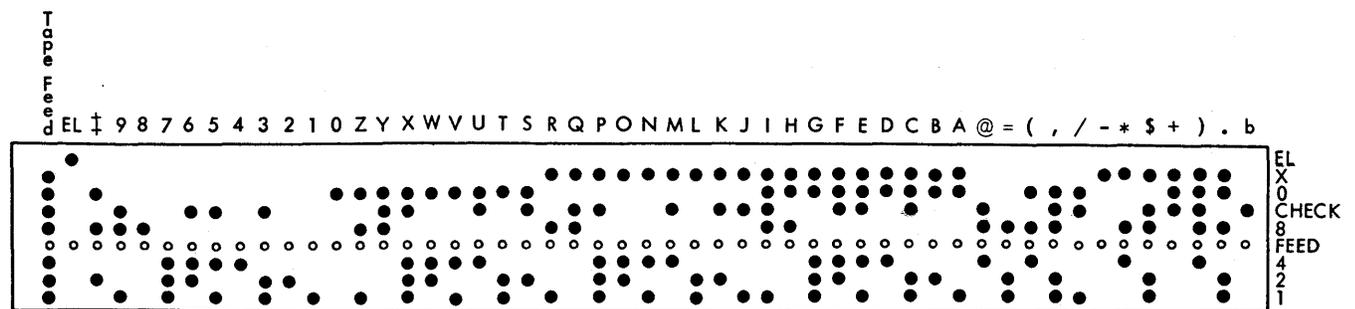


Figure 14-1. Paper Tape

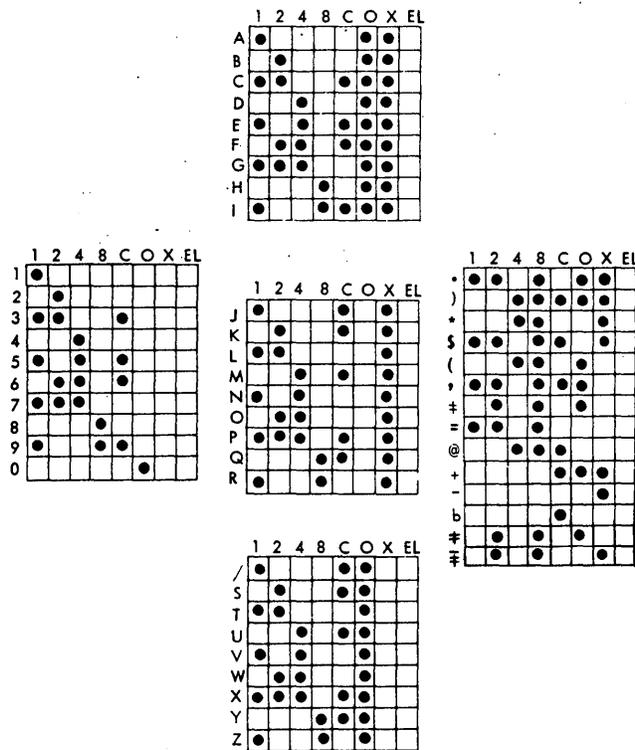


Figure 14-2. Paper Tape Code

IBM 1621 Paper Tape Reader

Characters are read serially by the paper tape reader at the rate of 150 characters per second and presented to computer input translator circuitry for conversion to the six-bit digit code (see Appendix B). If a character with incorrect parity is presented to the translator, the read check indicator is turned ON; however, the computer remains in the automatic mode and the read operation continues. Sensing of a hole in the EL channel of a tape column causes a record mark to be placed in memory and the read operation to be terminated. The setting of the I/O check switch on the console to STOP or PROGRAM determines whether the computer stops upon termination of the read operation during which the parity error was detected, or continues with the program. The read check indicator can be interrogated by a branch indicator or branch no indicator instruction and is turned OFF, if ON, by the execution of that instruction.

IBM 1624 Paper Tape Punch

On a write instruction with the paper tape punch as the selected output device, data from memory is presented to computer output translator circuitry for conversion to paper tape code. Characters are punched

into the tape at a rate of 15 per second. When a record mark is sensed from memory, a hole is punched in the EL channel of the tape and the write operation is terminated.

Transmittal of an invalid character or punching an even number of holes turns on the no feed and write check triggers. The run trigger is turned off, the tape feed does not advance, and the computer stops in the automatic mode without regard to the setting of the I/O check switch on the console. The punch no feed, manual, and write check lights are turned on.

When a valid character is transmitted but punches incorrectly, program processing can be resumed with the following procedure:

1. Operate the tape feed switch
 - a. The tape feed code is overpunched on the incorrectly punched character.
 - b. The write check and no feed trigger are turned off thereby turning off the write check and no feed lights.

2. Depress the start key

The run trigger is turned on and the computer attempts to repunch the same valid character that was previously punched incorrectly. If the attempt is successful the computer proceeds with the program. If the attempt is not successful the error will again stop the computer as previously explained.

If the error is caused by transmittal of an invalid character or if the program processing resumption procedure does not correct the error, the console reset key must be depressed to return the machine to the manual mode.

If the 1624 runs out of paper tape, the no-feed trigger is turned on, the computer stops in the automatic mode, and the no feed and manual lights turn on. Program processing can be resumed by following the procedure given previously.

Read Numerically (Code 36 — RN)

Sequence block diagrams for this operation are shown in Figures 14-3 and 14-4. Function charts for this operation are shown in the *Instructional System Diagrams*, pages 34 and 35. A sequence chart of the typewriter used as the input device (numerical) is shown in Figure 14-5.

Objectives

1. Read numerical information, including flag bits, into memory from the input device specified by the Q_8 and Q_9 digits of the instruction.

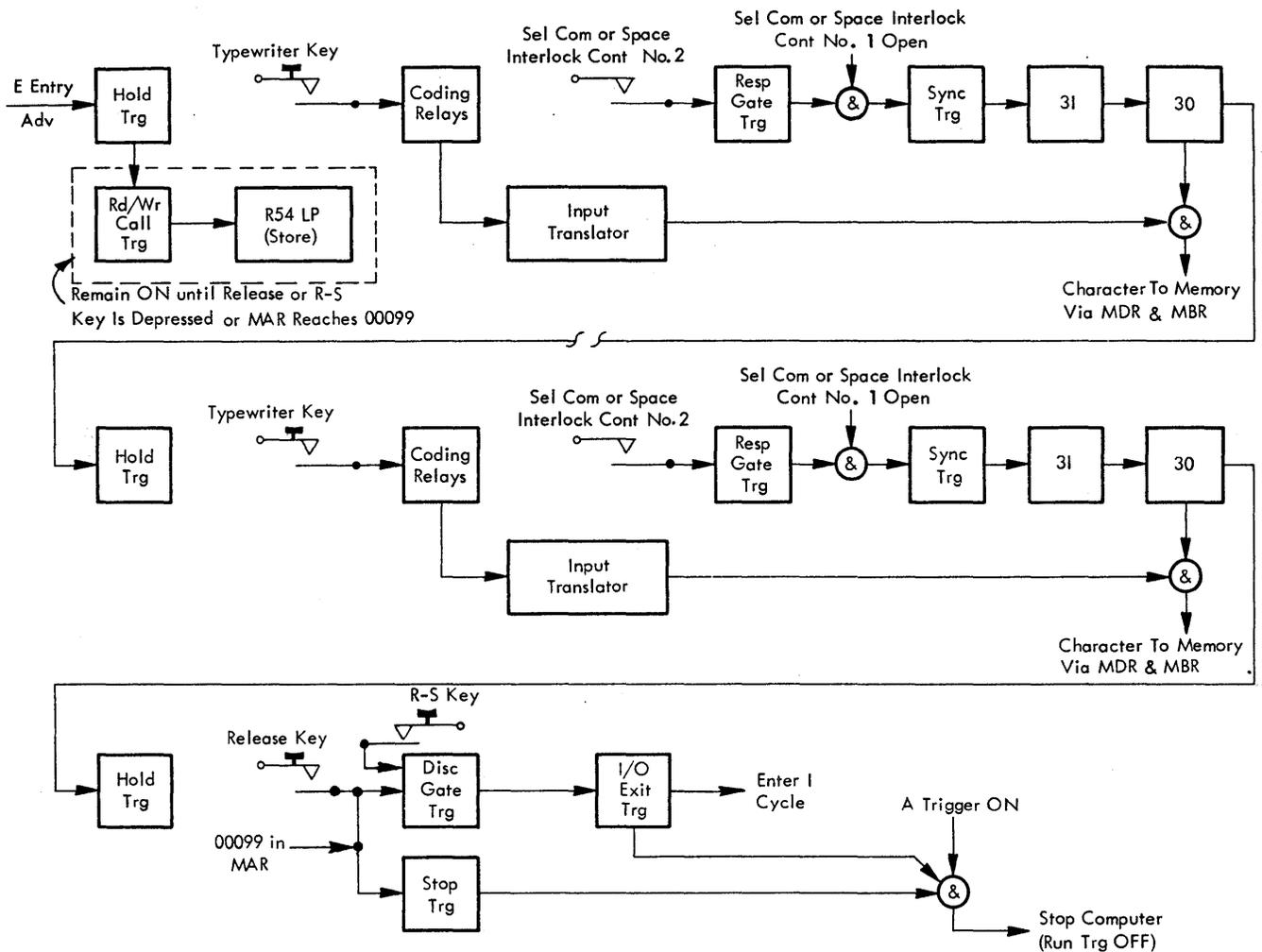


Figure 14-3. Read Numerically—Code 36 (36 P P P P P Q 0 1 Q Q)

2. Store the information at the memory location designated by the P address (OR-2) and successively higher memory locations.

Functions

During the I-cycle, the Q_8 and Q_9 digits of the instruction are placed in the digit/branch register and decoded to specify the input device. The typewriter keyboard is specified by 01 and the paper tape reader by 03.

The digits of input data are transmitted serially to memory at the location designated by the P address (OR-2) and successively higher memory locations. Flag bits with digits of input data are duplicated in memory with the digits.

Transmission of a numerical character of input data from the typewriter keyboard requires energization of the coding relay (or relays) corresponding to the typewriter character code for the specific digit. Coding

relays are assigned as follows:

Bit	Coding Relay
0	R56
1	R57
2	R58
4	R59
8	R60

Depression of the typewriter keyboard flag key turns ON the flag trigger, which provides for storage of an F bit with the digit that follows. C bits are developed as needed for odd parity.

Transmission of input data from the paper tape reader is accomplished by supplying the output of the read head photo cell for each of the 8 tape channels to the input translator. Tape feed codes (holes in channels 1 through 7 of a tape column) cause the

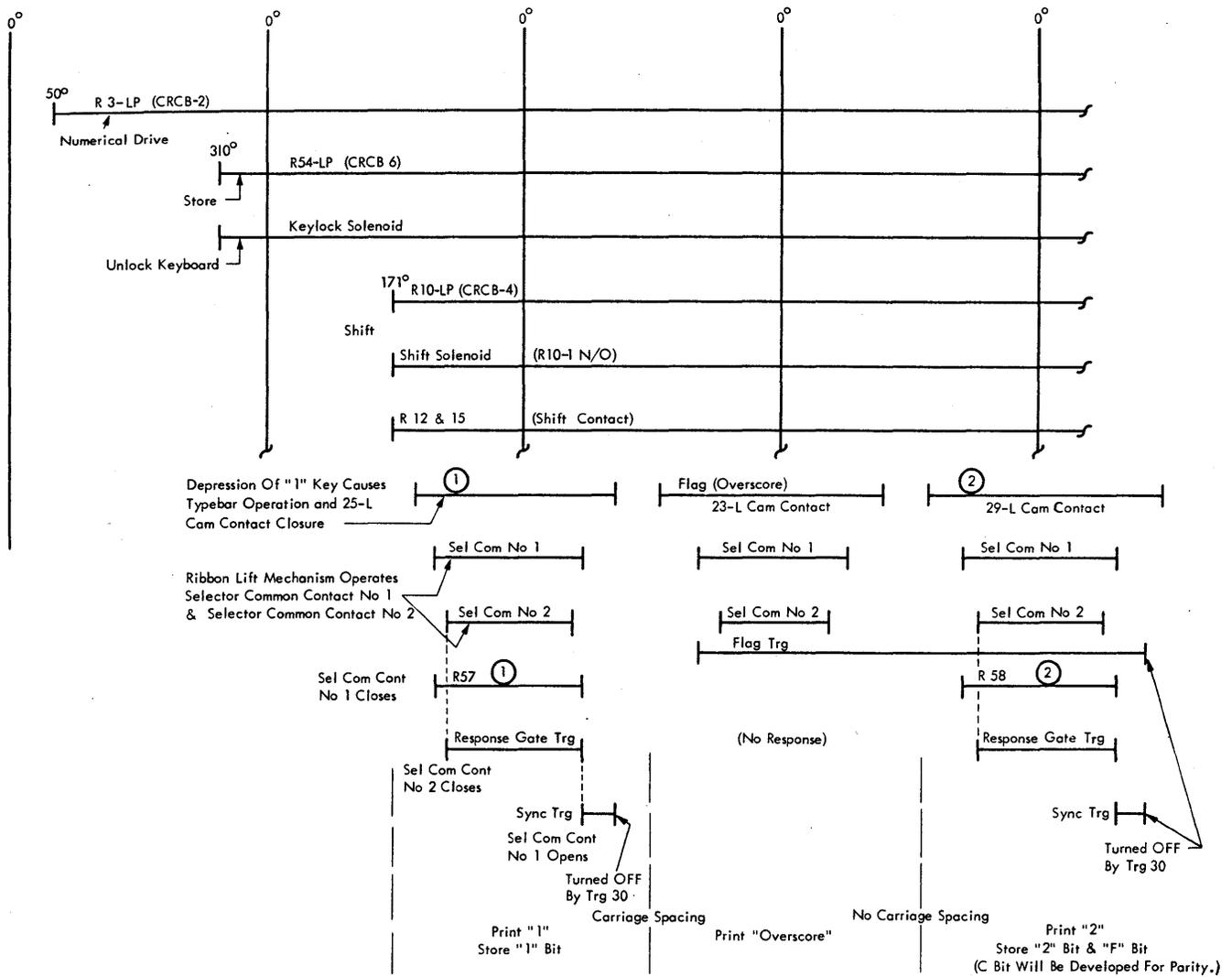


Figure 14-5. Typewriter Used as Input Device (Numerically)

This performs the same function as RELEASE.

A record mark is not automatically placed in memory if any of the three preceding items terminate the read operation. If a record mark is wanted, the record mark key must be depressed before RELEASE or the R-S key is used to terminate the read operation. In the case of MAR reaching 00099, a record mark cannot be placed in memory because the typewriter keyboard is inactivated.

NOTE: Depression of the R-S key blocks the response gate trigger to prevent the computer from cycling. The flag trigger is turned on when the R-S key is depressed; however, it has no function and is turned off by the i/o exit trigger.

Auxiliary Trigger Status

Decrement Trigger. (01.60.05.1).

1. Turned OFF (increment) when I-cycle is entered.
2. Remains OFF throughout E-cycle.

Read-Write Call Trigger. (01.64.14.1).

1. OFF when E-cycle is entered.
2. Turned ON by the hold trigger and remains ON until the read operation is terminated.
 - a. Unlocks typewriter keyboard.
 - b. Picks R54, the store relay.
3. Turned off by i/o exit trigger.

Response Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned ON by a signal from the input device that a character is available to the computer.
3. Turned OFF by the sync trigger.

Sync Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON during hold trigger time, when the input device response signal terminates, to signal the computer to proceed with storing the character in memory.
3. Turned OFF during trigger 30 time.

Disconnect Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned ON by release key or R-S key depression, MAR reaching 00099, or a tape level 8 (end-of-line) indication to signal termination of read operation.
3. Turned OFF by I/O exit trigger.

I/O Exit Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON when termination of read operation has been signalled by release key or R-S key depression, MAR reaching 00099, or end-of-line indication.
3. Turned OFF during trigger 1 time of following I-cycle.

Stop 1 Trigger. (01.06.11.1).

1. OFF when E-cycle is entered.
2. Turned ON by depression of the release key or MAR reaching 00099.
3. Turned OFF when run trigger goes OFF.

Response Check Latch. (01.82.30.1).

1. Turned ON during trigger 31 time.
2. Turned OFF on the next input cycle after having been turned on by a response signal from the input device. The response signal indicates that a character is available to the computer.
3. Turned OFF by the I/O Exit trigger ANDed with Read Call Gate. When selector common contact No. 2 or space interlock contact No. 2 makes, it turns off the response check latch and turns on the response gate trigger. When selector common contact No. 1 or space interlock contact No. 1 breaks, the sync trigger is turned on and, in parallel, a test is made of the status of the response check latch. If the latch is on, the write check trigger is turned on.

NOTE: The response check latch and its circuits are primarily designed for checking a write operation with

the typewriter designated as the output device. It is included here because it is turned on and off each cycle and will detect extra computer cycles during the read operation. Extra computer cycles could be caused by an extraneous noise spike turning on the response gate trigger, the sync trigger, trigger 31, trigger 30, and hold trigger. In this case the response check latch is on from the previous cycle and causes the read check latch to be turned on in parallel with the sync trigger, thereby indicating a read error.

Read Check Trigger. (01.81.45.1).

1. Turned ON during hold trigger time in parallel with the sync trigger if the response check latch is on.
2. Turned ON during trigger 30 time if a character with incorrect parity is presented to the input translator.
3. Turned OFF when interrogated by a branch indicator or branch no indicator instruction or when manually reset.

Flag Trigger. (01.81.10.1).

1. OFF when E-cycle is entered.
2. Turned ON by depression of the flag key (or R-S key) on the typewriter.
3. Turned OFF during trigger 30 time.

E-Timer Trigger Objectives

Hold Trigger. (01.64.12.1).

1. Turn on read-write call trigger.
2. Block MAR reset to prevent VRC error.
3. Stop the computer clock to prevent unnecessary cycling of memory.

Trigger 31. (01.64.11.1).

1. Read out of memory per OR-2.
2. Write back OR-2 bypassed.
3. Turn ON the response check latch.
(Trigger 31 is used as a dummy E-timer trigger in Code 36 - Read Numerically operation. See Code 37-Read Alphanumerically for Trigger 31 function.)

Trigger 30. (01.64.10.1).

1. Read out of memory per OR-2 with either the odd or even sense amplifiers blocked, depending on whether the OR-2 address is odd or even, to clear the memory location.
2. Write back OR-2 incremented plus 1.
3. Set input character into MDR.
4. Transfer MDR to MBR.
5. Write into memory per OR-2 from MBR.
6. Turn ON hold trigger.

Read Alphanumerically (Code 37 — RA)

Sequence block diagrams for this operation are shown in Figures 14-3 and 14-4. Sequence block diagrams for Read Numerically - Code 36 and Read Alphanumerically - Code 37 differ only in that, for Code 37 operations, the input translator supplies a digit (zone) to MBR-even as well as a digit to MDR. The function chart for the alphanumeric part (trigger 30) of this operation is shown in the *Instructional System Diagrams*, page 38. The remainder of trigger functions for this operation is shown in the *Instructional System Diagrams*, pages 34 and 35. The input translator function charts are shown in the *Instructional System Diagrams*, pages 36 and 37.

Objectives

1. With the computer in alphanumeric mode, read information into memory from the input device specified by the Q_8 and Q_9 digits of the instruction.
2. Store the information as two adjacent digits at the memory locations designated by the P address minus one and the P address (OR-2) and successively higher pairs of memory locations.

Functions

During the I-cycle, the Q_8 and Q_9 digits of the instruction are placed in the digit/branch register and decoded to specify the input device. The typewriter keyboard is specified by 01 and the paper tape reader by 03.

Characters of input data are transmitted serially. Each character is stored in memory as two adjacent digits beginning at the memory locations designated by the P address minus one and the P address (OR-2) and continuing with successively higher pairs of memory locations. The P address must designate an odd numbered memory location for storage of the numerical digit of the first character. The zone digit of the first character is placed at the next lower memory address, which is an even address. Increment plus two is used to provide memory addresses for successive characters.

If an even numbered memory location is erroneously designated by the P address, input data is not correctly placed in memory and parity errors may occur.

All flag bits existing in memory locations to which input data is being transmitted remain unchanged; flag bits are not allowed on the input data. Data from the input device may consist of a random mixture of numerical, alphabetic, and special characters.

Transmission of a character of input data from the typewriter keyboard requires energization of the coding relay (or relays) corresponding to the typewriter character code for the specific character. Coding relays are assigned as follows:

Bit	Coding Relay
X	R55
0	R56
1	R57
2	R58
4	R59
8	R60

C bits are developed as needed for odd parity.

Transmission of input data from the paper tape reader is accomplished by supplying the output of the read head photocell for each of the 8 tape channels to the input translator. Tape feed codes (holes in channels 1 through 7 of a tape column) cause the tape to feed through the paper tape reader with no cycling of the computer.

The read check trigger is turned on if:

1. A character with incorrect parity is presented to the input translator.
2. The response check latch is on because the input device failed to return a response signal to the computer.

However, the computer remains in the automatic mode and continues to accept input data until the read operation is terminated. Termination of the read operation directs the computer to enter the I-cycle for the next instruction in sequence.

If the input device is the paper tape reader, the read operation is terminated by one of the following:

1. Sensing a hole in the end-of-line channel (channel 8) of the tape. (An alphanumeric record mark is automatically placed in the two memory locations next higher than that occupied by the numerical digit of the last character read from the tape. An alphanumeric record mark consists of a zero zone digit and a numerical record mark, C-8-2, as the numerical digit.)
2. Depressing RELEASE on the console or the R-S key on the typewriter keyboard.

The computer stops when the I-cycle is entered if the read operation is terminated by use of the release key. Depression of START on the console is then required to restart the computer. The R-S key performs the function of RELEASE followed by START.

If the input device is the typewriter keyboard, the read operation is terminated by one of the following:

1. Depressing RELEASE on the console.

The computer stops when the I-cycle is entered. Depression of START on the console is then required to restart the computer.

2. Depressing the R-S key on the typewriter keyboard.

This performs the same function as RELEASE followed by START.

3. The address in MAR reaching 00099.

This performs the same function as RELEASE.

A record mark is not automatically placed in memory if any of the three preceding items terminate the read operation. If a record mark is wanted, the record mark key must be depressed before RELEASE or the R-S key is used to terminate the read operation. In the case of MAR reaching 00099, a record mark cannot be placed in memory because the typewriter keyboard is inactivated.

NOTE: Depression of the R-S key blocks the response gate trigger to prevent the computer from cycling. The flag trigger is turned on when the R-S key is depressed; however, it has no function and is turned off by the I/O exit trigger.

Auxiliary Trigger Status

See Read Numerically - Code 36.

E-Timer Trigger Objectives

Hold Trigger. (01.64.12.1).

1. Turn on read-write call trigger.
2. Block MAR reset to prevent VRC error.
3. Stop the clock to prevent unnecessary cycling of memory.

Trigger 31. (01.64.11.1).

1. Read out of memory per OR-2 to set F bit triggers in MBR and MDR for retention of any F bits in the memory locations receiving input data.
2. Write back OR-2 bypassed to retain the address for clearing and writing into the memory locations during trigger 30 time.

Trigger 30. (01.64.10.1).

1. Read out of memory per OR-2 with both the odd and even sense amplifiers blocked to clear the memory locations (Block Memory SA).
2. Write back OR-2 incremented plus 2.
3. Block reset of F bit triggers in MBR-even, MBR-odd, and MDR.
4. Set MBR-even and MDR triggers per character from input device.
5. Transfer MDR to MBR-odd.
6. Write into memory per OR-2 from MBR.
7. Turn on hold trigger.

Write Numerically (Code 38 — WN)

Sequence block diagrams for this operation are shown in Figures 14-6 and 14-7. Function charts for this operation are shown in the *Instructional System Diagrams*, pages 39, 40, 41, and 42. A sequence chart of the typewriter used as the output device (numerical) is shown in Figures 14-8, 14-9, 14-10, and 14-11.

Objective

Transmit numerical information, including flag bits, from the memory location designated by the P address (OR-2) and successively higher memory locations to the output device specified by the Q₈ and Q₉ digits of the instruction.

Functions

During the I-cycle, the Q₈ and Q₉ digits of the instruction are placed in the digit/branch register and decoded to specify the output device. The typewriter is specified by 01 and the paper tape punch by 02.

The digits of output data are transmitted serially from the memory location designated by the P address (OR-2) and successively higher memory locations to the selected output device. Flag bits in memory with digits of output data are recorded with the digits by the output device.

Alphabetic characters, special characters, and numerical characters, which have been stored in memory as two adjacent digits because the computer was in alphameric mode, are recorded by the output device as two numerical digits.

The write operation leaves output data unchanged in memory.

Transmission of a numerical character of output data to the typewriter requires energization of the typewriter bit relay (or relays) corresponding to the typewriter character code for the specific digit. Bit relays are assigned as follows:

Bit	Relay
C	R21
F (X)	R22 and R25
0	R28 and R31
1	R41
2	R42 and R45
4	R46 and R49
8	R50

The presence of an F bit with a digit of output data causes the energization of R22 and R25, which provides for printing an overscore. When a C bit exists with only an F bit (FC in MDR), R28 and R31 will be picked along with R22 and R25 and this will cause an overscored zero (0̄) to be printed. A C bit, when accom-

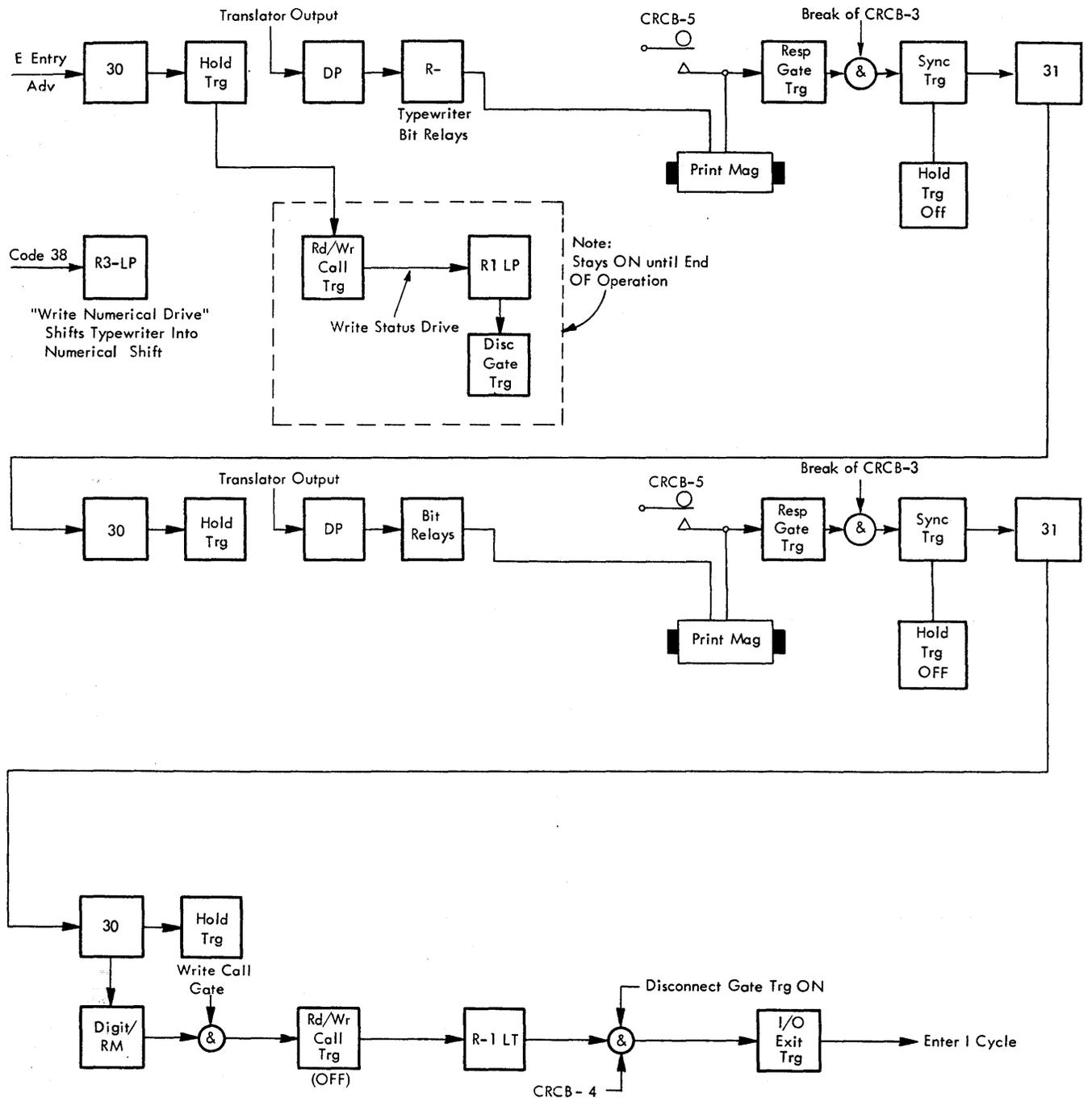


Figure 14-6. Write Numerically—Code 38 (38 P P P P P Q 0 1 Q Q)

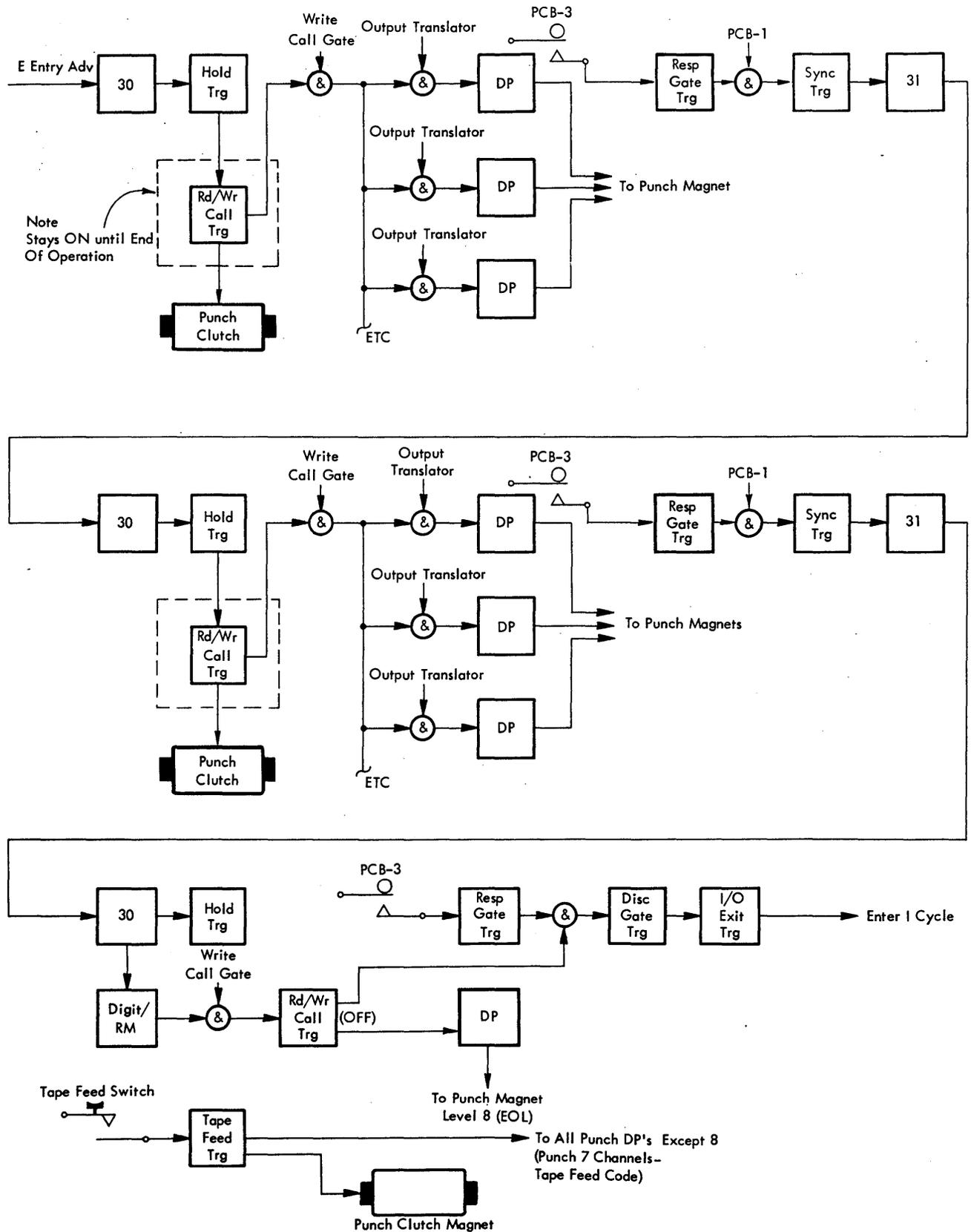


Figure 14-7. Write Numerically—Code 38 (38 P P P P P Q 0 2 Q)

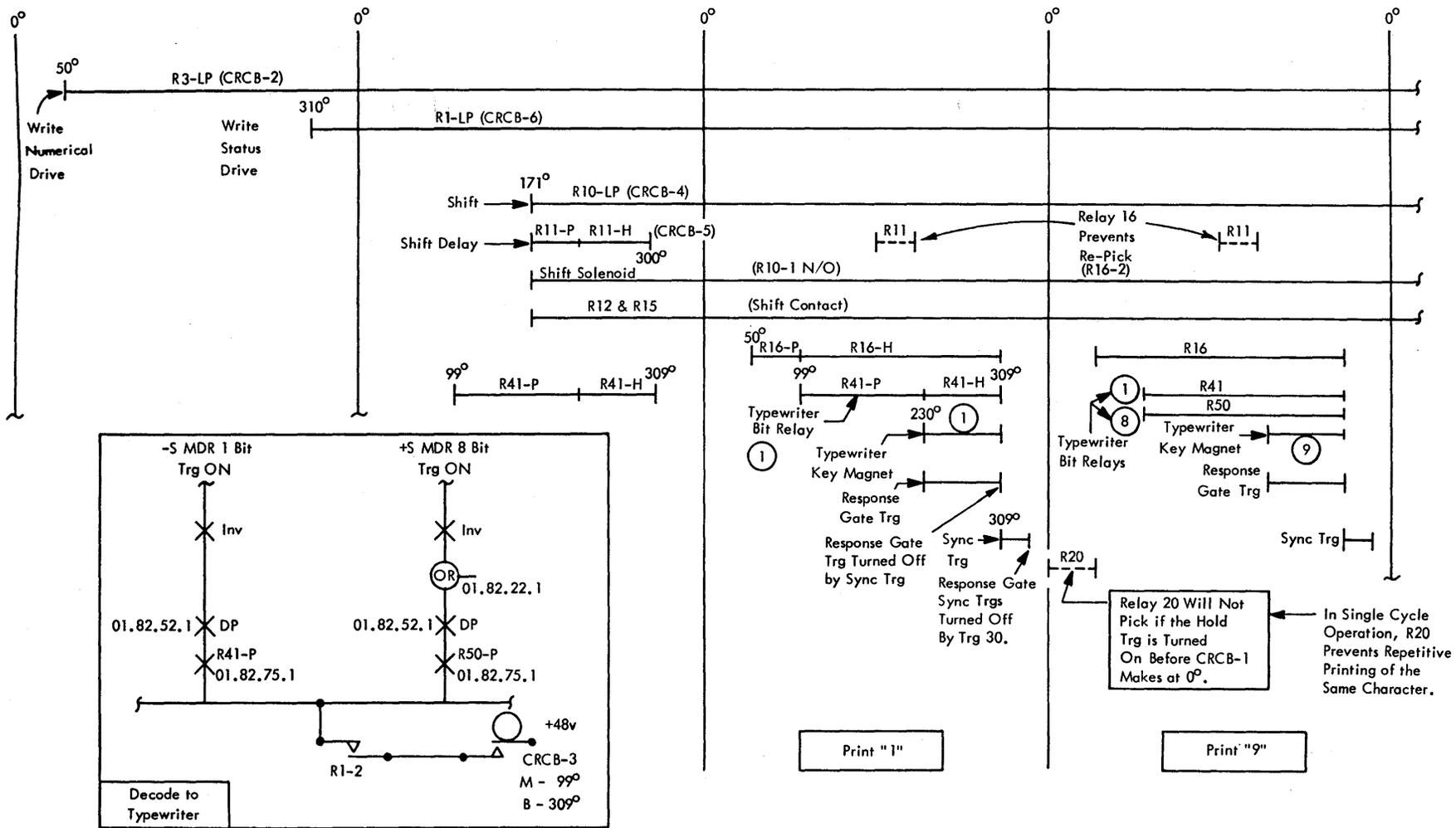


Figure 14-8. Typewriter Used as Output Device (Numerical)

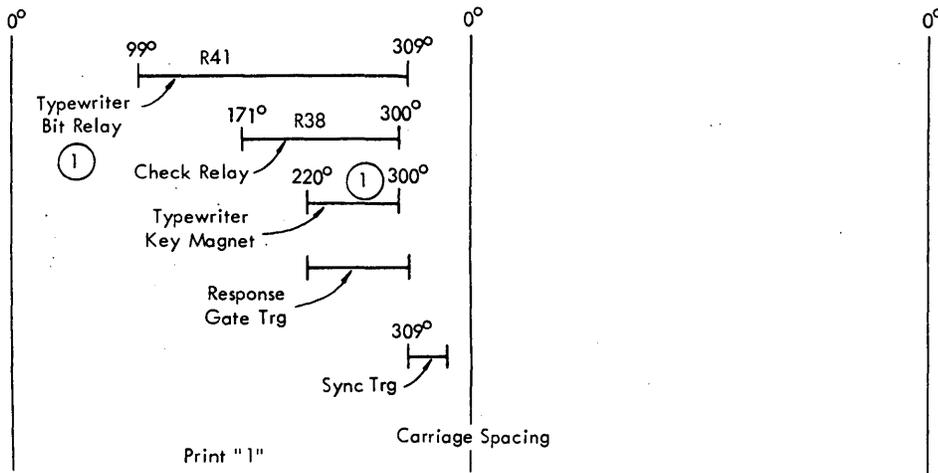


Figure 14-9. Operation with No Parity Error

panied by other bits to comprise a digit, causes energization of R21, which is used only for parity checking purposes.

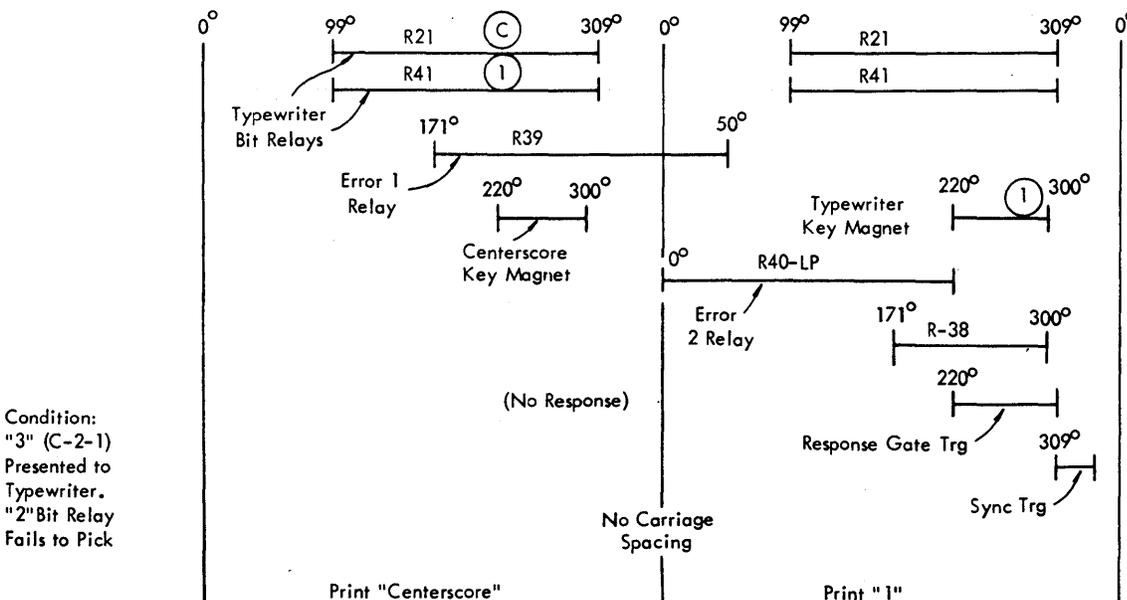
Transmission of a numerical character of output data to the paper tape punch requires energization of the punch magnet (or magnets) corresponding to the paper tape code for the specific digit.

The write check trigger is turned on when any one or more of the following conditions exists:

1. A character with incorrect parity is presented to the output device by the output translator.
2. Typewriter bit relays are energized for an even number of bits.

3. An even number of holes are punched in a tape column by the paper tape punch.
4. The response check latch is on because the output device failed to return a response signal to the computer.

If the write check trigger is turned on with the typewriter as the output device, the computer remains in the automatic mode and continues to supply data to the typewriter until the write operation is terminated. If the write check trigger is turned on with the paper tape punch as the output device, the punch is stopped immediately with the incorrectly punched tape column still under the punches.



Condition:
"3" (C-2-1)
Presented to
Typewriter.
"2" Bit Relay
Fails to Pick

Figure 14-10. Operation with Parity Error

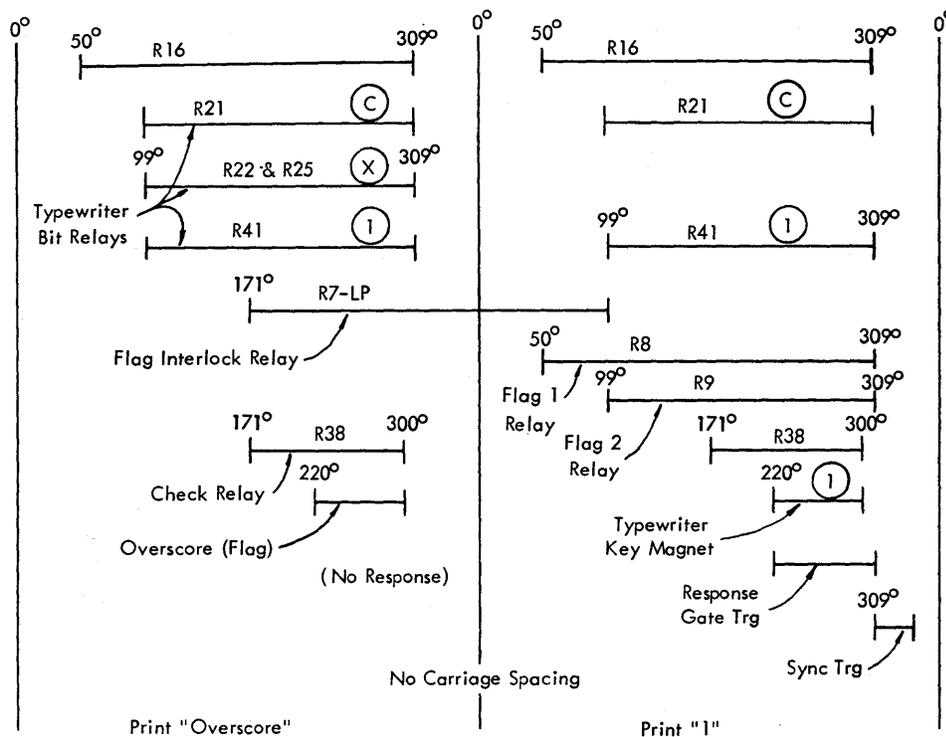


Figure 14-11. Operation to Print Flag (Overscore) with Numerical Character

With the typewriter as the output device, the write operation is terminated by one of the following:

1. Sensing a record mark from memory. Termination of the write operation directs the computer to enter the I-cycle for the next instruction in sequence. (The record mark is not printed.)
2. Depressing RELEASE on the console. The write operation is terminated immediately, and the computer stops. Depression of START on the console is required to restart the computer.

With the paper tape punch as the output device, the write operation is terminated by one of the following:

1. Sensing a record mark from memory. A hole is punched in the end-of-line channel (channel 8) of the tape in the tape column immediately following that occupied by the last output data character transmitted from memory. Termination of the write operation directs the computer to enter the I-cycle for the next instruction in sequence.
2. Depressing RELEASE on the console. The write operation is terminated immediately, and the computer stops. Depression of START on the console is then required to restart the computer.

Auxiliary Trigger Status

TYPEWRITER AND THE PAPER TAPE PUNCH TRIGGERS

Decrement Trigger. (01.60.05.1).

1. Turned OFF (increment) when I-cycle is entered.
2. Remains OFF throughout E-cycle.

Read-Write Call Trigger. (01.64.14.1).

1. OFF when E-cycle is entered.
2. Turned ON by the hold trigger and remains ON until the write operation is terminated.
3. Turned OFF during trigger 30 time when MDR contains a record mark.
4. Turned OFF by the I/O exit trigger if the release key is depressed during a write operation and the paper tape punch is the output device.
5. Turned OFF by the MEM-OF trigger ANDed with the response gate trigger if RELEASE is depressed during a write operation and the typewriter is the output device.

Response Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned ON by a signal from the output device that a character has been received from the computer.
3. Turned OFF by the sync trigger.

4. Turned OFF by the I/O exit trigger when the write operation is terminated.
5. Turned OFF by the no-feed trigger if the paper tape punch is the output device and a parity or response error occurs or the paper tape breaks.

Sync Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON during hold trigger time when CRCB-3 breaks for a typewriter operation or PCB-1 makes for a paper tape punch operation, to signal the computer to proceed with presenting the next character to the output device.
3. Turned OFF during trigger 30 time.
4. Turned OFF by the I/O exit trigger when the write operation is terminated.

Digit/Record Mark Trigger. (01.63.50.1)

1. OFF when E-cycle is entered.
2. Turned ON during trigger 30 time if MDR contains a record mark (C-8-2).
3. Turned OFF by the I-cycle trigger.

Memory Overflow Trigger. (01.64.14.1).

1. OFF when E-cycle is entered.
2. Turned ON by depression of RELEASE.
3. Turned OFF by I/O exit trigger.

Disconnect Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned on by relay 1-4 N/O point closed ANDed with the sync trigger when it is turned on by a response for the first output character if the output device is the typewriter.
3. Turned ON by the response gate trigger after the end-of-line hole has been punched, to terminate the write operation, if the output device is the paper tape punch.
4. Turned OFF by the I/O exit trigger.

I/O Exit Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON when the circuit to turn ON the disconnect gate trigger is terminated (and CRCB-4 makes in the case of the typewriter).
3. Turned OFF during trigger 1 time of the following I-cycle.

Response Check Latch. (01.82.30.1).

1. Turned ON during trigger 31 time.
2. Turned OFF during hold trigger time when selector common contact No. 2 or space interlock contact No. 2 makes.

NOTE: The status of the response check latch is tested at the same time the sync trigger is turned on. The on/off status of the response check latch at this test time was determined by the previous character. If the

previous character failed to turn off the response check latch, the write check trigger is turned on (see Figure 14-14).

Write Check Trigger. (01.81.45.1).

1. Turned ON (in parallel with the sync trigger) by the response gate trigger, if:
 - a. The output translator presents a character with incorrect parity to the output device.
 - b. The response check latch is on.
2. Turned ON when PCB-3 makes if an even number of holes is punched by the paper tape punch.
3. Turned ON when CRCB-5 makes, if an even number of typewriter coding relays are picked. (Relay 38 will not pick if an even number of coding relays are picked.)
4. Turned ON by trigger 1 of the I-cycle following a write operation, using the typewriter as the output device, if the response check latch is on. This circuit is functional only if the response failure occurs on the last character prior to sensing the record mark from memory.
5. Turned OFF by operating the tape feed switch if the paper tape punch is the selected output device.
6. Turned off when interrogated by a branch indicator or branch no indicator instruction or when manually reset.

PAPER TAPE PUNCH TRIGGERS

When the paper tape punch is the output device, the following additional triggers are used:

No Feed Trigger. (01.84.10.1).

1. OFF when E-cycle is entered.
2. Turned ON if tape tension contact closes (paper tape break).
3. Turned ON by the write check trigger (parity or response error).
4. Turned OFF by depression of tape feed key.

E-Timer Trigger Objectives

Trigger 30. (01.64.10.1).

1. Read out of memory per OR-2 and store the digit in MDR.
2. Write back OR-2 incremented plus 1.
3. Turn ON digit/record mark trigger when record mark appears in MDR.

Hold Trigger. (01.64.12.1).

1. Block MAR reset to retain output character in MDR.
2. Turn ON read-write call trigger.
3. Stop the computer clock to prevent unnecessary cycling of memory.

Trigger 31. (01.64.11.1).

1. Read out of memory per OR-2.
2. Write back OR-2 bypassed.

- Turn on the response check latch.
(Trigger 31 is used as a dummy E-timer trigger in Write Numerically - Code 38 operations. See Read Alphanumerically - Code 37 for trigger 31 function.)

Dump Numerically (Code 35 — DN)

Sequence block diagrams for this operation are shown in Figures 14-12 and 14-13. Function charts for this operation are shown in the *Instructional System Diagrams*, pages 39, 40, 41, and 42. Sequence charts for the typewriter, used as the output device, are shown in Figures 14-8, 14-9, 14-10, and 14-11.

Objectives

- Transmit numerical information, including flag bits and record marks, from memory starting at the location designated by the P address (OR-2) and continuing through location 19999.
- Record the information on the output device specified by the Q_8 and Q_9 digits of the instruction.

Functions

The Dump Numerically - Code 35, operation differs

from Write Numerically - Code 38, operation only as follows:

- Sensing of a record mark from memory does not terminate the dump operation. (Record marks in memory are recorded by the output device as characters of output data.)
- The dump operation is terminated when the digit in memory location 19999 has been transmitted to the output device. (If the output device is the paper tape punch, a hole is punched in the end-of-line channel of the tape in the tape column immediately following that occupied by the character transmitted from memory location 19999.)

Termination of the dump operation directs the computer to enter the I-cycle for the next instruction in sequence.

The content of every memory location is recorded by the output device if the P address of the dump instruction is 00000.

Auxiliary Trigger Status

TYPEWRITER AND PAPER TAPE PUNCH TRIGGERS

Decrement Trigger. (01.60.05.1).

- Turned OFF (increment) when I-cycle is entered.
- Remains OFF throughout E-cycle.

(See Write Numerically - Code 38 Sequence Block Diagram, Figure 14-7, for Sequence Other Than Termination Of Operation.)

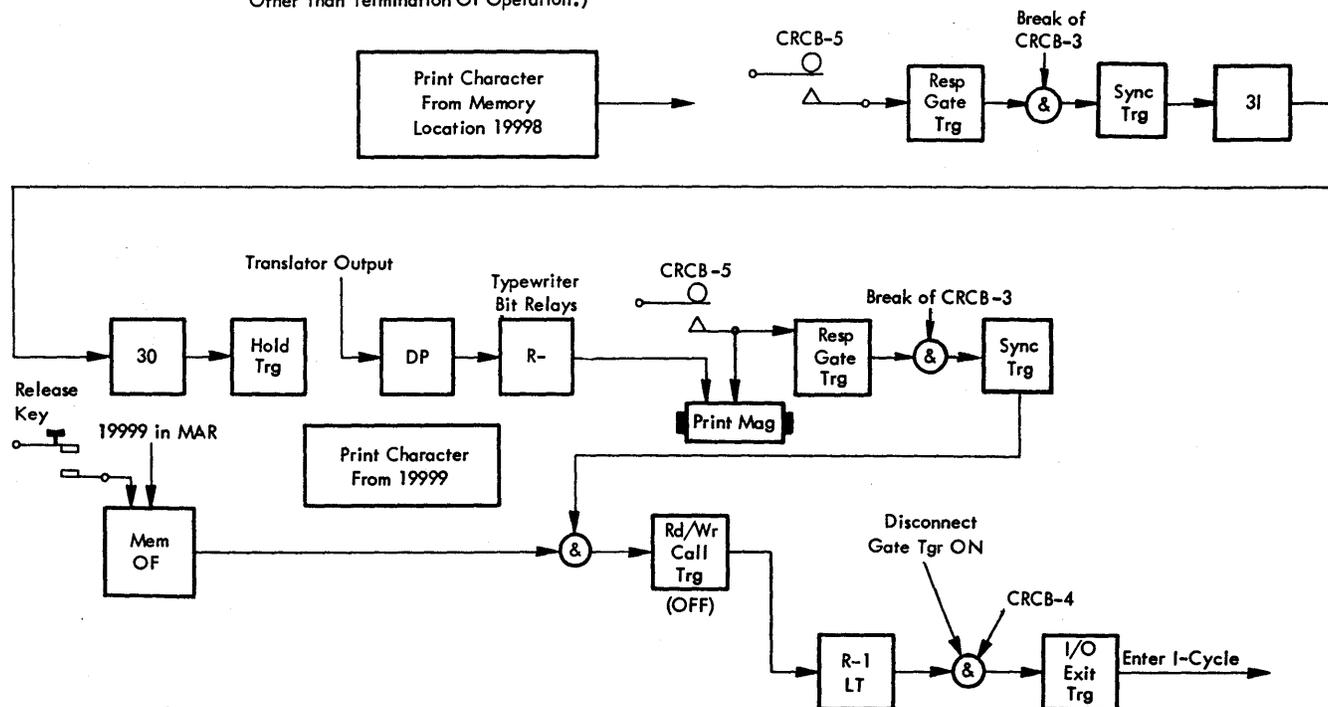


Figure 14-12. Dump Numerically—Code 35 (35 P P P P P Q 0 1 Q Q)

(See Write Numerically - Code 38 Sequence Block Diagram, Figure 14-7, For Sequence Other Than Termination Of Operation.)

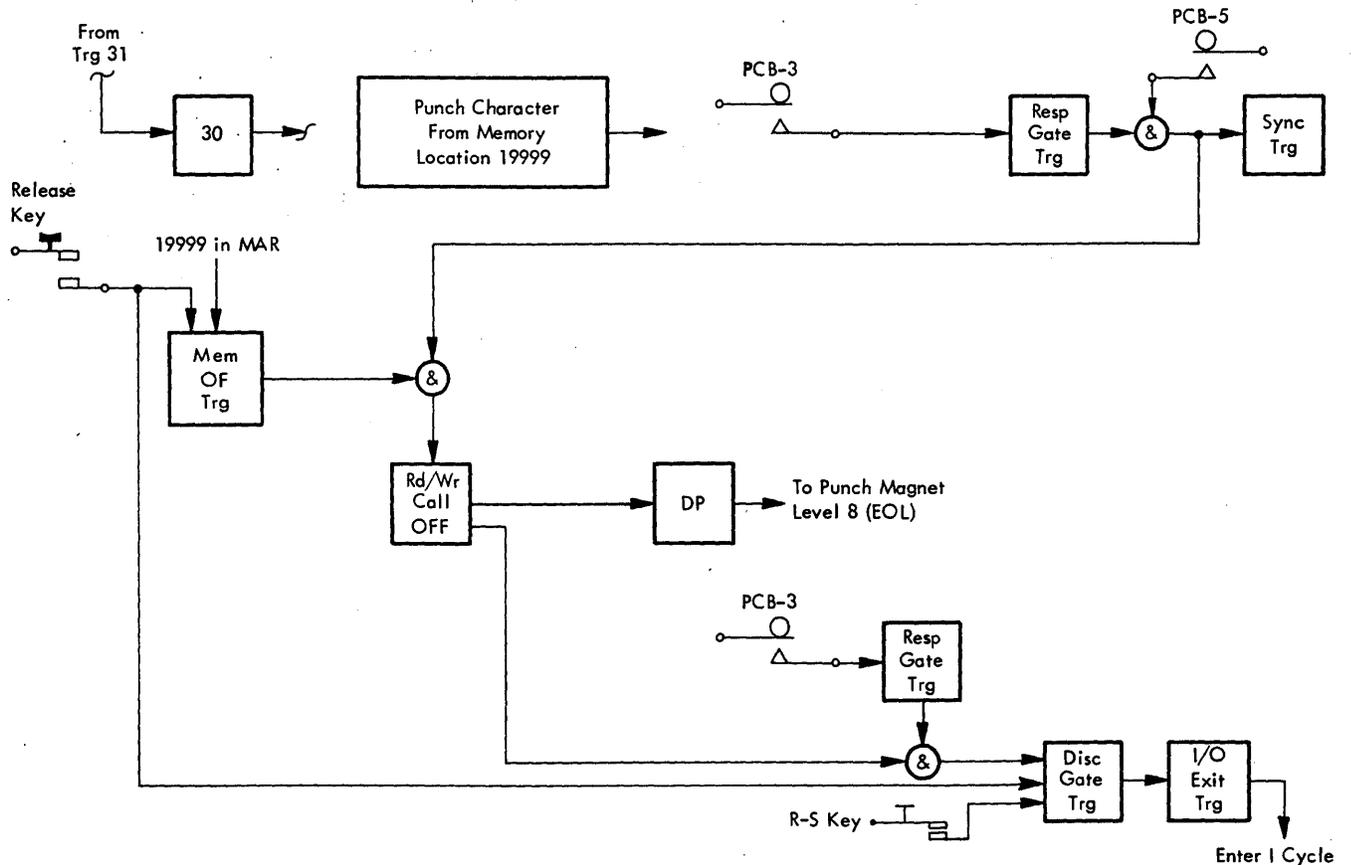


Figure 14-13. Dump Numerically—Code 35 (35 P P P P P Q 0 2 Q Q)

Read-Write Call Trigger. (01.64.14.1).

1. OFF when E-cycle is entered.
2. Turned ON by the hold trigger and remains ON until the dump operation is terminated.
3. Turned OFF by the MEM-OF trigger to terminate the dump operation.
4. Turned OFF by the I/O exit trigger if RELEASE is depressed during the dump operation.

Response Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned ON by a signal from the output device that a character has been received from the computer.
3. Turned OFF by the sync trigger.
4. Turned OFF by the I/O exit trigger when the dump operation is terminated.
5. Turned OFF by the no feed trigger if the paper tape punch is the output device and a parity or response error occurs or the paper tape breaks.

Sync Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON during hold trigger time when CRCB-3 breaks for a typewriter operation or PCB-1 makes for paper tape punch operation to signal the computer to proceed with presenting the next character to the output device.
3. Turned OFF during trigger 30 time.
4. Turned off by the I/O exit trigger when the write operation is terminated.

Memory Overflow. (01.64.14.1).

1. OFF when E-cycle is entered.
2. Turned ON during memory cycle in which 19999 is placed in MAR.
3. Turned ON by depression of RELEASE.
4. Turned OFF by I/O exit trigger.

Disconnect Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.

Write Check Trigger. (01.81.45.1).

1. Turned ON (in parallel with the sync trigger) by the response gate trigger, if:
 - a. The output translator presents a character with incorrect parity to the output device.
 - b. The response check latch is on.
2. Turned ON when PCB-3 makes if an even number of holes is punched by the paper tape punch.
3. Turned ON when CRCB-5 makes, if an even number of typewriter coding relays are picked. (Relay 38 will not pick if an even number of coding relays are picked.)
4. Turned ON by trigger 1 of the I-cycle following a write operation, using the typewriter as the output device, if the response check latch is on. This circuit is functional only if the response failure occurs on the last character prior to sensing the record mark from memory.
5. Turned OFF by operating the tape feed switch if the paper tape punch is the selected output device.
6. Turned OFF when interrogated by a branch indicator or branch no indicator instruction or when manually reset.

PAPER TAPE PUNCH TRIGGERS

When the paper tape punch is the output device, the following additional triggers are used:

No Feed Trigger. (01.84.10.1).

1. OFF when E-cycle is entered.
2. Turned ON if tape tension contact closes (paper tape break).
3. Turned ON by the write check trigger (parity or response error).
4. Turned OFF by depression of tape feed key.

E-Timer Trigger Objectives

Trigger 30. (01.64.10.1).

1. Read out of memory per OR-2 and store the digit in MDR.
2. Write back OR-2 incremented plus 1.

Hold Trigger. (01.64.12.1).

1. Block MAR reset to retain output character in MDR.
2. Turn ON read-write call trigger.
3. Stop the computer clock to prevent unnecessary cycling of memory.

Trigger 31. (01.64.11.1).

1. Read out of memory per OR-2.
2. Write back OR-2 bypassed.
3. Turn ON the response check latch.
(Trigger 31 is used as a dummy E-timer trigger in Dump Numerically - Code 35 operations. See Read Alphanumerically - Code 37 for Trigger 31 function.)

Write Alphanumerically (Code 39 — WA)

Sequence block diagrams for this operation are shown in Figures 14-6 and 14-7. Sequence block diagrams for Write Numerically - Code 38 and Write Alphanumerically - Code 39 differ only in that, for Code 39 operations, the output translator receives a digit (zone) from MBR-even as well as from MDR. Function charts for this operation are shown in the *Instructional System Diagrams*, pages 39, 40, 41, and 42.

Objectives

1. With the computer in alphanumeric mode, transmit characters stored as two adjacent digits from the memory locations designated by the P address minus one and the P address (OR-2) and successively higher pairs of memory locations.
2. Decode each "two digit" memory character into the proper numerical, alphabetic, or special character.
3. Record the information on the output device specified by the Q_8 and Q_9 digits of the instruction.

Functions

During the I-cycle, the Q_8 and Q_9 digits of the instruction are placed in the digit/branch register and decoded to specify the output device. The typewriter is specified by 01 and the paper tape punch by 02.

Output characters are transmitted serially from memory as two adjacent digits beginning at the memory locations designated by the P address minus one and the P address (OR-2) and continuing with successively higher pairs of memory locations. Each "two digit" memory character is decoded into the proper numerical, alphabetic, or special character by the output translator and presented to the selected output device.

The P address of the instruction must designate the odd numbered memory location at which the numerical digit of the first character to be transmitted is stored. The zone digit of the first character is located at the next lower memory address (P minus one), which is an even address. Increment plus two is used to provide memory addresses for successive characters.

If an even numbered memory location is erroneously designated by the P address, data presented to the output device is incorrect and parity errors may occur.

Output data may consist of a random mixture of numerical, alphabetic, and special characters which exist in memory as "two digit" characters. If an attempt is made to write a record containing single digit numerical characters by means of this instruction, in-

valid combinations of disassociated numerical digits are the result and parity errors may occur.

Flag bits existing in memory locations from which an output record is written are not transmitted to the output device. The write operation leaves output data unchanged in memory.

Transmission of a character of output data to the typewriter requires energization of the typewriter bit relay (or relays) corresponding to the typewriter character code for the specific character. Bit relays are assigned as follows:

<i>Bit</i>	<i>Relay</i>
C	R21
X	R22 and R25
0	R28 and R31
1	R41
2	R42 and R45
4	R46 and R49
8	R50

Relay 21, the C bit relay, is used only for parity checking purposes. If no bit relay other than R21 is energized on an output character cycle, the typewriter performs a spacing operation.

Transmission of a character of output data to the paper tape punch requires energization of the punch magnet (or magnets) corresponding to the paper tape code for the specific character. A space is recorded as a hole in the "C" channel of the tape.

The write check trigger is turned ON when any one or more of the following conditions exists:

1. A character with incorrect parity is presented to the output device by the output translator.
2. Typewriter bit relays are energized for an even number of bits.
3. An even number of holes are punched in a tape column by the paper tape punch.
4. The response check latch is on because the output device failed to return a response signal to the computer.

If the write check trigger is turned ON with the typewriter as the output device, the computer remains in the automatic mode and continues to supply data to the typewriter until the write operation is terminated. If the write check trigger is turned ON with the paper tape punch as the output device, the punch is stopped immediately with the incorrectly punched tape column still under the punches.

With the typewriter as the output device, the write operation is terminated by one of the following:

1. Sensing a record mark at an odd memory address. Termination of the write operation directs the computer to enter the I-cycle for the next instruction in sequence. The record mark is not printed.

2. Depressing RELEASE on the console. The write operation is terminated immediately and the computer stops. Depression of START on the console is then required to restart the computer.

With the paper tape punch as the output device, the write operation is terminated by one of the following:

1. Sensing a record mark at an odd memory address. A hole is punched in the end-of-line channel (channel 8) of the tape in the tape column immediately following that occupied by the last output data character transmitted from memory. Termination of the write operation directs the computer to enter the I-cycle for the next instruction in sequence.
2. Depressing RELEASE on the console. The write operation is terminated immediately and the computer stops. Depression of START on the console is required to restart the computer.

Auxiliary Trigger Status

See Write Numerically - Code 38.

E-Timer Trigger Objectives

Trigger 30. (01.64.10.1).

1. Read out of memory per OR-2.
2. Write back OR-2 incremented plus 2.
3. Turn ON digit/record mark trigger when record mark appears in MDR.

Hold Trigger. (01.64.12.1).

1. Block MAR reset to retain the zone digit of the output character in MBR-even and the numerical digit of the output character in MDR.
2. Turn ON read-write call trigger.
3. Stop the computer clock to prevent unnecessary cycling of memory.

Trigger 31. (01.64.11.1).

1. Read out of memory per OR-2.
2. Write back OR-2 bypassed.
3. Turn on response check latch. (Trigger 31 is used as a dummy E-timer trigger in Write Alphamerically - Code 39 operations. See Read Alphamerically - Code 37 for Trigger 31 function.)

Control (Code 34 — K)

The sequence block diagram for this operation is shown in Figure 14-15. The function chart for this operation is shown in the *Instructional System Diagrams*, page 43. Sequence charts, Figures 14-16, 14-17, and 14-18 show the operation of: space, carriage return, and tabulate.

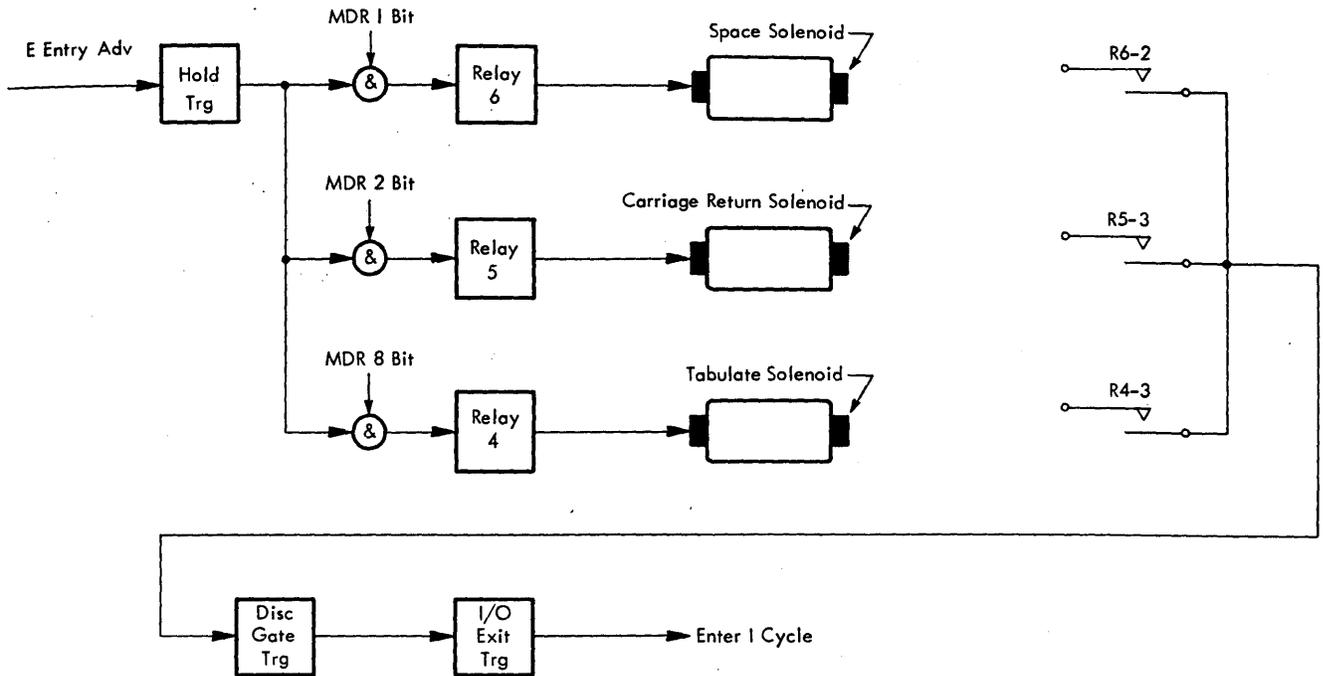


Figure 14-15. Control-Code 34

Objective

Execute the control function designated by the Q_{11} digit of the instruction on the input-output device specified by the Q_8 and Q_9 digits of the instruction.

Functions

An 01 in the Q_8 and Q_9 digits of the instruction is placed in the digit/branch register during the I-cycle and decoded to select the typewriter.

The Q_{11} digit of the instruction is available in MDR at the end of the I-cycle for use in specifying the con-

trol function. One-digit codes are assigned as follows:

- 1 — Space
- 2 — Carriage return
- 8 — Tabulate

When the hold trigger is turned ON upon entry into E-cycle, an MDR 1, 2, or 8 bit ANDs with select 1 (typewriter) to pick Relay 6 (space), Relay 5 (carriage return), or Relay 4 (tabulate). Each of these relays controls the performance of its particular control function.

The typewriter remains selected only until the execution of the control function is completed.

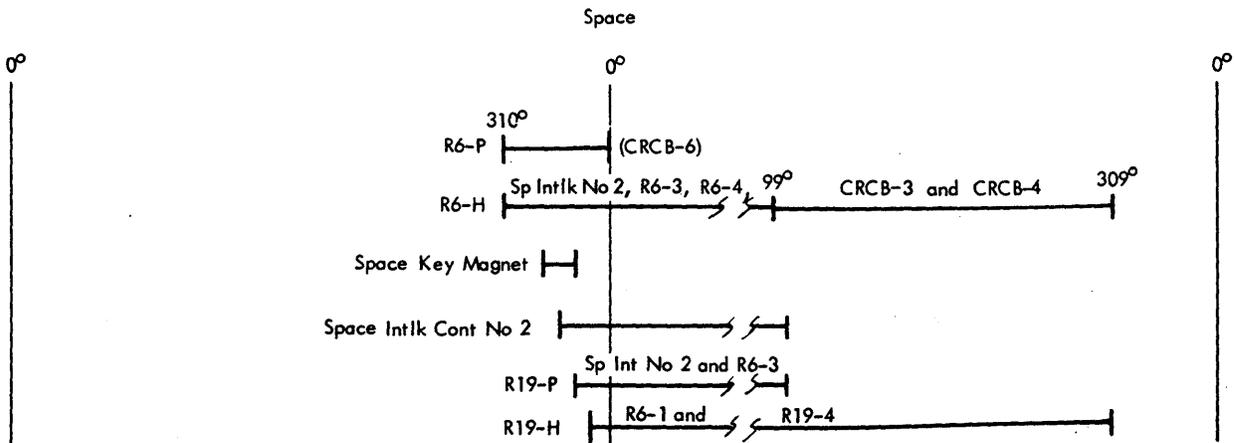


Figure 14-16. Space

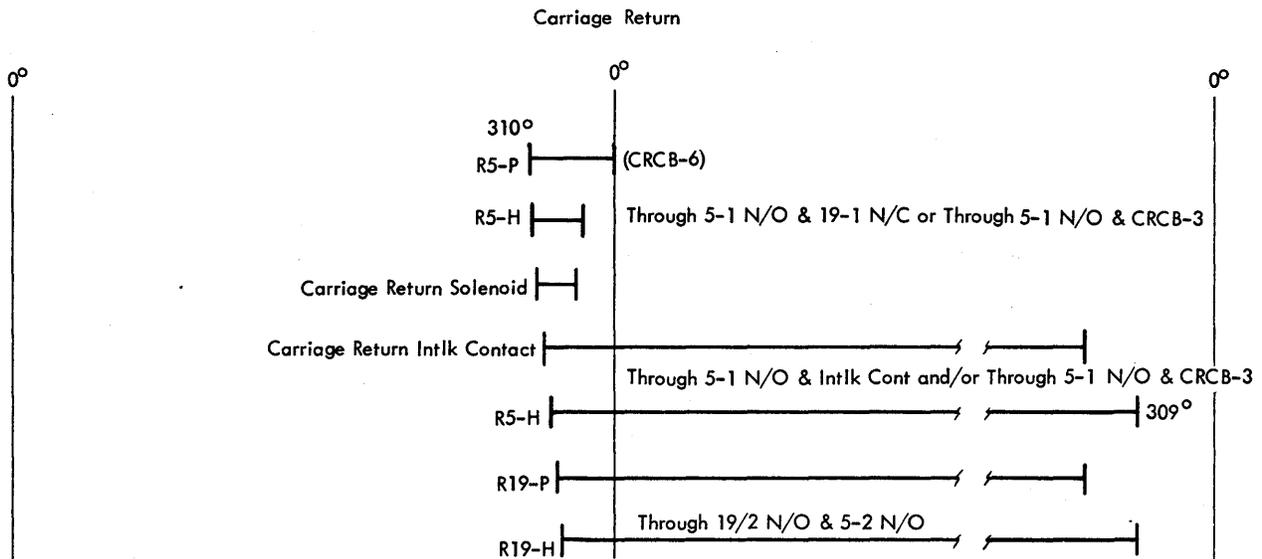


Figure 14-17. Carriage Return

See *Customer Engineering Manual of Instruction, B1 Electric Typewriter As Modified for DP Equipment, (Form 223-6653)* for a description of the following typewriter components:

- Space solenoid
- Carriage return solenoid
- Tabulate solenoid
- Carriage return interlock contact
- Tab interlock contact

Auxiliary Trigger Status

Disconnect Gate Trigger. (01.80.25.1).

1. OFF when E-cycle is entered.
2. Turned ON by signal from typewriter that control function has been initiated.
3. Turned OFF by I/O exit trigger.

I/O Exit Trigger. (01.64.13.1).

1. OFF when E-cycle is entered.
2. Turned ON by signal from typewriter that the control function has been completed.
3. Turned OFF during trigger 1 time of the I-cycle for the next instruction.

E-Timer Trigger Objectives

Hold Trigger. (01.64.12.1).

1. Block MAR reset to maintain the control function code digit in MDR and to prevent VRC error.
2. Pick Relay 6, 5, or 4 for performance of control function.
3. Stop the computer clock to prevent unnecessary cycling of memory.

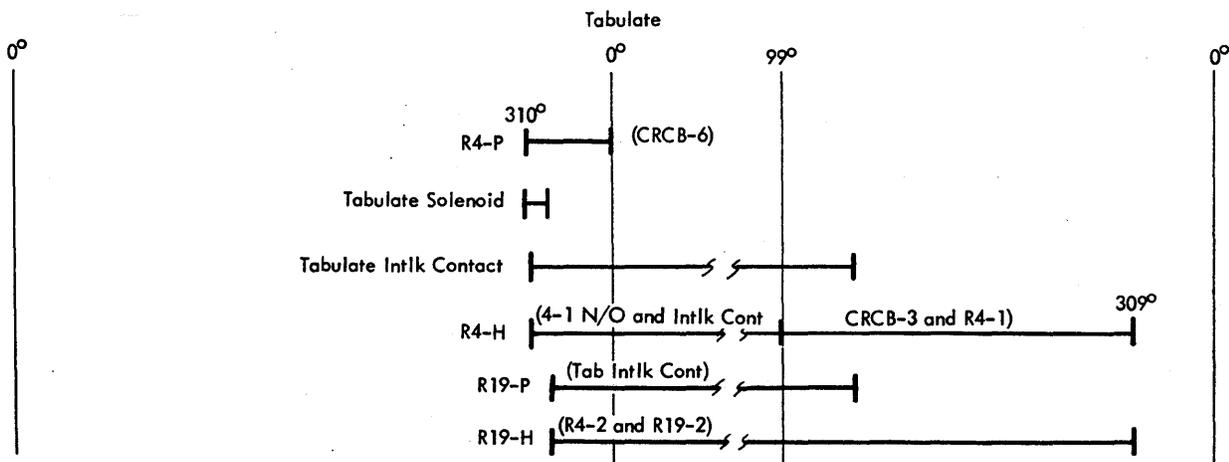


Figure 14-18. Tabulate

Appendix A. Abbreviations and Terms

A or B Advance	A or B Adv	Microampere	μ a
Above 50 (greater than)	>50	Microsecond	μ sec
Below 50 (Less than)	<50	Multiplicand	Mcnd
Branch Indicator	BI	Multiplier/Quotient (Register)	M/Q (Reg)
Branch No Indicator	BNI	Negative "or"	NOR
Carriage Return (typewriter)	CR	Non Process Run Out	NPRO
Carry In (trigger)	CI (trg)	Operand Register	OR (-1, -2, -3)
Carry Out (trigger)	CO (trg)	Operation Register	Op Reg
Check (bit)	C (bit)	Paper Tape Reader	PT Rdr
Circuit Breaker	CB	Product Register	PR (-1, -2, -3)
Continuously Running Circuit Breaker	CRCB	Read Alphanumerically	RA
Digit/Record Mark (trigger)	D/RM (trg)	Read Check	Rd Chk
Decrement (trigger)	Decr (trg)	Read Driver	Rd Dr
Digit/Branch Register	D/B Reg	Reader (data gate)	Rdr (data gate)
Disconnect (gate trigger)	Disc. (gate trg)	Read Numerically	RN
Divide	Div	Read/Write (call) (trigger) (gate)	Rd/Wr (call) (trg) (gate)
Driver, Sample Pulse	DSP	Recomplement	Recomp
End of Line	EL or EOL	Record Mark	RM
Field Definition	Fld Def	Resistor-Capacitor	RC
Field Mark (trigger)	FM (trg)	Response (gate) (trigger)	Resp (gate) (trg)
Flag	F	Selector Common Contacts	Sel Com Cont
Increment/Decrement (switch)	Incr/Decr (Sw)	Single Cycle Execute	SCE
Input/Output Exit (trigger)	I/O Exit (trg)	Single Instruction Execute	SIE
Instruction Cycle	I-Cycle	Synchronizing (trigger)	Sync (trg)
Instruction Register	IR (-1, -2)	Tabulate (typewriter)	Tab
Integrator	Int	Trigger	Trg
Inverter	Inv	True/Complement (switch) (trigger)	T/C (sw) (trg)
Memory Address Register	MAR	Vertical Redundancy Check (Parity Check)	VRC
Memory Address Register Storage	MARS	Write Alphanumerically	WA
Memory Buffer Register	MBR	Write Check	Wr Chk
Memory Data Register	MDR	Write Drivers	Wr Dr
Memory Overflow (trigger)	Mem OF (trg)	Write Numerically	WN

Appendix B. Character Code Chart

Character	Alphameric Mode					Numerical Mode		
	Memory Code				Typewriter & Paper Tape Code	Memory Code	Typewriter & Paper Tape Code	
	Zone		Numerical					
	Digit	Bits	Digit	Bits				
	C 4 2 1		C 8 4 2 1	X 0 C 8 4 2 1	C F 8 4 2 1	X 0 C 8 4 2 1		
bl space	0	C	0	C	C			
.	0	C	3	C 2 1	X 0 8 2 1			
)	0	C	4	4	X 0 C 8 4			
+	1	1	0	C	X 0 C			
\$	1	1	3	C 2 1	X C 8 2 1			
*	1	1	4	4	X 8 4			
-	2	2	0	C	X			
/	2	2	1	1	0 C 1			
'	2	2	3	C 2 1	0 C 8 2 1			
(2	2	4	4	0 8 4			
=	3	C 2 1	3	C 2 1	8 2 1			
@	3	C 2 1	4	4	C 8 4			
A	4	4	1	1	X 0 1			
B	4	4	2	2	X 0 2			
C	4	4	3	C 2 1	X 0 C 2 1			
D	4	4	4	4	X 0 4			
E	4	4	5	C 4 1	X 0 C 4 1			
F	4	4	6	C 4 2	X 0 C 4 2			
G	4	4	7	4 2 1	X 0 4 2 1			
H	4	4	8	8	X 0 8			
I	4	4	9	C 8 1	X 0 C 8 1			
J	5	C 4 1	1	1	X C 1			
K	5	C 4 1	2	2	X C 2			
L	5	C 4 1	3	C 2 1	X 2 1			
M	5	C 4 1	4	4	X C 4			
N	5	C 4 1	5	C 4 1	X 4 1			
O	5	C 4 1	6	C 4 2	X 4 2			
P	5	C 4 1	7	4 2 1	X C 4 2 1			
Q	5	C 4 1	8	8	X C 8			
R	5	C 4 1	9	C 8 1	X 8 1			
S	6	C 4 2	2	2	0 C 2			
T	6	C 4 2	3	C 2 1	0 2 1			
U	6	C 4 2	4	4	0 C 4			
V	6	C 4 2	5	C 4 1	0 4 1			
W	6	C 4 2	6	C 4 2	0 4 2			
X	6	C 4 2	7	4 2 1	0 C 4 2 1			
Y	6	C 4 2	8	8	0 C 8			
Z	6	C 4 2	9	C 8 1	0 8 1			
0	7	4 2 1	0	C	0	C	0	
1	7	4 2 1	1	1	1	1	1	
2	7	4 2 1	2	2	2	2	2	
3	7	4 2 1	3	C 2 1	C 2 1	C 2 1	C 2 1	
4	7	4 2 1	4	4	4	4	4	
5	7	4 2 1	5	C 4 1	C 4 1	C 4 1	C 4 1	
6	7	4 2 1	6	C 4 2	C 4 2	C 4 2	C 4 2	
7	7	4 2 1	7	4 2 1	4 2 1	4 2 1	4 2 1	
8	7	4 2 1	8	8	8	8	8	
9	7	4 2 1	9	C 8 1	C 8 1	C 8 1	C 8 1	
Record Mark	‡	0	C	‡	C 8 2	0 8 2	C 8 2 0 8 2 or E.L.	
Flag						F	X	
Flag RM	‡	5	C 4 1	‡	C 8 2	X 8 2	F 8 2 X 8 2	

Appendix C. Numbered Triggers

Trigger Number	System Diagram Page Number
1	01.15.11.1
2	01.15.12.1
3	01.15.13.1
4	01.15.14.1
5	01.15.15.1
6	01.15.16.1
7	01.15.17.1
8	01.15.18.1
9	01.15.19.1
11	01.60.11.1
12	01.60.12.1
13	01.60.13.1
14	01.60.14.1
15	01.60.45.1

Trigger Number	System Diagram Page Number
16	01.60.45.1
18	01.60.48.1
19	01.60.49.1
21	01.60.31.1
23	01.60.32.1
26	01.60.57.1
27	01.60.57.1
28	01.60.58.1
29	01.60.59.1
30	01.64.10.1
31	01.64.11.1
32	01.62.30.1
33	01.62.33.1
34	01.62.34.1

Trigger Number	System Diagram Page Number
35	01.62.35.1
36	01.62.36.1
37	01.62.37.1
38	01.62.38.1
39	01.62.39.1
40	01.62.40.1
41	01.62.41.1
42	01.65.05.1
43	01.65.05.1
44	01.65.03.1
45	01.65.03.1

Appendix D. Named Triggers

Trigger Name	System Diagram Page Number
A/B	01.10.12.1
Branch Test	01.25.35.1
Carry In	01.63.40.1
Carry Out	01.63.40.1
Clock 1 & Clock 2	01.10.06.1
Clock 3 & Clock 4	01.10.07.1
Clock 5 & Clock 6	01.10.08.1
Clock 7 & Clock 8	01.10.09.1
Clock 9 & Clock 10	01.10.10.1
Cycle Control	01.62.31.1
Decrement	01.60.05.1
Digit/Record Mark	01.63.50.1
Disconnect Gate	01.80.25.1
Display Address	01.06.30.1
Divide Add	01.65.04.1
Dividend & Remainder Sign	01.65.06.1
E-Cycle Entry	01.15.18.1
Equal/Zero	01.60.41.1
Field Mark Number 1	01.63.30.1
Field Mark Number 2	01.63.30.1
First Cycle	01.63.10.1
First Divide Cycle	01.65.02.1
Flag	01.81.10.1
Greater than 50 (> 50)	01.55.15.1
High/Plus	01.60.40.1
Hold	01.64.12.1
I-Cycle	01.15.10.1
I/O Exit	01.64.13.1
Insert 1	01.06.20.1
Insert 2	01.06.20.1

Trigger Name	System Diagram Page Number
Insert Control	01.06.20.1
Last Divide Cycle	01.65.02.1
MAR VRC	01.58.09.1
MBR - Odd Check	01.40.10.1
MBR - Even Check	01.40.20.1
Memory Overflow	01.64.14.1
Non-Process Run Out	01.83.10.1
Odd-Even	01.43.10.1
Overflow	01.60.40.1
Paper Tape Feed	01.84.10.1
Punch No Feed	01.84.10.1
Read Check	01.81.45.1
Read/Write Call	01.64.14.1
Recomplement	01.60.32.1
Recomplement Control	01.60.24.1
Response Gate	01.80.25.1
Run	01.06.10.1
Save 2	01.06.15.1
Save Control	01.06.15.1
Sel 1	01.25.15.1
Sel 5	01.25.15.1
Single Cycle	01.06.10.1
Start 1	01.06.05.1
Start Control	01.06.05.1
Stop 1	01.06.11.1
Sync	01.64.13.1
Sync Extender	01.80.30.1
Tape Feed Interlock	01.84.10.1
True/Complement	01.63.20.1
Write Check	01.81.45.1

Appendix E. Register Triggers

Register	Trigger Name	System Diagram Page Number
Multiplier/Quotient	C Bit, 8 Bit, & 4 Bit	01.62.50.1
	2 Bit & 1 Bit	01.62.52.1
Operation	C Bit, 8 Bit, & 4 Bit (Units)	01.20.07.1
	2 Bit & 1 Bit (Units)	01.20.09.1
	C Bit, & 4 Bit (Tens)	01.20.10.1
	2 Bit & 1 Bit (Tens)	01.20.12.1
MBR - Odd	C Bit & F Bit	01.40.07.1
	8 Bit & 4 Bit	01.40.08.1
	2 Bit & 1 Bit	01.40.09.1
MBR - Even	C Bit & F Bit	01.40.17.1
	8 Bit & 4 Bit	01.40.18.1
	2 Bit & 1 Bit	01.40.19.1
MDR	C Bit, F Bit, & 8 Bit	01.45.05.1
	4 Bit, 2 Bit, & 1 Bit	01.45.07.1
Digit/Branch	C Bit, 8 Bit, & 4 Bit (Units)	01.50.07.1
	2 Bit & 1 Bit (Units)	01.50.09.1
	C Bit, 8 Bit, & 4 Bit (Tens)	01.50.17.1
	2 Bit & 1 Bit (Tens)	01.50.19.1
MAR	C Bit & 8 Bit (Units)	01.57.05.1
	4 Bit & 2 Bit (Units)	01.57.07.1
	1 Bit (Units)	01.57.09.1
	C Bit (Tens)	01.57.09.1
	8 Bit & 4 Bit (Tens)	01.57.16.1
	2 Bit & 1 Bit (Tens)	01.57.18.1
	C Bit & 8 Bit (Hundreds)	01.57.20.1
	4 Bit & 2 Bit (Hundreds)	01.57.22.1
	1 Bit (Hundreds)	01.57.24.1
	C Bit (Thousands)	01.57.24.1
	8 Bit & 4 Bit (Thousands)	01.57.26.1
	2 Bit & 1 Bit (Thousands)	01.57.28.1
	C Bit, 2 Bit, & 1 Bit (10 Thousands)	01.57.30.1

Appendix G. Auxiliary Trigger Chart

Operation Code	E-Cycle Entry	Decrement	First Cycle	Field Mk. No. 1	Field Mk. No. 2	I/C	High/Plus	Equal/Zero	Carry Out	Carry In	Recomp Cnt	Recomp	Cycle Cnt	Dividend & Remainder Sign	Divide Add	First Divide Cycle	Last Divide Cycle	Save Cnt.	Digit/M	Branch Test	Response Gate	Sync	Disconnect Gate	I/O Exit	Stop 1	Rd/Mr Call	Write Chk.	Read Chk.	Mem O'Flow	Overflow	Flag
11 & 21	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
12 & 22	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
13 & 23	Δ	I&D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
14 & 24	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
15 & 25	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
16 & 26	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
17 & 27	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
18 & 28	Δ	I&D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
19 & 29	Δ	D&I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
31	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
32	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
33	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
34	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
35	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
36	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
37	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
38	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
39	Δ	I	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
41	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
42	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
43	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
44	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
45	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
46	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
47	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
48	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	
49	Δ	D	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	

- Brake:** A mechanism (composed of an armature, rotor, coil, and housing) that hastens stopping of the pinwheel shaft and (tape) when reading ceases. A brake potentiometer controls the brake current.
- Buffer Arm:** A lever interposed between the tape reels and read head to control the motion of the reels; actuated by the tape.
- Buffer Arm Cam:** A cam operated by a buffer arm; actuates the buffer-arm contacts and limit switches.
- Buffer Arm Limit Switch:** The lower switch controls the tape reels on tape break; the upper switch prevents reel operation when the buffer arms are detented prior to tape loading.
- Cam Detent:** A mechanical detent for holding the buffer arms in the raised position prior to tape loading.
- Center Plate:** An assembly, mounted centrally on the main plate, that holds the read head, pinwheel, pinwheel shaft, timing disk, brake and clutch.
- Clutch:** A mechanism, similar to the brake in construction, that drives the pinwheel shaft and moves tape past the read head. The clutch potentiometer controls the clutch current.
- Clutch Shaft:** The shaft that drives the clutch rotor; power to the clutch shaft is supplied by the main drive motor through a belt and pulley.
- Main Plate:** The major casting, on which are mounted the center plate, tape reels, and buffer arms.
- Pinwheel:** A 24-tooth sprocket that drives the tape past the read head.
- Pinwheel Shaft:** The shaft, driven by the clutch, that turns the pinwheel and timing disk.
- Read Head:** An assembly, mounted on the front of the center plate, that houses the read photocells and encloses the pinwheel.
- Read Lamp:** The source of light for the read photocells.
- Read Lamp Potentiometer:** A potentiometer for adjusting the voltage to the read lamp and controlling the output of the read photocells.
- Read Photocell:** A light-sensitive duo-diode that detects the presence of a hole in the tape. There are eight read photocells in the read head.
- Read Photocell Adjusting Potentiometer:** A potentiometer for adjusting the output of a photocell about a base level.
- Reel Power Switch:** A switch, located at the bottom of the main plate, that permits the reel to be driven after the tape is loaded.
- Reel Strip Mode Switch:** A switch that puts the reader in reel mode or strip mode.
- SMS Card Gate:** A chassis for holding the electronic circuits, such as photocell amplifiers and clutch selector and control.
- Supply Buffer Arm Contacts:** The making of the upper contact causes the supply reel to deliver tape to the buffer arm; the lower contact, normally operated during the tape load operation, takes up slack in the tape.
- Supply Reel:** The right-hand reel on the main plate which delivers tape to the read head.
- Supply Reel Motor:** The motor that drives the supply reel and is actuated by the making of the supply buffer arm contacts.
- Take-up Buffer Arm Contacts:** The making of the lower contact causes the take-up reel to store the tape delivered from the read head; the upper contact puts more slack in the tape at the take-up buffer arm.
- Take-up Reel:** The left-hand reel on the main plate which is used to store the tape after it is read.
- Take-up Reel Motor:** Actuated by the take-up buffer arm contacts. This motor drives the take-up reel and is actuated by the take-up buffer arm contacts.
- Tape Guide:** The mechanical component that holds the tape in contact with the read head and in mesh with the pinwheel.
- Tape Guide Assembly:** A mechanism composed of the tape guides and associated stops, arms, and linkages; all parts except the guides are mounted on the rear of the center plate.
- Tape Guide Microswitch:** The switch that indicates to the using system that the tape guides are closed (against the read head).
- Tape Guide Stop Arm:** The lever whose position determines the clearance between a tape guide and the read head.
- Tape Runout Bar:** The lower of two studs adjacent to the read head; its function is to control the sequence of operations during runout of the tape. The runout bar is mounted on the lower end of the tape runout arm, which actuates the tape runout microswitch.
- Tape Tension Arm:** The lever that supports the tape tension bar and actuates the tape tension microswitch.
- Tape Tension Bar:** The upper of two studs adjacent to the read head; its function is to signal the using system that the tape is not in proper contact with the read head at the read photocells.
- Tape Tension Microswitch:** Signals the using system that the tape is not against the read head on a tape break.
- Timing Disk:** (Sync Disk) The 24-hole disk, mounted on the pinwheel shaft, that sets the time relation between the read photocell output and the timing disk output.
- Timing Disk Lamp:** The source of light for the timing-disk photocell.
- Timing Disk Potentiometer:** A potentiometer for controlling the voltage to the timing disk lamp, and the output of the timing-disk photocell.
- Timing Disk Photocell:** A light-sensitive duo-diode that detects the holes in the timing disk as the disk rotates, and thus recognizes the position of the pinwheel and tape at the read head.
- Timing Disk Photocell Adjusting Potentiometer:** A potentiometer for positioning the output of the photocell about a base level.

Index

	<i>Page</i>		<i>Page</i>
Abbreviations and Terms	173	Clock and Timing Chart	40
Add, Auxiliary Trigger Status	88	Clutch	134, 179
Add (Code 21)	85	Clutch Shaft	179
Add, E-Timer Trigger Objectives	90	Code Check Unit	147
Add Immediate, Auxiliary Trigger Status	92	Compare, Auxiliary Trigger Status	100
Add Immediate (Code 11-AM)	92	Compare (Code 24-c)	97
Add Immediate, E-Timer Trigger Objectives	92	Compare, E-Timer Trigger Objectives	101
Automatic Light	69	Compare Immediate, Auxiliary Triggers	102
Automatic Program Repetition Program Loop	19	Compare Immediate (Code 14-cm)	102
Auxiliary Trigger Functions	76	Compare Immediate E-Timer Trigger Objectives	102
Brake	135, 179	Component Circuits	22
Brake and Clutch Selector and Control Circuit	142	Console	60
Branch and Transmit, Auxiliary Trigger Status	122	Control, Auxiliary Trigger Status	172
Branch and Transmit (Code 27-BT)	122	Control (Code 34-κ)	170
Branch and Transmit, E-Timer Trigger Objectives	122	Control, E-Timer Trigger Objectives	172
Branch and Transmit Immediate, Auxiliary Trigger Status	124	Cores, Magnetic	21
Branch and Transmit Immediate (Code 17-BTM)	124	Data Format	7
Branch and Transmit Immediate, E-Timer Trigger Objectives	124	Description of System	5
Branch, Auxiliary Trigger Status	122	Development of a Program	17
Branch Back, Auxiliary Trigger Status	124	Digit/Branch Register and Decoder	55
Branch Back (Code 42-BB)	124	Digit/Branch Register Display	60
Branch Back, E-Timer Trigger Objectives	124	Divide, Auxiliary Trigger Status	117
Branch (Code 49-B)	122	Divide (Code 29-b)	111
Branch, E-Timer Trigger Objectives	122	Divide, E-Timer Trigger Objectives	119
Branch Indicator, Auxiliary Trigger Status	129	Divide Immediate, Auxiliary Triggers Status	120
Branch Indicator (Code 46-BI)	127	Divide Immediate (Code 19-dm)	120
Branch Indicator, E-Timer Trigger Objectives	129	Divide Immediate, E-Timer Trigger Objectives	120
Branch No Flag, Auxiliary Trigger Status	126	Doubler	57
Branch No Flag (Code 44-BNF)	125	Dump Numerically, Auxiliary Trigger Status	166
Branch No Flag, E-Timer Trigger Objectives	126	Dump Numerically (Code 35-dn)	166
Branch No Indicator, Auxiliary Trigger Status	130	Dump Numerically, E-Timer Trigger Objectives	169
Branch No Indicator (Code 47-BNI)	130	Execution Cycle	76
Branch No Indicator, E-Timer Trigger Objectives	130	Execution of a Program	17
Branch No Record Mark, Auxiliary Trigger Status	127	E-Timer Triggers, Execution Cycle	76
Branch No Record Mark (Code 45-BNR)	126	Format, Data	7
Branch No Record Mark, E-Timer Trigger Objectives	127	Functional Units	40
Branch On Digit, Auxiliary Trigger Status	125	Glossary	179
Branch On Digit (Code 43-BD)	124	Halt Auxiliary Trigger, Status	132
Branch On Digit, E-Timer Trigger Objectives	125	Halt (Code 48-H)	132
Branch Operations	122	Halt E-Timer Trigger, Objectives	132
Buffer Arm	179	I-Timer Triggers	74
Buffer Arm Cam	179	Immediate Off Switch Activated	38
Buffer Arm Contacts, Supply	179	Immediate Off Switch—Console	63
Buffer Arm Limit Switch	179	Increment/Decrement Switch	14, 53
C Bit Correctors	57	Indicator Conditions, Add	88
Cam, Detent	179	Indicator Conditions, Compare	100
Carry Out Trigger, Function of	79	Indicator Conditions, Divide	111
CE Panel	69	Indicator Conditions, Multiply	104
CE Switches	69	Indicator Conditions, Subtract	95
Center Plate	179	Input-Output Devices	133
Center Roll Feed (CRF) Mode	138	Input-Output Operations	151
Character Code	6	Input-Output Operations, Paper Tape Punch	153
Clear Flag, Auxiliary Trigger Status	132	Input-Output Operations, Paper Tape Reader	153
Clear Flag (Code 33-CF)	131		
Clear Flag, E-Timer Trigger Objectives	132		

	<i>Page</i>		<i>Page</i>
Input-Output Operations, Typewriter	151	Motor, Take-Up Reel	179
Insertion of Program Into Memory (Paper Tape Reader)	19	Multiplier/Quotient Register	56
Insertion of Program Into Memory (Typewriter Keyboard) ..	17	Multiplier/Quotient Register Display	60
Instruction Cycle	71	Multiply, Auxiliary Trigger Status	104
Instruction Cycle, Auxiliary Triggers	72	Multiply (Code 23-M)	102
Instruction Format	6	Multiply, E-Timer Trigger Objectives	106
Internal Transmission Operations	82	Multiply Immediate, Auxiliary Trigger Status	109
Introduction	11	Multiply Immediate (Code 13-MM)	107
		Multiply Immediate, E-Timer Trigger Objectives	109
Keys, Control	63	No Operation, Auxiliary Trigger Status	132
Key, Display MAR	63	No Operation (Code 41-NOP)	132
Key, Insert	63	No Operation, E-Timer Trigger Objectives	132
Key Instant Stop/Single Execute (SCE)	65	NOR Latch	22
Key, Release	65		
Key, Reset	63	Operation Register and Decoder	55
Key, Save	63	Operation Register Display	60
Key, Start	65	Operations, Miscellaneous	131
Key, Stop/Single Instruction Execute (SIE)	65	Overflow Check	60
Key, Tape Switch	149	Overtemp Trip, Thermal	37
Lamp, Read	179	P & Q Parts of Instruction, Significance of	80
Lamp, Timing Disk	140	Paper Tape Code	152
Lamps, Incandescent	140	Paper, Tape Punch, Code Check Unit	147
Latch, Response Check	157, 165, 168	Paper Tape Punch, Drive Mechanism	143
Latches, Quotient Counter	119	Paper Tape Punch, IBM 1624	143
Light, Automatic	69	Paper Tape Punch, Latch Lock	146
Light, Check Stop	69	Paper Tape Punch, Tape Supply Disk and Rewind Reel Assembly	149
Light, Insert	69	Paper Tape Punch, Sequence of Operation	148
Light, Manual	69	Paper Tape Punch, Tape Feed	147
Light, Power On	65	Paper Tape Reader, Circuit Electronics	140
Light, Power Ready	69	Paper Tape Reader, Component Locations	133
Light, Punch No Feed	69	Paper Tape Reader, Glossary	179
Light, Reader No Feed	69	Paper Tape Reader, IBM 1621	133
Light, Save	69	Paper Tape Reader, Main Plate, Front	133
Light, Thermal	69	Paper Tape Reader, Main Plate, Rear	134
Lights, Machine Check Trigger	60	Paper Tape Reader, Mechanical Components	134
Lights, Machine Timer Trigger, Auxiliary Trigger, and Control Gate	62	Paper Tape Reader, Reel Operation	136
Lights, Signal	65	Paper Tape Reader, Reading Mechanism	133
Load Dividend, Auxiliary Trigger Status	110	Paper Tape Reader, Strip Mode Operation	138
Load Dividend (Code 28-LD)	109	Photocell, Read	179
Load Dividend, E-Timer Trigger Objectives	110	Photocell, Timing Disk	179
Load Dividend Immediate, Auxiliary Trigger Status	111	Pinwheel	179
Load Dividend Immediate (Code 18-LDM)	111	Pinwheel Shaft	179
Load Dividend Immediate, E-Timer Trigger Objectives	111	Power Circuit Breaker (PCB) On	35
		Power Off Sequence	37
Machine Language	6	Power On Reset	37
Magnetic Cores	21	Power On Sequence	35
Main Plate	179	Power Off Sequence	37
Memory	11, 44	Power Supplies	35
Memory Address Register Check	60	Power Supply Overload Trip, SMS DC	37
Memory Address Register Display	60	Power Switch, Console	63
Memory Address Register (MAR)	53	Power Switch On, Console	35
Memory Address Register Storage (MARS)	13, 49	Power Switch Off, Console	37
Memory Addressing and Data Flow	11	Program, Development of	17
Memory Buffer Register Display	60	Program, Execution of	17
Memory Buffer Register — Even Check	60	Program Repetition Program Loop	19
Memory Buffer Register (MBR)	49, 57	Programming	17
Memory Buffer Register — Odd Check	60	Punch Lever Latch and Magnet Armature Restoration	146
Memory Cores	21	Punch-Magnet Yoke Assembly, Paper Tape Punch	148
Memory Data Register Display	60	Punch Operation, Paper Tape Punch	144
Memory Data Register (MDR)	11, 49, 57		
Memory Heater, R105, and R-102 Operation	37	Read Alphanumerically, Auxiliary Trigger Status	159
Microswitch, Tape Guide	179	Read Alphanumerically (Code 37-RA)	158
Microswitch, Tape Tension	179		

	<i>Page</i>		<i>Page</i>
Read Alphanumerically, E-Timer Trigger Objectives	159	Take-up Buffer Arm Contacts	179
Read Check	60	Tape Driving Mechanism	133
Read Head	179	Tape Feed Switch Auxiliary Trigger, Status	150
Read Head Assembly	135	Tape-Feed-Pressure Arm and Adjustable Tape Guide	147
Read Lamp Potentiometer	179	Tape Guide	179
Read Numerically, Auxiliary Trigger Status	156	Tape Guide Assembly	179
Read Numerically (Code 36-RN)	153	Tape Runout Bar	179
Read Numerically, E-Timer Trigger Objectives	157	Tape Tension Arm	179
Read Y Point	79	Tape Tension Bar	179
Reader Ready Lines	140	Timing Disk Photocell Amplifier Lock Circuit	141
Reading Mechanism	133	Timing Disk Potentiometer	179
Reading Tape	139	Timing Disk (Sync Disk)	136, 179
Reel Driving Mechanism	133	Transmit Digit, Auxiliary Triggers	82
Reel Motor, Supply	179	Transmit Digit (Code 25-TD)	82
Reel Operation, Preparing For	136	Transmit Digit, E-Timer Trigger Objectives	82
Reel Power	137	Transmit Digit Immediate, Auxiliary Trigger	82
Reel, Supply	179	Transmit Digit Immediate (Code 15-TDM)	82
Register Displays	60	Transmit Digit Immediate, E-Timer Trigger Objectives	82
Sense Amplifiers and Memory Buffer Register (MBR)	11	Transmit Field, Auxiliary Trigger Status	83
Set Flag, Auxiliary Trigger Status	131	Transmit Field (Code 26-TF)	82
Set Flag (Code 32-SF)	131	Transmit Field, E-Timer Trigger Objectives	83
Set Flag, E-Timer Trigger Objectives	131	Transmit Field Immediate, Auxiliary Trigger Status	83
Sign Analysis, Add	90	Transmit Field Immediate (Code 16-TFM)	83
Sign Analysis, Compare	101	Transmit Field Immediate, E-Timer Trigger Objectives	84
Sign Analysis, Subtract	95	Transmit Record, Auxiliary Trigger Status	84
SMS Card AFR	27	Transmit Record (Code 31-TR)	84
SMS Card NN	29	Transmit Record, E-Timer Trigger Objectives	84
SMS Card NP	27	Trigger 1	74
SMS Card QU	35	Trigger 2	74
SMS Card RE	29	Trigger 3	74
SMS Card TAJ	25	Trigger 4	75
SMS Card VL	28	Trigger 5	75
SMS Card WJ	35	Trigger 6	75
SMS Card WM	28	Trigger 7	75
SMS Card WY	27	Trigger 8	75
SMS Card Extenders	22	Trigger, Branch Test	79, 129, 130
SMS Card Gate	179	Trigger, Carry In	79, 89, 101, 105, 119
SMS Cards — AHK, CAB, CD, DAX, DAW, MX, VE and VF	22	Trigger, Carry Out	79, 89, 100, 105, 117
SMS Cards — CAU, DFD	27	Trigger, Cycle Control	105, 110
SMS Cards — FL, FM and ALY	29	Trigger, DECR 72, 77, 83, 84, 88, 100, 104, 110, 117, 123, 156, 164, 166	79, 84, 101, 125, 165
SMS Cards — PG, PH, PJ, and PK	29	Trigger, Digit/Record Mark	157, 165, 167, 172
SMS Cards — TCZ and DFA	34	Trigger, Disconnect Gate	117
Stop Arm, Tape Guide	179	Trigger, Divide Add	110, 119
Stored Program Concept	7	Trigger, Dividend and Remainder Sign	72, 74, 76
Subtract, Auxiliary Trigger Status	95	Trigger, E-Cycle Entry	177
Subtract (Code 22-s)	92	Trigger, E-Timer (Numbered) See Appendix F	79, 88, 100, 104, 117
Subtract, E-Timer Trigger Objectives	96	Trigger, Equal/Zero	76, 129, 177, 181
Subtract Immediate, Auxiliary Trigger Status	97	Trigger, Execution Cycle (E-Cycle)	78, 83, 88, 100, 105, 110, 117, 123, 126
Subtract Immediate (Code 12-sm)	97	Trigger, Field Mark No. 1	78, 89, 100, 105, 117,
Subtract Immediate, E-Timer Trigger Objectives	97	Trigger, Field Mark No. 2	157
Switch Core Not Selected, Circuit Operation	32	Trigger, Flag	77, 83, 88, 100, 105, 110, 117, 123
Switch Core Selected, Circuit Operation	32	Trigger, First Cycle	117
Switch Core, Selecting	32	Trigger, First Divide Cycle	79, 88, 100, 105, 117
Switch Cores	22	Trigger, High/Plus	157, 159, 165, 169, 170, 172
Switch, Display MAR Selector	62	Trigger, Hold	72, 74
Switch, I/O Check	62	Trigger, i-Cycle	157, 165, 168, 172
Switch, Overflow Check	62	Trigger, I/O Exit	119
Switch, Parity Check	62	Trigger, Last Divide Cycle	165, 167
Switch, Reel Power	179	Trigger, Memory Overflow	165, 169
Switch, Reel-Strip Mode	134, 138, 179	Trigger, No Feed	79
Switches, Check	62	Trigger, Overflow	157
Switches, Console Power	63	Trigger, Read Check	156, 164, 167
Switches, Limit	140	Trigger, Read-Write Call	79, 89
		Trigger, Recomplement	79, 89

	<i>Page</i>
Trigger, Récomplement Control	79, 89
Trigger, Digit/Record Mark	79, 84, 101, 125, 165
Trigger, Response Gate	157, 164, 167
Trigger, Save Control	124
Trigger, Stop 1	157
Trigger, Sync	157, 165, 167
Trigger, Tape Feed	150
Trigger, Tape Feed Interlock	150
Trigger, True/Complement	78, 88, 95, 100, 104, 117
Trigger, Write Check	165, 169
True/Complement (τ/c) Switch	55, 70

	<i>Page</i>
Twenty-four Volt (24v) Sequencing Control, Loss of	38
Typewriter	133
Typewriter and Paper Tape Punch, Auxiliary Triggers	164
Write Alphamerically, Auxiliary Trigger Status	170
Write Alphamerically (Code 39-wA)	169
Write Alphamerically, E-Timer Trigger Objectives	170
Write Check	60
Write Numerically, Auxiliary Trigger Status	164
Write Numerically (Code 38-wN)	159
Write Numerically, E-Timer Trigger Objectives	165