Robert F. Fracchia

DATAmatic 1000

CONSOLE

OPERATOR'S MANUAL

TABLE OF CONTENTS

SECTION V

SECTION VI

SECTION VII

SECTION VIII

SECTION I

SYSTEM DESCRIPTION

In order to better understand the overall role and function of the Console, a detailed description of some of the integral units which constitute the Central Processor should prove beneficial.

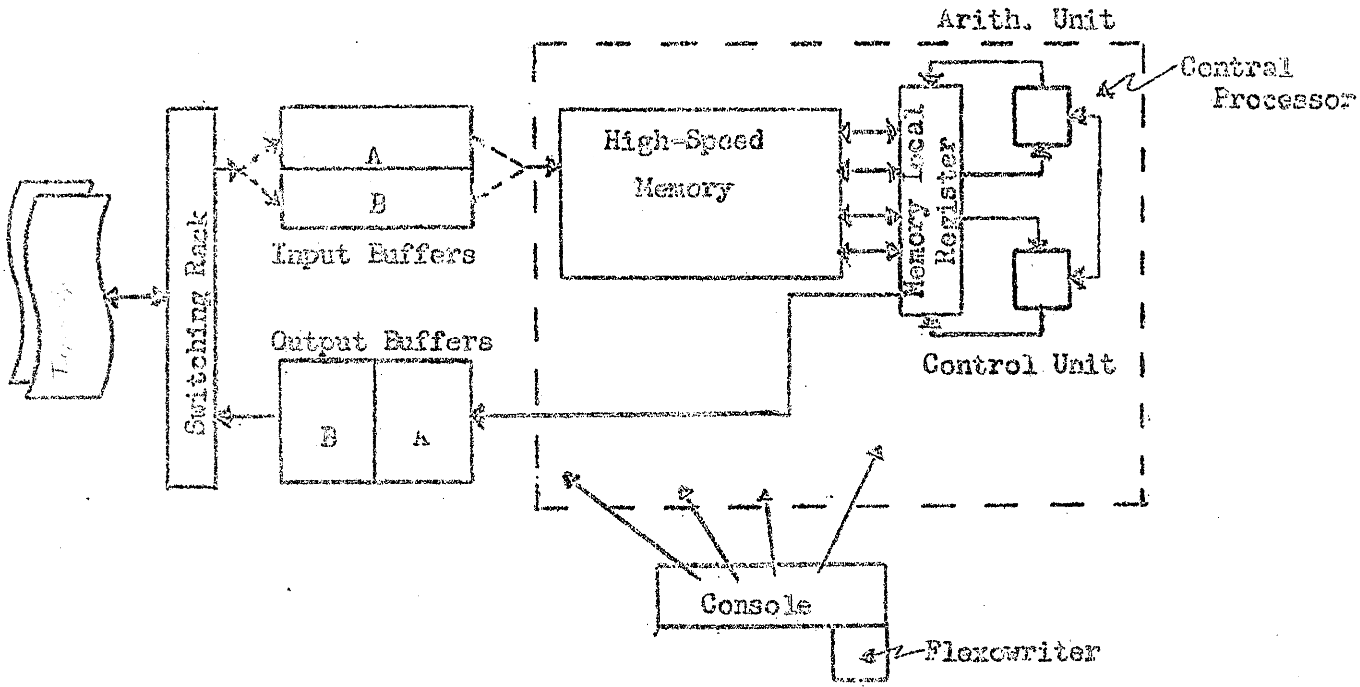


Figure 1

Figure 1 shows a block diagram of the Central Processer and peripheral equipment. Lines connecting any unit with any other unit represent paths of data flow. In some instances (i.e., Arithmetic Unit and Control Unit) the data flow lines are bi-directional indicating data flow in either direction between these units. Figure 1 also indicates that the Console exercises control over the entire system. As seen in Figure 1, the Central Processor is divided into three basic units with the Memory Local Register being a supplement to the High-Speed Memory.

I. HIGH-SPEED MEMORY

The High-Speed Memory contains 2000 locations each capable of handling one 52-bit DATAmatic word. Within the DATAmatic System a word is defined as the smallest group of information that is handled as a unit. The DATAmatic word consists of 48 binary digits (bits) and 4 weight count or check bits. The word can be further subdivided into 12 numerics, 11 numerics plus sign, 8 alphabetic characters, or any combination of alphabetic characters and numeric digits which make up 48 bits. All instructions and data must enter the memory from the input buffer and exit via the output buffers with the exception of words inserted or read directly via the Flexowriter.

The Memory Local Register is an additional one word register (non-addressable) which essentially is a temporary storage location for all words prior to being written into memory or immediately after being read out of memory and prior to being sent to either the Arithmetic or Control Unit. A more complete description of the Memory Local Register will be found under Special Registers.

II. ARITHMETIC AND CONTROL UNITS

The Arithmetic Unit performs all the arithmetic computations for the system.

This includes computations called for by the program and also other automatic sequencing operations as dictated by each instruction. These operations include indexing the sequence register, indexing a counter to determine the number of words transferred, etc.

The Control Unit directs the extraction of the next instruction from the High-Speed Memory, interprets operation codes and generally controls the activity of the Central Processor. As expected, the Control and Arithmetic Units are very closely inter-related and no fine line can divide the two units.

III. TIMING THE DATAmatic 1000

A. Pulse Rate

The heart of the timing system is a crystal-controlled oscillator operating at a fundamental frequency of 2.22 megacycles. (2.22 million cycles per second.) Successive pulses come at this 2.22 megacycle rate and hence the pulses are 0.45 microseconds apart.

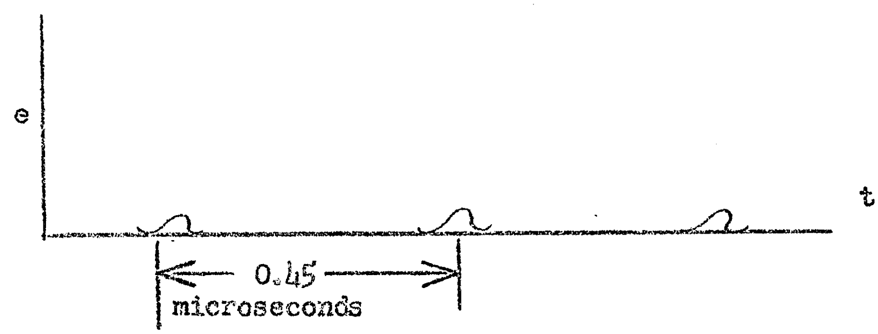


Figure 2

These pulses are amplified and distributed throughout the high-speed portions of the Central Processor. Thus the Computer is internally timed by a never-ending stream of pulses from the crystal-controlled oscillator.

B. Timing Clock

Establishment of a definite starting or reference point is accomplished with a timing clock and associated logical circuits which distribute the time clock pulses. Basically the timing clock and the time clock distributor establish word cycles which are 64 pulses in duration, each pulse being at the 2.22 megacycle rate. (Cycle time equals $64 \times 0.45 \times 10^{-6} = 28.8$ microseconds.) The timing clock initiates the start of each cycle and the time clock distributor supplies pulses to specified logical functions as defined by the instruction.

C. Instruction Timing

Each instruction consists of a number of word cycles. There are in all eight discreet word cycles. Certain instructions utilize all eight cycles while other instructions use only a portion of the eight cycles.

Still other instructions consist of a repetitive use of a particular cycle so that the entire instruction takes longer than eight word cycles.

To time an instruction, one need only multiply the number of cycles times 28.8 microseconds (cycle-time) to arrive at the answer.

For example:	ADD A B C	(8 cycles)	230.4 microseconds
	TIA A 16	4 + n cycles (n = 16)	576 microseconds

Cycle times for each instruction may be found on the instruction summary sheet (see Appendix I).

IV. FUNCTION OF SPECIAL REGISTERS

Of the 2000 registers within core memory, some perform special functions in order to implement an instruction. In addition to this, there are several other non-addressable registers which perform special functions. A summary of the special registers (important to the Console) outlining their specific functions will be given in order to facilitate the understanding of Console operation.

A. The Sequence Register

The Sequence Register is a 52-bit register containing the address of the next-normally executed instruction. The address circulates through this register so that on cycle 2 the address appears in the A subaddress position. The memory designator bit however appears in the B position or bit 50 (i.e., location 0352 would be represented in the Sequence Register as 800 352 000 000 and location 1352 would be B00 352 000 000). The prime function of the Sequence Register is to dictate where the next normal instruction is located.

The Sequence Register can be written into or interrogated via the Flexowriter. This facility is extremely important in emergency situations to help determine program errors and to aid in initiating new starts.

B. Control Register

The Control Register (1990) is a one-word register containing the instruction currently being performed. It directs the selection of the operands from memory and controls where to place the result in memory.

During the execution of an instruction, the addresses of the instruction in the Control Register are permutated. The memory designator bits and the Op Code are not permuted. Thus, at the completion of an instruction, the addresses appear in 1990 in the permuted sequence of C A B rather than in the original A B C order. Furthermore, during the execution of certain instructions, the contents of the Control Register are modified. For example, during a transfer instruction the A operand is modified by one each time a new word is transferred. In the execution of shift instructions the B address is modified as each bit is shifted until the desired number of bits are shifted.

In general, the Control Register is the nerve center of each instruction. This register can also have data inserted manually or be interrogated via the Flexowriter making it a very valuable aid in determining the cause of a program failure. Programs can be restarted by entering an instruction directly into the Control Register and depressing the appropriate button on the Console (i.e., START AT CONTROL REGISTER button).

C. Memory Local Register

The Memory Local Register is a one-word non-addressable temporary storage register. All words which pass into or out of High-Speed Memory pass through the Memory Local Register.

To further qualify this, it is noted that words are read in and out of memory in a parallel fashion, and transferred to buffers and internally within the Central Processor on a serial basis. Therefore, prior to converting serial to parallel or vice versa, the word is temporarily stored in the Memory Local Register. The contents of this register are continually displayed on the Central Console.

D. Operation Code Register

One of the first things that must be determined with every instruction is the operation that the Computer is to perform. When the instruction reaches the Control Register, the eight bits comprising the operation code are immediately transmitted to the eight-bit Operation Code Register.

These eight bits are decoded, producing specific trains of pulses which establish for the Control Unit the particular operation to be carried out.

The Operation Code Register visibly displays this information on the Console at all times; for the present time it is sufficient to say that the Op Code display will always show the current instruction being performed. Thus, in case of a program error, the contents of the Op Code Register are observable for determining the type instruction which caused the error.

E. Current Instruction Register

The Current Instruction Register (1999) stores two types of information depending upon the instruction being processed and which of the eight instruction cycles (see paragraph VI, Instruction Cycles) is being performed.

Instruction entered in normal sequence.

Cycle 1 through 7 - Address of instruction being executed.
Cycle 8 - Instruction just executed.

Instruction chosen by subsequence.

Cycle 2 through 7 - Previous instruction.
Cycle 8 - Instruction just executed.

When the Current Instruction Register contains the address of the instruction being executed, the subaddress portion of this address is located in the A subaddress position. However, the bank designator for this address resides in the B bank designator position, all other bit positions being zeroes. When the Current Instruction Register contains the instruction just executed, the address positions of the instruction are in the order C A B rather than the normal A B C. Thus, the address of the instruction being executed resides in the same bit positions (A address) whether it is a programmed C address or a normal instruction sequence address.

Since this register contains the address of the instruction being processed or the instruction just executed (Cycle 8) its importance can not be underestimated and should be understood completely and used to the fullest extent if a program fails.

1. Auxiliary Register - The Auxiliary Register is a one-word register which can be written into and monitored via the Flexowriter. There is no specific address associated with it and it is selected by the Memory Register Selector of the Console typewriter simply by depressing the button coded AR. During the execution of an instruction, Cycles 2 through 7, the AR is used by the Arithmetic and Control Units to store information or partial results pertaining to the instruction. During Cycle 8, the address of the instruction just processed is in the AR. The previous instruction is stored in AR during Cycle 1 of a normal instruction.

2. Accumulator - The Accumulator can be monitored only by the Console operator. Its contents are dependent upon the instruction being performed and the operation cycle in which it is interrogated. All arithmetic operations make use of the Accumulator. In case of machine malfunction, the ability to monitor the Accumulator can be extremely helpful in trouble-shooting. This facility is of considerable value to the Field Service Engineer.

V. SEQUENCE OF OPERATIONS

Once a program is loaded into High-Speed Memory and started under Automatic Control, a well-defined sequence of events take place as each instruction is executed.

The contents of the Sequence Register are sent to a logical block called the High-Speed Memory Address Selector. There, the numeric code is deciphered into a specific memory location and its contents are read out to the Memory Local Register.

The word in the Memory Local Register is then transmitted to the Control Register where the various bits of the word are interpreted and translated into action.

Assuming an instruction is entered in the normal mode (i.e., selected via the Sequence Register) prior to any action from the Control Register, the Sequence Register sends its contents (i.e., address of instruction being performed) to the Accumulator where a +1 is added to the address. This indexed address is then returned to the Sequence Register.

The interpretation of the instruction in the Control Register proceeds basically in the following manner. The operation code bits are sent to the Operation Code Register where the eight bits initiate action of particular functions within the Control Unit.

The A and B addresses of the instruction are in turn sent to the Address Selectors to allow read out of the operands. The Control and Arithmetic Units carry out the necessary steps initiated by the Operation Code and the result is returned to memory under control of the C address in the Address Selector.

Only DATAmatic words which enter the Control Register (1990) can actually be called instructions in the true sense of the meaning. At this point the Central Processor can distinguish between data and instructions.

Once having completed the instruction as dictated by the Control Register, the Computer cycles back to its starting point, i.e., inspecting the Sequence Register to determine where in memory the next instruction is located. Since the contents of the Sequence Register were incremented by +1, the Address Selectors are now called upon to read out the next-higher memory location to the Memory Local Register. This assumed the previous instruction did not change the sequence.

This process continues in a normal sequence step-by-step until either the program is complete and stops or the program itself causes a deviation. A deviation is one in which there is a sequence change or a subsequence call.

In the case of a sequence change, part of the operation as dictated by the Control Register is the placing into the Sequence Register of an address other than that contained by it. Thus, at the start of the next instruction, the Sequence Register causes the Address Selector to read a memory location specified by the previous instruction. In processing the new order, all steps described previously are repeated.

The Subsequence Call is simply an interruption of the normal sequence. In this case, the contents of the Sequence Register are not interrogated or altered in any way. In processing an instruction which contains a Subsequence Call, the Control Register designates where in memory the next instruction is located. During the Subsequence Call, +1 is not added to the Sequence Register and therefore it will be set to designate the next normal instruction when the normal sequence is resumed.

In summary then, the sequence of operation is controlled by the Sequence Register which is incremented at the beginning of each instruction. The Control Register then takes over and implements the instruction. This instruction may itself alter the Sequence Register, or cause the next instruction to be taken out of sequence.

VI. INSTRUCTION CYCLES

To facilitate an understanding of the internal operations which are necessary to carry out an instruction, a knowledge of individual instruction cycles should be beneficial.

As was mentioned previously, all instructions carried out on the DATAmatic 1000 require varying numbers of word cycles. There are in total eight fundamental cycles involved. All eight, less than eight, or all eight plus repetitive use of some cycles are needed to carry out the complete list of

DATAmatic instructions. For example, addition requires all eight cycles while an internal transfer of one word requires five word cycles.

Some cycles are basic and are used regardless of the instruction while other cycles may be skipped completely by some instructions. Identical cycles need not carry out the same internal operations. That is, the Central Processor will perform completely unrelated or independent operations during a particular cycle depending upon the type instruction being executed.

A. Cycle Description

The following description of fundamental cycles is provided to further qualify the operation of instructions within the DATAmatic 1000.

CYCLE 1

The operations carried out during Cycle 1 are dependent upon whether or not a check condition is sensed (i.e., overflow, end of tape, TWC, etc.). If it is, a special version of Cycle 1 is entered.

1. Sequence Register sends address of instruction to be processed to Address Selector.

1[a]. Check condition sends special address to Address Selector (special Cycle 1).

2. Sequence Register sends address of instruction to be processed to Accumulator where it is incremented by one.

3. Control Register receives instruction to be processed.

4. Current Instruction Register receives address of instruction to be processed.

4[a]. Current Instruction Register receives previous instruction.

5. Auxiliary Register receives previous instruction.

Notes: 1[a]. and 4[a]. are carried out in place of 1. and 4. in the event of a check situation which causes an automatic subsequence.

CYCLE 2

1. Incremented address is sent from Accumulator to Sequence Register.

If Cycle 2 is entered from Cycle 8 or via an automatic Subsequence Call, Cycle 2.1. is omitted.

2. Operation Code Register receives and decodes Op Code.

3. Auxiliary Register 1 is reset to zero.

CYCLE 3

1. Select and process first operand - During a transfer instruction, Cycle 3 is repeated each time a new word is transferred. It is performed as many times as necessary to complete the transfer. The transfer instruction in the Control Register is modified each time a word is transferred. The modification consists of indexing the A address by one as each word is transferred.

CYCLE 4

1. Select and process the second operand.

2. If a Sequence Change instruction, start preparing to change Sequence Register.

3. Pass orders, if needed, are generated during Cycle 4.

CYCLE 5

1. Select third operand.

2. Execute arithmetic type orders.

3. If a Sequence Change instruction, continue work started in Cycle 4.2.

CYCLE 6

1. Complete arithmetic type orders (results are generated).

CYCLE 7

1. Prepare to deliver result of arithmetic type instructions to High-Speed Memory.

CYCLE 8

1. Results of arithmetic type instructions sent to High-Speed Memory.

2. Auxiliary Register 1 receives contents of Current Instruction Register (1999).

3. Instruction just processed goes to Current Instruction Register.

4. Current instruction is sensed for subsequence (Normal go to Cycle 1).

5. If Subsequence Call, place contents of C address (next instruction) into Control Register (1990) and go to Cycle 2.2. If automatic Subsequence Call is made, go to Cycle 1.

Note: Address of subsequence instruction is the C address of the Current Instruction Register (1999). The order of the addresses are now C A B instead of A B C.

Console operators should become as familiar as possible with Cycles 1 and 8. It is during these cycles that the instruction location and the instruction itself are transferred in the Central Processor. In the event of unexpected trouble with a program, the Console operator

CYCLE 1 - Normal (No Subsequence)

Control Register (1990) - New order
Sequence Register - Address of order to be processed
Label Register (1999) - Address of order to be processed
Auxiliary Register - Order just processed

CYCLE 8 - Normal (No Subsequence)

Control Register (1990) - Original order
Sequence Register - Incremented by 1 in cycle 2
Label Register (1999) - Order just processed
Auxiliary Register - Order location

CYCLE 8 - Subsequence Mode
(Cycle 1 is omitted in a subsequence)

Control Register (1990) - Contains new order specified by "C" address
Sequence Register - Remains unchanged
Label Register (1999) - Order just processed "CAB"
Auxiliary Register - Order location prior to subsequence or address of subsequence instruction if more than two ~~or more~~ consecutive instructions are made in subsequence mode.

Cycle 1 -	Control	Register	-- ABC
Cycle 2 -	Control	Register	-- ABC
" 3 -	"	"	-- BCA
" 4 -	"	"	-- CAB
" 5 -	"	"	-- CAB
" 6 -	"	"	-- CAB
" 7 -	"	"	-- CAB
" 8 -	"	"	-- CAB

should be aware of which memory locations he can interrogate for pertinent information. In addition, it should be noted that the Memory Local Register may be a source of valuable information. It will have as its contents the last word transferred between memory and Central Processor or buffers be it a transfer into or out of memory.

Finally, it should be noted that at the completion of an instruction, Cycle 8, the following are key locations within High-Speed Memory:

Location	Contents
Auxiliary Register No. 1	Location of instruction just processed
Sequence Register	Location of next instruction in normal sequence
Current Instruction Reg.	Instruction just processed
Control Register	Dependent upon instruction just processed If Normal - instruction just processed If Subsequence Call - Next instruction If Comparison is met - Next instruction

Figure 3 is a diagrammatic description of how instructions are implemented on the DATAmatic 1000.

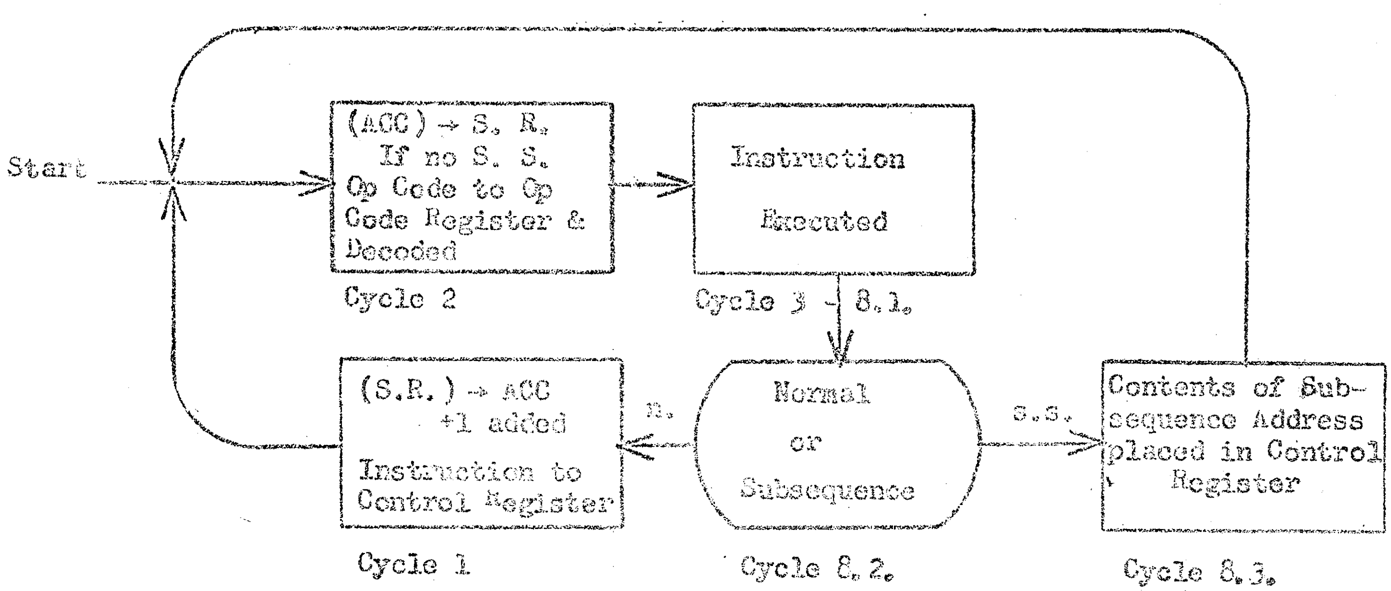


Figure 3

On the Console panel there is a cycle counter which enables the operator to determine at a glance on which cycle an instruction stopped. In general, instructions go to Cycle 8 with the following exceptions:

1. Stops on Cycle 3 of transfer orders when a TWC occurs and ERROR SUBSEQUENCE switch OFF.

2. Stops on Cycle 7 of arithmetic type instructions in the event of invalid operands (unsigned numbers).

SECTION II

FUNCTION OF THE CONSOLE

The DATAmatic Console provides complete supervisory and monitoring facilities over the operation of the entire electronic data processing installation. All activities pertaining to the reading, processing and writing of data are controlled at the Console. At the Console, the operator selects tapes to be read from or written upon, loads the program, initiates starts, performs manual intervention and checks out new programs. It is also evident that the Console is the sole on-line link of communication between the electronically-controlled processing of data and the outside world.

One of the important functions of the Console is its ability to communicate with the operator. It is evident that when a program is operating at electronic speeds, the process should not be interrupted. However, in the event of machine error, the Console should relay to the operator the necessary information to allow him to restart the program as soon as possible.

In most cases, it is possible to distinguish between a machine check stop and a program error. If a program error, the interrogation of the key locations often leads to the spotting of errors and allows for an immediate correction. If the operator definitely ascertains a machine check stop, an engineer should be called promptly. Therefore, in trouble-shooting a machine check or a programming error, the ability to interrogate registers and manually insert data is very important.

To summarize, the Console directs the overall processing operation, forms a link between the Central Processor and the programmer to allow for manual intervention and relays information in the event of errors to determine whether the program stopped due to a programming error or a machine check stop.

SECTION III

DESCRIPTION OF THE CONSOLE

The DATAmatic 1000 Console has two basic main sections. One section houses the active controls which allow control over starting and stopping the Central Processor and manual insertion of data via the Flexowriter. The second section comprises a set of display lights which allow for a passive communication from the machine to the operator. In essence, the display lights relay to the operator the current status of the DATAmatic 1000 as it proceeds through a problem. Both the active controls and the display lights represent a powerful tool which are used effectively in determining errors in programs and machine check functions.

The active controls include four independent starting switches which can be utilized in a number of ways. In general, two of the starts are reserved for automatic re-runs in the event that some unexpected error occurs. Operators should be careful when initiating programs since depressing the incorrect start button could conceivably destroy valuable information thus necessitating a new start.

An AUTOMATIC-MANUAL switch provides the facility for either manually stepping through a program, monitoring it instruction-by-instruction or automatically sequencing through a program at machine speeds.

The Console is provided with an array of OPTIONAL STOP BREAKPOINT CONTROLS which are very useful when checking out new programs. Clear concise instructions should accompany all programs describing the procedures to be followed at each optional stop.

On the top of the Console are a set of display lights and switches which give the status of each Tape Drive Unit in the system.

The Flexowriter, one of the components of the Console, is that unit which implements the interrogation and insertion of data into High-Speed Memory.

The display lights basically represent the current status of the Computer. During automatic operation these lights change too rapidly to indicate any particular condition. During a stop, however, these lights become an important source for determining the cause of the stop. In general, if a stop occurs, it is the operator's function to determine whether the machine stop was due to an optional stop instruction, a transfer weight count error, a machine check function, etc. The display lights can, in most cases, yield this information. For more specific information, the active controls would now have to be utilized.

In summary, the active controls on the Console used in conjunction with the Flexowriter and its associated Memory Register Selector permit manual control of operations on the DATAmatic 1000. They afford control over starting and stopping the machine and manual insertion of data.

Furthermore, the indicator lights represent a passive communication facility in that they display a continuous picture of the status of the DATAmatic 1000 as it processes a program instruction-by-instruction. This property, in particular, is an exceptionally useful diagnostic tool for program debugging.

SECTION IV

DETAILED DESCRIPTION OF CONSOLE

This section contains a detailed description of the function and use of all switches and display lights on the Central Console. As each switch or display light is described, reference should be made to Figure 4. The description is in two parts; the active controls, and the display lights.

I. OPERATOR CONTROL SWITCHES

A. CLEAR Buttons

1. CLEAR BUFFER Switch - The CLEAR BUFFER switch will cause the output buffer and High-Speed Memory location 1992 to be filled with 48 zero bit words with the appropriate weight counts. It also resets the output buffer counter.

USE: Before starting new program.

2. MACHINE CLEAR Button - The MACHINE CLEAR button resets all error interlocks which have caused the DATAmatic 1000 to stop. The High-Speed Memory and Accumulators are not changed.

USE: Before starting new program.
Before restarting program which has stopped due to interlock error.

B. START Buttons

1. CONTROL REGISTER Button (1990) - Depressing this button will initiate the execution of the instruction currently positioned in

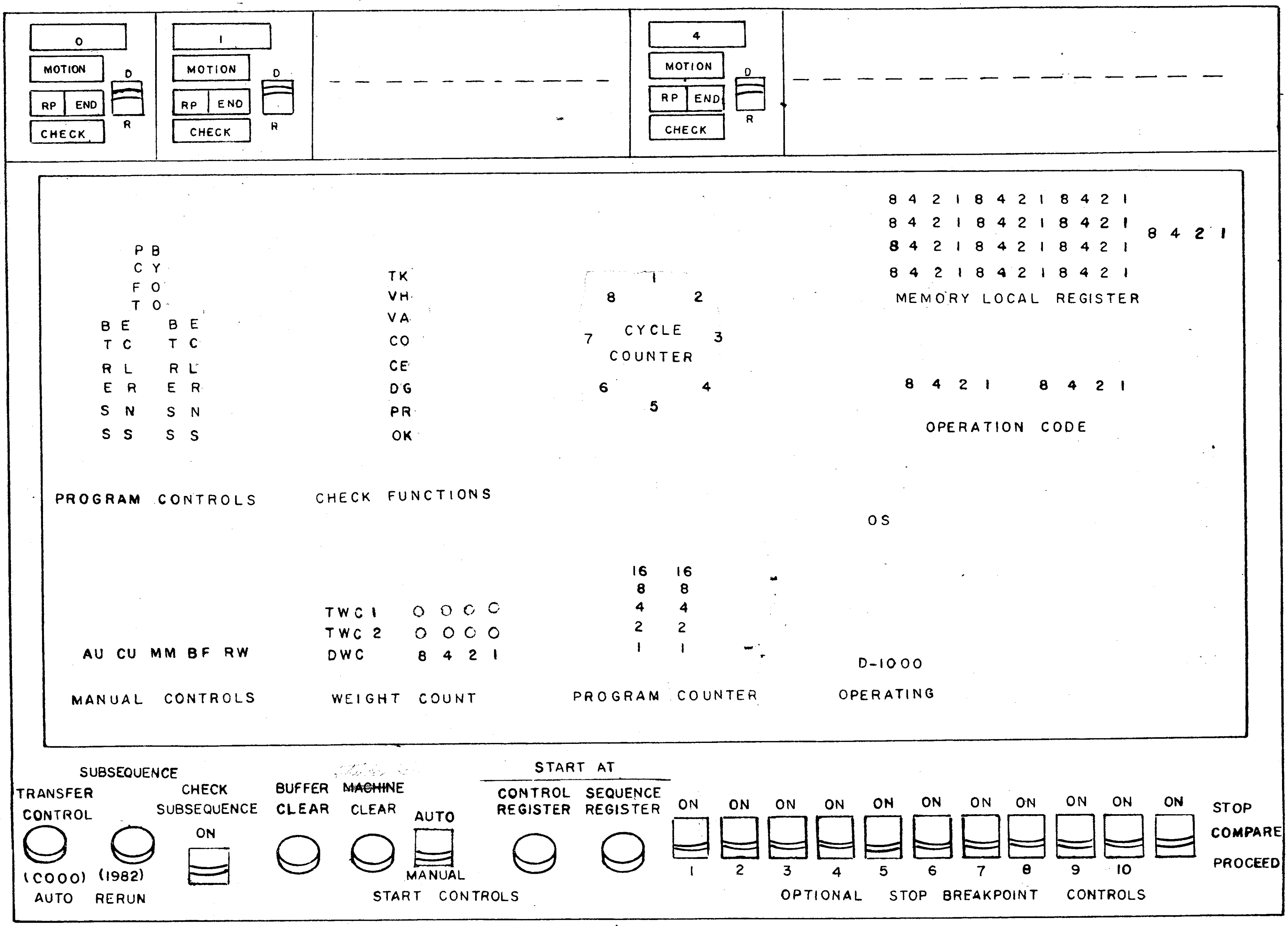

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
PB
CY
FO
TO
BE
TC
RL
ER
SN
SS
BE
TC
RL
ER
SN
SS
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(C000) AUTO
(1982) RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 4.

the Control Register. If the AUTOMATIC-MANUAL switch is in the AUTOMATIC position, the program will continue to operate under control of the stored program. If the AUTOMATIC-MANUAL switch is in the MANUAL position, the instruction in the Control Register will be executed and the next instruction selected will be placed in the Control Register. The DATAmatic 1000 will then stop.

2. SEQUENCE REGISTER Button - Pressing this button causes the machine to place in the Control Register the instruction whose address is in the Sequence Register. If the AUTOMATIC-MANUAL switch is in the AUTOMATIC position, the DATAmatic 1000 will continue to operate under control of the stored program. If the AUTOMATIC-MANUAL switch is in the MANUAL position, the Computer will stop with the instruction in the Control Register but the instruction will <u>not</u> have been executed.

3. TRANSFER CONTROL Button - Pressing the TRANSFER CONTROL button will cause the DATAmatic 1000 to place the instruction at location 0000 into the Control Register. If the AUTOMATIC-MANUAL switch is in the AUTOMATIC position, the Computer will continue to operate under control of the stored program. If the AUTOMATIC-MANUAL switch is in the MANUAL position, the Computer will stop with the instruction from location 0000 in the Control Register, but the instruction will <u>not</u> have been executed.

4. SUBSEQUENCE 1982 Button - Depressing the SUBSEQUENCE 1982 button causes the Computer to subsequence to location 1982. The position of the AUTOMATIC-MANUAL switch will have the same effect on this button as described above under TRANSFER CONTROL except that the instruction will come from location 1982 in the subsequence mode. These two starts, TRANSFER CONTROL and SUBSEQUENCE 1982 are generally reserved for automatic re-run procedures.

C. STOP Buttons

1. OPTIONAL STOP BREAKPOINT CONTROLS - The eleven switches in the lower right section of the Console are the OPTIONAL STOP BREAKPOINT CONTROLS. The rightmost switch determines the STOP, COMPARE, PROCEED option of this instruction. The remaining 10 switches define the possible optional stop selections. The optional stop is dependent upon three conditions namely: instruction composition; setting of the conditional stop switch; the position of the applicable breakpoint switches. The Console allows for 10 independent optional stops. Each time the Computer halts as a result of executing an optional stop instruction, register 1990 will contain the next instruction to be executed. The optional stop instruction will be in one of two locations depending upon whether or not the instruction has a Subsequence Call.

a. No Subsequence Call - Stop on Cycle 1, instruction in AR 7

b. Subsequence Call - Stop on Cycle 8, instruction in 1999

Figure 5 and Figure 6 illustrate the results of various settings of the conditional and breakpoint switches.

Digit	D_b	B_h				B_t						B_u	
	X	X	X	X	X	X	X	X	X	X	X	X	X
Name	Not Used	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_g	B_s
Function	Arbitrary	Correspond to 10 breakpoint switches on the Console											Stop if 1

Figure 5

STOP –

B_s	B_g	Conditional Stop Switch	Breakpoint Switch	Result
1	*	*	*	Stop
0	1	*	*	Proceed
0	0	Stop	*	Stop
0	0	Proceed	*	Proceed
0	0	Compare	**	-

* has no effect

** coincidence of a binary one in bit positions B_1 to B_{10} and an ON of the corresponding breakpoint switch will stop the machine

Figure 6

D. CHECK SUBSEQUENCE Switch

1. TWC Check - During the transfer of each word within the DATAmatic 1000, the weight count is recomputed and compared. If a weight count error is detected at any time except when reading from the buffers, the DATAmatic 1000 will stop. This insures valid operations and data transmission within the Central Processor at all times.

If a weight count error occurs when reading from a buffer and the CHECK SUBSEQUENCE switch is ON, the Computer will automatically take the next instruction from location 1987. If the CHECK SUBSEQUENCE switch is OFF, the machine will stop and appropriate lights will indicate the reason.

E. Magnetic File Unit (MFU) Switches and Display Lights

1. DEMAND-RELEASE Switches - Along the top of the Console are switches and display lights which correspond to the various Magnetic File Units (see Figure 7). Each switch corresponds to a particular Magnetic File Unit and is referred to as the DEMAND-RELEASE switch. In the DEMAND position, the Central Processor has absolute control over the designated Magnetic File Unit regardless of any subsequent attempt by peripheral equipment to use it. If the switch is in the RELEASE position, any peripheral unit can use the MFU. Changing the switch position from RELEASE to DEMAND will not necessarily give control to the Central Processor.

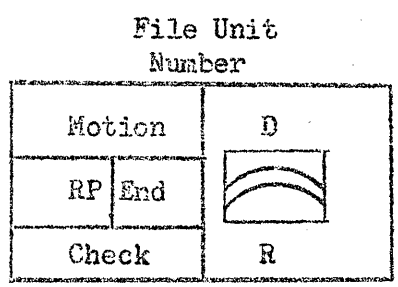


Figure 7

If the File Reference Unit is using the designated MFU, it will yield to the Central Processor. However, if any other piece of peripheral equipment is using the MFU, the Central Processor will not have control until that peripheral unit releases it.

2. MFU Lights - Associated with each MFU are five status lights which indicate various operating conditions of the MFU. The File Unit number indicates the address by which the MFU is recognized by the Central Processor. An appropriate indication is also displayed indicating whether any peripheral unit is currently using an MFU. For example, if the Input Converter is using an MFU, the File Unit number will display 01.

The MOTION light when ON, indicates that an MFU is moving because of a drive command such as Read, Write, or Rewind. Thus, if the Computer has come to an apparent stop with a read or write operation code in the Op Code register, it would be wise to check the MOTION light. If it is ON and not flickering, more than likely the read or write order is waiting for the previous drive command to be completed.

It is interesting to note that the MOTION light is an analog device. That is, the brighter the light glows, the more continuous the motion of the tape. Therefore, a good indication of the efficiency of a program is the relative brightness of the MOTION lights.

The END light is ON when the tape is rewound to its physical beginning. The RP light ON indicates that the designated MFU is now operating in the Run Protected Mode, i.e., the tape may be read from, but not written upon. If an attempt is made to write on a "protected tape", the writing will not take place, the tape will advance by one block and unsatisfied checks will cause the DATAmatic 1000 to stop the next time the tape is addressed. Switching to and from the Run Protected status is accomplished by a switch at the MFU and not at the Console.

The CHECK light for each Magnetic File Unit indicates ON when an instruction to the file unit was executed incorrectly. The Computer will not stop immediately, but rather will continue operating until the same Tape Unit receives a subsequent instruction.

In the event a Read Forward instruction is interpreted as Read Backward, a TWC error would result when the buffer information is subsequently transferred into memory. Before a corrective positioning procedure can be initiated, the MACHINE CLEAR button must be depressed to clear the interlocks.

II. CONSOLE DISPLAY LIGHTS

On the display panel are lights where an operator can observe the status of a problem being solved. The principal use of these display lights are for error detection in new problems and machine trouble analysis.

A. MEMORY LOCAL REGISTER

In the upper right corner of the display panel is a set of 52 lights representing the MEMORY LOCAL REGISTER and grouped as shown in Figure 8.

```
8421 8421 8421
8421 8421 8421
8421 8421 8421   8421
8421 8421 8421
```

Figure 8.

The leftmost light (8) in the top row represents the 52nd bit of a DATAmatic word, the next light the 51st, etc. The numbers represent the assigned weight for each bit position. The four rightmost lights are the weight count bits.

When a machine stop occurs, the MEMORY LOCAL REGISTER will contain the last transferred word either into or out of High-Speed Memory. If the Computer stops due to a TWC error when transferring into memory from the buffer, the invalid word with its incorrect weight count will be in the MEMORY LOCAL REGISTER.

B. OPERATION CODE REGISTER

The OPERATION CODE REGISTER has a set of associated display lights, grouped directly below the MEMORY LOCAL REGISTER (see Figure 9). This

```
8421 8421
OPERATION CODE
```

Figure 9.

eight-bit register displays in hexadecimal notation the operation code of the last instruction performed provided a normal halt is encountered in Cycle 8 or Cycle 1. In the event the stop is between Cycle 2 and Cycle 7, the Op Code of the current instruction being performed is in the OPERATION CODE REGISTER.

C. OPTIONAL STOP Light

The OS light located below the Operation Code Register will be ON only if the machine has stopped on an Optional Stop instruction (see Figure 10).

OS

Figure 10.

D. D-1000 OPERATING

The D-1000 OPERATING light (see Figure 11) is ON when the DATAmatic 1000 is operating in the Automatic mode. It is OFF during the manual operation except for an occasional flicker during a one-shot Print instruction. It is also ON when instructions are entered via the Flexowriter.

D-1000
OPERATING

Figure 11.

E. CYCLE COUNTER

The CYCLE COUNTER display lights (see Figure 12) are located in the center of the Console panel and reflect successive cycles in the performance of a particular instruction. In the event of a stop, these lights will indicate the last completed cycle. Instructions which make repetitive use of the same cycle, such as transfer instructions, may not complete all the repetitions of this cycle when the computer stops. For

1

8 2

7 CYCLE COUNTER 3

6 4

5

Figure 12.

example, a TWC error during a transfer in operation will stop the Computer during Cycle 3 if the CHECK SUBSEQUENCE switch is OFF.

F. PROGRAM COUNTER

Directly below the CYCLE COUNTER is a series of lights reflecting the status of the PROGRAM COUNTER (see Figure 13). The PROGRAM COUNTER is used during the execution of instructions which make repetitive use of particular cycles. For example, during an n-word transfer, Cycle 3 is repeated for each word until the entire transfer has been completed. This counter is a five-stage binary counter. In the

16	16
8	8
4	4
2	2
1	1

PROGRAM COUNTER

Figure 13.

execution of an n-word transfer instruction, the program counter initially has 32-n (binary 32 = 00000) as its contents. Each time a word is transferred, the PROGRAM COUNTER is incremented by one. When the last word is transferred, the counter is automatically reset to 00000. In the event a transfer weight count error stops the Computer, CHECK SUBSEQUENCE switch OFF, the number of words being transferred minus the 32's complement of the PROGRAM COUNTER is the word with the incorrect weight count. See example next page.

Assume 10 word transfer.

PROGRAM COUNTER starts at 32-n = 22. It then transfers the first word and is incremented to 23.

Therefore,

PROGRAM COUNTER Start	Words Transferred	PROGRAM COUNTER Finish
22	1	23
23	2	24
.	.	.
28	7	29
.	.	.
31	10	0
0	-	

Assume the seventh word transferred is the incorrect word. If the CHECK SUBSEQUENCE switch is OFF, the Computer will stop with the PROGRAM COUNTER = 29.

Therefore,

$10 - (32-29) = 10 - 3 = 7$

or in general, $N + P.C. - 32$ = bad word

where N = number of words to be transferred.

The PROGRAM COUNTER is also used in multiply and divide instructions and also the shift instructions.

In general, it is far easier to determine the incorrect word by monitoring the contents of 1990 and 1999.

G. WEIGHT COUNT Checks

To the left of the PROGRAM COUNTER lights are three WEIGHT COUNT checks (see Figure 14). Each word, be it an instruction or data, is checked for a valid weight count every time it is transferred within the Central Processor. The transfer weight adders verifies the validity of each word generating a new weight count and comparing it with the original.

The TWC is arranged so that a remainder of six indicates verification from the addition of bits 5 through 52 reduced modulo 9.

$$\frac{(15 \text{ minus weight count bits } 1 - 4) + (\text{weight count bits } 5 - 52)}{9} = 6$$

Therefore, <u>whenever</u> a word is transferred, 0110 must exist in TWC1 and TWC2, or the Computer will indicate a weight count error. Because of the dual transmission circuitry in the DATAmatic 1000, it is entirely possible to be transmitting two completely independent words and therefore, the use of the two transfer weight count checks is required. TWC1 represents the High-Speed Bus. TWC2 represents the Auxiliary High-Speed Bus.

The row of lights directly below labeled DWC refers to the Decoded Weight Check. The binary display of 0110 of the Decoded Weight Check verifies that each address to be processed in a given operation has been accurately selected and utilized. If the display is other than 0110, the machine will stop on a DWC error.

TWC1 0000
TWC2 0000
DWC 8421

Figure 14

These three rows of lights are arranged on the Console panel so that the two center lights are green and the end lights are red. Thus, the green lights indicate all checks have been satisfied and a red light indicates a check situation.

H. <u>CHECK FUNCTIONS</u>

Directly above the WEIGHT COUNT checks are the CHECK FUNCTION display lights. Their prime use is for diagnostic procedures in case of machine malfunction. Below is a description of the CHECK FUNCTION lights.

Check Function	Description
TK	Timing Clock Check is normally ON. OFF when the Timing clock, which supplies the necessary pulses for timing the performance of each individual phase of machine operation, fails to function properly.
VH	Void Transmission, High-Speed Bus is normally ON. OFF if the transmission circuit through the High-Speed Bus is opened and an invalid (weight count of zero) word is transmitted, but TWC1 and TWC2 will still be 0110.
VA	Void Transmission, Auxiliary Bus is normally ON. OFF is the transmission circuit through the Auxiliary Bus is opened and an invalid (weight count of zero) word is transmitted, but TWC1 and TWC2 will still be 0110.
CO	Operation Code is normally ON. OFF if invalid operation code is detected in the Operation Code Register.
CE	Clocking Error is normally OFF. On if pulsing clock associatedwith the writing of information from the Output Buffer onto tape incurs a malfunction.
DG	Duplication Good is normally ON. Will be OFF if the duplication circuits which check the Key control devices, such as the Subsequence or Sentinel Circuits, malfunction.
PR	Proceed will be ON if, when proceeding automatically, the machine is set to perform the next instruction. This light will be OFF in the event of any type of stop.
OK	Okay will be ON only if the machine is operating under automatic mode of operation. Flickers when entering data from the Flexowriter.

To the left of the Check Function indicators is a bank of lights labelled PROGRAM CONTROLS which, in general, provide a means for detecting specific types of errors in a new program. They are described briefly below.

Program Control	Description
PB	Printer Busy, ON when the Flexowriter is in use.
CY	Cycle Counter will be ON if a malfunction is detected in the Cycle Counter Circuitry. For example, if two cycles came up at the same time in the execution of an order.
FO	False Overflow will flicker ON if an overflow occurs in a DIV instruction, i.e., when the numerator is greater than the denominator in absolute value. This light will also flicker during the execution of an ADD or SUB instruction under certain conditions.
TO	True Overflow flickers ON when during an ADD or SUB an excessive accumulation results, and the machine capacity is exceeded.
BE	Buffer Error will be ON only if an error occurs in the Input Buffer circuitry.
TC	Transfer Control flickers ON when the contents of the Sequence Register are replaced during a transfer of control.
RL	Ready Line, ON when the tape called for in a Read, Write, or Search instruction is available.
ER	Error Subsequence to 1987 flickers ON only if the ERROR SUBSEQUENCE switch is ON and a subsequence is made.
SN	Sentinel flickers ON when a Sentinel is sensed.
SS	Subsequence, ON when machine is in subsequence mode of operation.

I. MANUAL CONTROLS

Directly below the PROGRAM CONTROLS in the lower left corner are five display lights called MANUAL CONTROLS (see Figure 15). The MAN-

UAL CONTROL lights represent a set of interlocks on the AU (Arithmetic Unit), CU (Control Unit), MM (High-Speed Memory), BF (Input or Output Buffers), RW (Read-Write Switching Rack). If any of the manual switches in any of these units have been mispositioned, the appropriate interlock will be set and the corresponding light will be turned OFF.

AU CU MM BF RW

MANUAL CONTROLS

Figure 15.

J. <u>Flexowriter</u>

The Flexowriter is the link which facilitates communication with the Computer. Its main function is to insert data and monitor specific memory locations and also to be used as an intermediate on-line printer. Directly below and in front of the Flexowriter keyboard are eight control switches (see Figure 16). These essentially control the type of information going into or out of the Central Processor. The Manual Register Selector which is used in conjunction with the Flexowriter is a separate control panel which will be described later.

1. PRINT STYLE - This switch is not effective (see Chart 1) during programmed print orders. It is only effective when reading or writing ~~to~~ from and ~~from~~ to memory.

a. Alphabetic

~~<u>Printing</u>~~ <u>Reading</u>: Breaks computer word into 6-bit groupings with a 4-bit weight count, printing A through Z and any of the 16 hexadecimal characters.

<u>Writing</u>: Allows insertion into memory of all alphabetics plus the 16 hexadecimal characters, inserting 00 in the 5th and 6th binary positions for the numerics. (Note: eight characters must always be entered or interlocks will prevent writing in.)

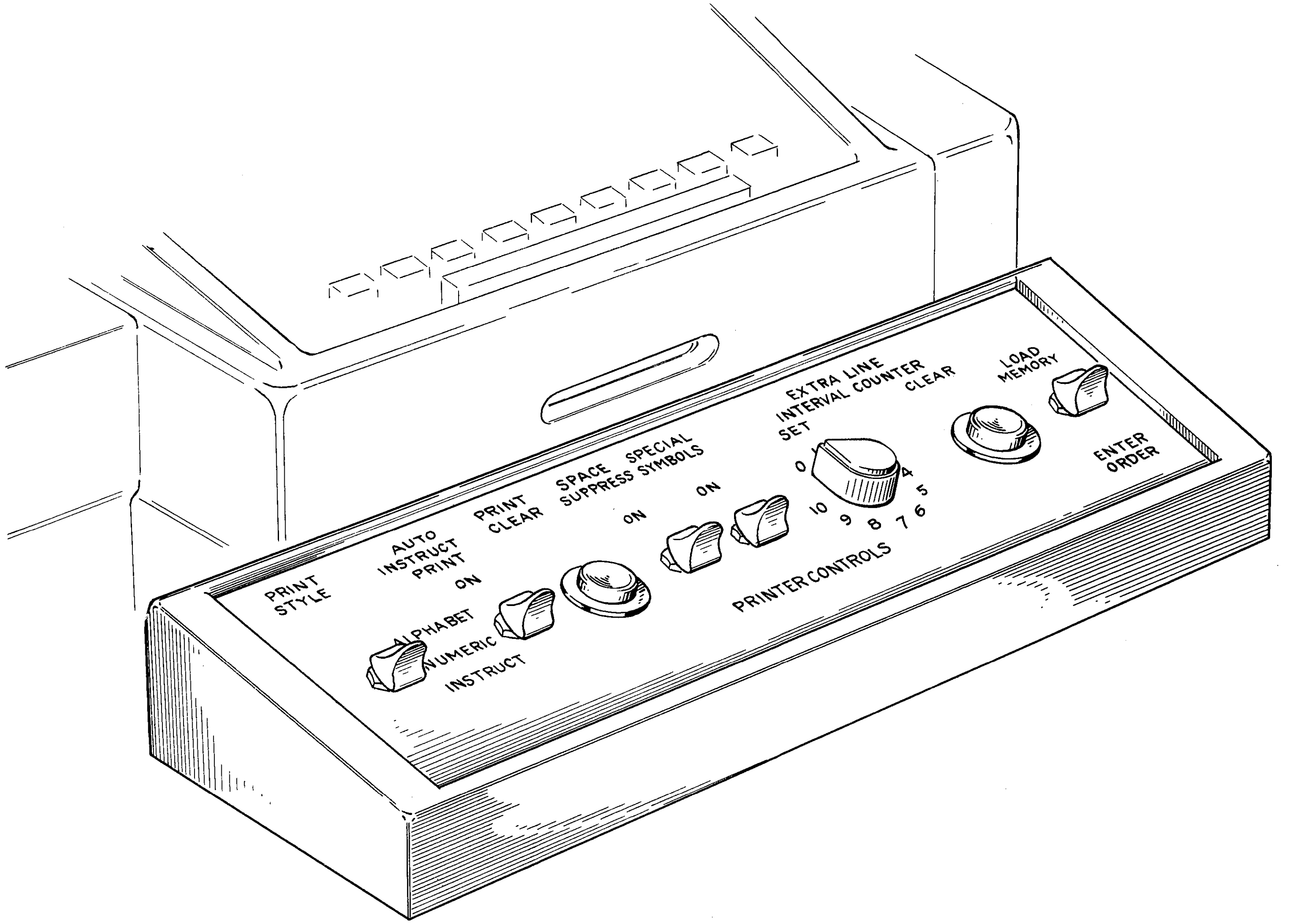


Figure 16. Flexowriter Switch Panel

SWITCH OR BUTTON	PROGRAMMED PRINT ORDERS			MANUAL PRINT—OUTS			
	NUMERIC	ALPHABETIC	INSTRUCTION	NUMERIC	ALPHABETIC	INSTRUCTION	NU
PRINT STYLE	← INACTIVE →			SET TO NUMERIC	SET TO ALPHABETIC	SET TO INSTRUCTION	SE NU
AUTO INSTR. PRINT	INACTIVE	SET TO "OFF"	SET TO "ON"	←			
PRINT CLEAR	← FOR EMERGENCY USE ONLY →						FO
SPACE SUPPRESS	← OPERATIVE →		INACTIVE	← OPERATIVE →		←	
SPECIAL SYMBOLS	← OPER →	INACTIVE	← OPERATIVE →		INACTIVE	←OPERATIVE→	←
EXTRA LINE INTERVAL SET	← OPERATIVE →			←			
EXTRA LINE INTERVAL CLEAR	← OPERATIV						
LOAD MEMORY ENTER ORDER	← INACTIVE						
MANUAL REGISTER	← INACTIVE →			← OPERATIVE			
COMMENTS	AUTOMATIC OPERATION			MANUAL OPERATION UN IMMEDIATE OPERATOR			

CHART I —— CONSOLE FLE

MANUAL WRITE—INS			LOAD MEMORY			ENTER ORDER		
ERIC	ALPHABETIC	INSTRUCTION	NUMERIC	ALPHABETIC	INSTRUCTION	NUMERIC	ALPHABETIC	INSTRUCTION
T TO IERIC	SET TO ALPHABETIC	SET TO INSTRUCTION	SET TO NUMERIC	SET TO ALPHABETIC	SET TO INSTRUCTION	SET TO NUMERIC	SET TO ALPHABETIC	SET TO INSTRUCTION
INACTIVE								
R CLEARING FLEXOWRITER N THE EVENT OF TYPING VRONG CHARACTER			FOR EMERGENCY USE ONLY					
INACTIVE								
INACTIVE								
INACTIVE								
E								
			SET TO "LOAD MEMORY"			SET TO " ENTER ORDER "		
			OPERATIVE FOR INITIAL LOCATION			INACTIVE		
DER CONTROL			MANUAL OPERATION UNDER PAPER TAPE CONTROL					

XOWRITER CONTROLS

Note that more than the specified number of characters or numbers will clear the board and cause an automatic reset.

b. Numeric

Reading ~~Printing~~: Breaks the computer word into 12 four-bit groupings. The printed symbols are 0 through 9 and a choice of special symbols for the six non-decimal configurations.

Writing: Allows insertion into memory of all 16 four-bit configurations. In the event an alphabetic character is accidently typed, the 5th bit from the right is tested.

If it is a one, it replaces the 4th bit and the rightmost four bits, as corrected are retained.

If the 5th bit is a zero, however, the 4th bit is unchanged and the rightmost four bits are again retained.

This results in the following:

Alphabetic Letter Typed	Entered Binary Configuration	Hexadecimal Equivalent
A	1001	9
B	1010	B
C	1011	C
D	1100	D
E	1101	E
F	1110	F
G	1111	G
H	1000	8
I	1001	9
J	0001	1
K	0010	2
L	0011	3
M	0100	4

Alphabetic Letter Typed	Binary Configuration	Hexadecimal Equal
N	0101	5
O	0110	6
P	0111	7
Q	1000	8
R	1001	9
S	1010	B
T	1011	C
U	1100	D
V	1101	E
W	1110	F
X	1111	G
Y	1000	8
Z	1001	9

Therefore, before numeric words are entered they should be checked for accuracy (Note: 12 characters must always be entered in the numeric mode or interlocks will prevent writing).

Note that more than the specified number of characters or numbers will clear the board and cause an automatic reset.

c. Intruction - The instruction mode is provided to facilitate both printing out or inserting instructions. The most convenient pattern of machine instruction is

$$D_s \ OC \ D_a \ A \ D_b \ B \ D_c \ C \ WC.$$

When the ~~printer~~ Flexowriter reads a word from the machine under control of this switch setting, it automatically regroups the designator digits and prints in the above form. If the operator writes an instruction in the above form, the Flexowriter control circuits will automatically group the bits D_s D_a D_b D_c into a single hexadecimal character, D, and form a machine word in the form DOC ABC WC.

2. AUTOMATIC INSTRUCTION PRINT Switch - The AUTOMATIC INSTRUCTION PRINT switch is only effective for programmed print orders of the alphabetic type (PRA). In the OFF position, the Flexowriter will print out eight alphabetic characters. In the ON position, the printout will be an instruction of the following format.

D_s OC D_a A D_b B D_c C W.C.

Example,

FRA 0250 --- --- (Contents of 0250 ADD 0032 1115 0116)

0250	1010	1110	1110	0000	0011	0010	0001	0001	0101	0001	0001	0110
Auto- Instr. ON	1	F	F	0 0	3	2 1	1	1	5	0 1	1	6
Auto- Instr. OFF	Bl.		Bl.	Bl.		S	Bl.		E	Bl.		F

As can be observed from the above example any time a six-bit configuration does not correspond to a type key a blank space will be printed.

3. PRINT CLEAR Button - The PRINT CLEAR button is for emergency use only. For example, if a wrong character is typed or if the keys jam, the PRINT CLEAR button should be depressed.

4. SPACE SUPPRESS Switch - The SPACE SUPPRESS switch has two positions as described below. The SPACE SUPPRESS position is effective only for manual reading and programmed print orders in the numeric or alphabetic mode. The spacing described under normal will always occur when ~~writing~~ manual reading or writing is done from the Console or when ~~an~~ a programmed print instruction ~~type word is read or written~~. is executed

a. Normal Spacing is as shown, depending upon the word type.

Numeric:	DDD DDD DDD DDD	W.C.	LOC. WC
Alphabetic:	AA AA AA AA	W.C.	LOC. WC
Instruction:	$A_s OC_1 OC_2 D_a A_h A_t A_u D_b B_h B_t B_u D_c$	W.C.	LOC. WC

b. Space Suppression causes characters to be typed in sequence without any automatic spacing. in manual or programmed modes The weight count and address is not printed during manual write in with Space Suppression ON.

5. SPECIAL SYMBOLS Switch - This switch is effective when reading or writing in the instruction or numeric mode either manually or automatically (under program control). It has two positions.

a. Normal will cause the hexadecimal representations 10 through 15 to be printed as alphabetic characters B through G.

b. Special Symbols will cause the hexadecimal representations 10 through 15 to be printed as below. In this position it will also cause the four-bit code 1101 in P_{52} through P_{49} to print as + (plus) and 0101 to print as - (minus).

(10)	dollar sign	$
(11)	comma	,
(12)	decimal point	.
(13)	plus sign	+
(14)	blank	
(15)	asterisk	*

6. EXTRA LINE INTERVAL COUNTER SET Switch - This switch determines the number of lines which are printed consecutively before a space between lines is emitted. It is only active during programmed print instructions. (The number of lines already counted in a block is indicated in lights around the switch.) As many as ten lines may be printed between spaces.

7. EXTRA LINE INTERVAL COUNTER CLEAR Button - This button may be depressed at any time if necessary. It resets the line counter to zero.

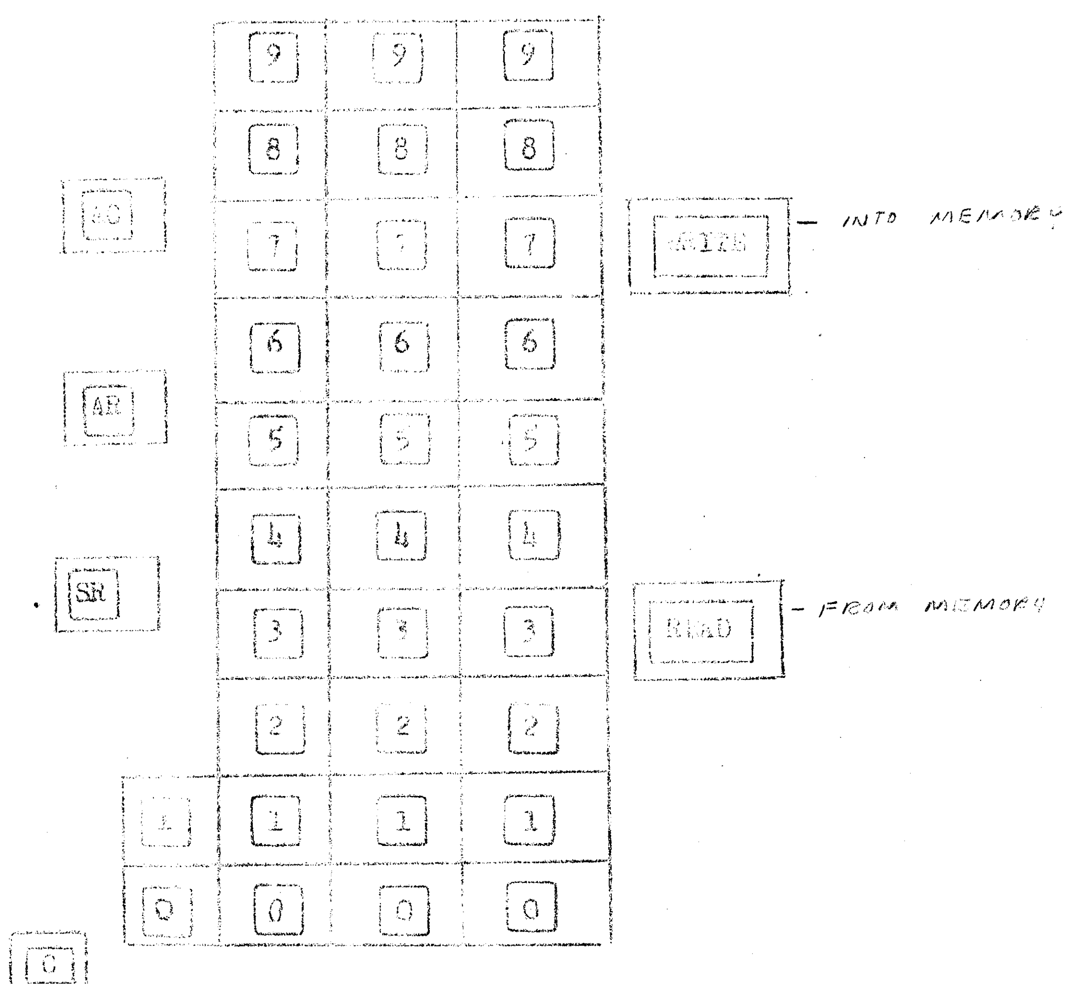
8. LOAD MEMORY - ENTER ORDER Switch - The LOAD MEMORY-ENTER ORDER switch is only active during paper tape input. In the Load Memory Mode up to 2000 words may be directly entered in memory from the control console without intervention of the manual operation features

of the machine. The Flexowriter prints the data as it is entered. All three styles of printing may be used. Special punched codes on tape indicate the end of a word and the end of tape. The Manual Register Selector specifies where the first word is to be placed in memory.

In the Enter Order Mode, all data on punched paper tape will be entered as instructions. Each instruction will be placed in the Control Register and carried out before the next one is entered. The next instruction always comes from the paper tape regardless of subsequences or transfer of control requested by the current order. If a Sequence Register change is requested by the current order this will be carried out but the next instruction still comes from the Control Register.

9. Manual Register Selector - The Manual Register Selector (see Figure 17) is a special keyboard by which machine registers or memory locations may be specified for manual print out or data insertion.

Figure 17



a. ADDRESSED REGISTERS Button - An adding machine type keyboard is available with which any address 0000 through 1999 may be specified.

b. UNADDRESSED REGISTERS Button - Since the machine contains three unaddressed registers whose contents could be of interest to the operator, three buttons are provided to handle the selection of these locations.

<u>Sequence Register</u> can be monitored and written into. It can also be cleared to zero.

<u>Auxiliary Register No. 1</u> can be monitored and written into.

<u>Accumulator</u> can be monitored only. No provision is made to insert data manually into the Accumulator.

c. CLEAR Button - A button permits clearing the MRS keyboard in the event of a human error. Normally, the keyboard is cleared automatically each time it is used.

d. READ Button - Associated with the MRS and the Flexowriter is a button serving the following functions.

<u>Initiate</u> the reading of the desired register to the binary indicator lights, the Flexowriter and the Tape Punch.

<u>Actuate</u> the Flexowriter control circuits to execute its printing, and perform a line feed and carriage return in anticipation of its further use.

<u>Clear</u> the MRS in anticipation of its further use.

e. WRITE Button - Associated with the MRS and the Flexowriter is a button serving the following functions.

Initiate the writing into the desired register or memory location of the data inserted via the Flexowriter.

Actuate the Flexowriter control circuits to perform a line feed and carriage return in anticipation of its further use.

Clear the MRS in anticipation of its further use.

III. USE OF THE FLEXOWRITER

A. Programmed Print Instructions

1. Function Description - These orders are incorporated in a prepared program and cause the print out of any memory location automatically. Some format control is available at the Console as itemized in Chart 1.

Note that an Instruction style print out occurs when a PRA order is given and the AUTO INSTRUCTION PRINT switch is ON.

2. Operation Procedure -

   a. For Numeric Print orders no operation action is necessary.

   b. For Alphabetic Print orders, set AUTO INSTRUCTION PRINT switch OFF.

   c. For Instruction Print orders, set AUTO INSTRUCTION PRINT switch ON. This switch controls whether a PRA order is interpreted alphabetically or instructionally, as defined above.

d. The PRINT CLEAR button should be used only in the exceptional case of some type of Flexowriter malfunctioning where clearing is necessary; for example, if the typing mechanism jams . Otherwise, it should not be disturbed.

e. Space Suppression is usable as shown. (see Chart 1)

f. Special Symbol control is usable as shown. (See Chart 1)

g. The EXTRA LINE INTERVAL COUNTER SET switch can be set as desired. Note value of lines already counted in indicator lights around switch.

h. The EXTRA LINE INTERVAL COUNTER CLEAR button may be used if necessary.

During Automatic Operation the operator should be constantly aware that he controls the following functions:

(1) "Alphabetic" or "Instructional" interpretation of the PEA command.

(2) The use of Space Suppression subject to the defined rules.

(3) The use of Special Symbols subject to the defined rules.

(4) The Extra Line Interval controls.

The equipment is designed so that controls shown on Chart 1 as "not effective" are literally that, and can be moved at will with no effect whatever on the print outs. Similarly, control switches such as are mentioned above will complete the previously defined action if they are changed during a print out, and will not take effect until the next complete printing. Therefore, all scheduled controls should be carefully anticipated to prevent oversights, bearing in mind that a minimum of one full print time exists (about 2½ seconds) before a change can be accomplished. The Extra Line Interval controls can be set at any time other than during the small fraction of a second (about 1/3 second) while a Carriage Return is taking place. During this time it should not be touched.

B. Manual Print out

1. Function Description - This techniques is used when the Central Processor is not operating, and the operator wishes to examine the contents of any location in the memory (addresses 0000 through 1999) or the Accumulator, Sequence Register or the Auxiliary Register No. 1. The format is controlled entirely by the Console controls (see Chart 1). Note that addresses are printed when manual readout is done except when Space Suppression is used.

2. Operation Procedure -

a. Set PRINT STYLE switch as desired.

b. AUTO INSTRUCTION PRINT switch has no effect during manual operation.

c. PRINT CLEAR button has no effect during manual operation.

d. Set SPACE SUPPRESS switch ON or OFF as desired.

e. Set SPECIAL SYMBOL switch ON or OFF as desired.

f. Extra Line Interval Counter controls have no effect during manual printouts.

g. Set desired Address or Register into <u>Manual Register Selector (MRS).</u>

h. Press READ key on MRS. Printing will take about three seconds. Characters will be in black and address will or will not be printed depending upon the SPACE SUPPRESS switch.

The same remarks made above regarding interlocks and switch changing are applicable here; however, there is one interesting point to be made. Since the very nature of the operation is manual, then certainly if the operator realizes that he has set a switch incorrectly, he merely corrects his oversight and repeats the procedure.

C. <u>Manual Write-In</u>

1. Function Description - Manual Write-In permits insertion by hand of new computer words via the Flexowriter keyboard. Address locations are handled identically as in the Manual Print outs with the single exception of the Accumulator, into which no writing can be done.

2. Operation Procedure -

a. Set PRINT STYLE switch as desired.

b. AUTO INSTRUCTION PRINT switch is not effective.

c. SPACE SUPPRESS switch is not effective.

d. SPECIAL SYMBOLS switch is not effective.

e. Extra Line Interval Counter controls are not effective.

f. Type in the desired data (or instruction) on the Flexowriter keyboard, <u>using no spaces</u> and no weight counts. Print either:

(1) 12 characters for numeric, or

(2) 8 characters for alphabetic, or

(3) 15 characters for instruction.

Unless exactly 12, 8, or 15 characters (depending upon style) are inserted, the equipment will not operate. If too few are used, the equipment will not cycle when the WRITE button (see below) is depressed. If too many are inserted, the Flexowriter will automatically Carriage Return and clear out the characters which were being stored within the Console for transmission to the Central Processor.

g. The PRINT CLEAR button should be used in the event that a wrong character is typed. The action will Carriage Return the Flexowriter and clear out the partial word, permitting the operator to insert the word correctly. The PRINT CLEAR button should also be used to permit unlocking interlocks if the operator realizes the wrong word style has been used. The new setting of the PRINT STYLE switch becomes effective when the old word is cleared out, permitting insertion of the word in the correct style. If reference words, such as notes or page titles, are printed, use the PRINT CLEAR button to reset the Flexowriter carriage.

h. Set the desired Address or Register into the MRS.

i. Press WRITE key on MRS. The word to be entered into the High-Speed Memory goes in as the result of depressing the WRITE key. However, many interlocks must be satisfied before actual transmission takes place. They are listed below.

(1) 12 numeric, 8 alphabetic or 15 instruction characters must be inserted.

(2) If a memory address (0000 through 1999) is not set in correctly (one digit per column), the equipment will not operate.

(3) If a special unaddressed register SR or AR (not AC) is used, no address digits on the keyboard should be depressed.

Printing takes place in red when information is entered into the Central Processor. Weight counts and addresses are always printed. Spaces between groups of characters are always suppressed.

D. Load Memory Operation from Paper Tape

1. Function Description - This permits the operator to insert up to 2000 words directly into the memory from the control console without intervention of the manual operation features of the machine. The paper tape (previously prepared) used for this purpose is fed through the Tape Reader to the memory via the Flexowriter. The Flexowriter prints the data as it is inserted. All three styles of printing can be used. Special punched codes on tape mark the end of each word and the end of the tape. The memory address at which the loading is to start is specified by the use of the Manual Register Selector keyboard.

2. Operation Procedure -

a. Prepare tape by punching paper in standard computer characters in any one of the three standard styles. After each word (12, 8, or 15 characters) insert an End of Word code on tape. After the entire message and the last End of Word code, insert a STOP code. Using the Tape Feed key, a convenient leader and trailer of tape can be prepared to facilitate handling. The Reader need not be started on the first character, but started rather on the feed holes of the leader. These codes are completely ignored by the equipment. The preparation of the tape should be done on an identical Flexowriter, remote from the Console Flexowriter.

b. Set the LOAD MEMORY - ENTER ORDER switch to LOAD MEMORY.

c. Set the PRINT STYLE switch to a position compatible with the style of punched data.

d. The AUTO INSTRUCTION PRINT switch is not effective.

e. The SPACE SUPPRESS switch is not effective.

f. The SPECIAL SYMBOLS switch is not effective.

g. The PRINT CLEAR button should be used only in the event of a malfunction or to clear interlocks which would detect an incorrectly prepared tape (too few or too many characters per word).

h. Extra Line Interval controls are not effective.

i. Load the paper tape onto the Reader unit taking care to insure that the tape is resting in the guides properly, and that the length of tape to be read will be free of snags or bunching as it feeds through the Reader.

j. Set initial location (memory location of the first word to be read) into address keys of the MRS.

Do not use READ or WRITE keys.

k. Depress key on upper left side of the Flexowriter labelled START READ. Action will begin after the finger is removed from the depressed key.

Equipment will cycle automatically, and no intervention should be allowed. In order to insert mixed groups of data such as groups of numeric, alphabetic, and instructions, handle each group separately, i.e., place a Stop Code at the end of each group. When the Stop Code is read, the equipment stops, the MRS keeping track of sequential addresses stops and is cleared out. Thus, the next group can be started by changing to Print Style accordingly, setting a new address in the MRS, and depressing the START READ button again. Addresses are automatically printed.

E. Enter Order Operation from Paper Tape

1. Function Description - This is a facility whereby the data read from the paper tape is entered as orders, one word at a time, into the Control Register. After each order word is entered, the order is executed, then the Reader is automatically started again and a new order is read in. This process continues to the end of the paper tape.

A more meaningful way of describing this type of operation is to say that the orders for the machine are supplied externally, from the Tape Reader. The next order always comes from the Tape Reader regardless of subsequences or transfers of control requested by the current order. Note, however, that if a transfer of control is performed, it will change the Sequence Register. This permits "presetting" the Sequence Register as desired prior to executing a "Start at Sequence Register". Theoretically, there is no limit to the number of orders which could be used in this mode of operation. The paper tape (previously prepared) used for this process is fed through the Tape Reader to the Control Register via the Flexowriter. The Flexowriter prints each word as it is inserted. Only one type of printing, viz., Instruction can be used. Special punched codes (same as those used for "load memory") mark the end of each order word and the end of tape.

2. Operation Procedure -

a. Prepare instructions on tape in any desired print style, using the End of Word code and the STOP code like in the Load Memory procedure.

b. Set LOAD MEMORY - ENTER ORDER switch to ENTER ORDER.

c. Set PRINT STYLE switch to chosen style.

d. Press START READ button.

Do not use MRS.

e. All other switches are not effective.

f. Print outs are in red.

g. The initials CR appear in the address position indicating that the order entered the Control Register and was there executed.

Interlocks guarantee that all control actions were performed properly. In the event that a tape was prepared incorrectly (too few or too many characters, an omitted End of Word code, etc.), the equipment stops. The printed data and actual tape should be examined. The PRINT CLEAR button will always clear out the last (incorrect) word (or order) before it is inserted (executed). Hence, the tape should be moved back and started again from that point. These comments are applicable to both uses of the Paper Tape equipment.

SECTION V

GENERALIZED STARTING PROCEDURES

Prior to the running of any program on the Computer, a set of instructions should be drawn up so that the time spent on the Console for any particular problem can be optimized. This may not be the duty of the Console operator, but it should be his responsibility to make sure that all programs, especially newly-written programs, have detailed operating procedures.

I. INITIAL SET-UP

The initial set-up of any run should include a description of the setting of all switches on the Console and Flexowriter. This insures proper automatic subsequence where applicable in the event of a TWC error, optional stops will be executed correctly, and any machine printouts via the Flexowriter will be carried out as expected. Precaution must be taken to insure that all tape addresses that are utilized by the program are free for the Console to use. A careful check should be made to determine if the proper tapes are mounted.

II. STARTING

There are four ways a program can be started. The correct procedure should be outlined, including automatic re-run procedures.

III. RUNNING

Each optional and the appropriate steps to be taken as each occurs should be defined. In an emergency or unexpected stop, the Console operator should have the contents of 1990, 1999, and S.R. printed on the Flexowriter.

The following is an example of loading a program to be assembled.

Load Assembly Program (AP) or Fixed Assembly Program (FAP) from Utility Tape and operate them.

A. Initial Set-Up

1. Mount Tapes

   Utility Tape on 07
   Program to be assembled on 01
   Tape for Assembly output on 03

2. Switches

   a. Console

      CHECK SUBSEQUENCE - OFF
      AUTO-MANUAL - MANUAL
      OPTIONAL STOP 1 - ON
      OPTIONAL STOP switch - COMPARE

   b. Flexowriter

      PRINT STYLE - Numeric
      SPACE SUPPRESS - ON
      SPECIAL SYMBOL - OFF
      LOAD MEMORY - ENTER ORDER to LOAD MEMORY

B. Load Bootstrap from Paper Tape

Insert paper tape into Flexowriter
Set ~~1997~~ 1977 on Manual Register Selector (MRS)
Depress START READ on Flexowriter

C. Starting

1. Depress 1982 START Button
2. Switch AUTO-MANUAL to AUTOMATIC
3. Depress C.R. button

   a. Central Processor prints out "Load - PSC"
   b. Central Processor stops

D. Loading Assembly

AUTO-MANUAL - MANUAL

Set 1963 on MRS

PRINT STYLE switch to Alpha

Type ASSEMBØØ

PRINT STYLE switch to Numeric

Type B00 000 945 000 in 1990

AUTO-MANUAL to AUTO

Start at CR

Flexowriter prints - AP READY (for AP) or IN HSM (for FAP)

E. Operate Assembly or Fixed AP

Error SS - ON

Load program search code into 1978

Start at (0000) to assemble --- If this is the first program on output tape, write B00 000 207 000 in 1990 and start at C.R.

Printout at completion of Assembly

ASSEMBLY COMPLETE --- If assembly operation is to be halted before completion, write B00 000 352 000 in 1990 --- Start at C.R. --- Printout: COMPLETE.

F. Summary of Loading Procedures for Assembly

Utility tape on 07

Load bootstrap starting at ~~1997~~ 1977 (8 words)

Press 1982 Printout Load PSC. Machine stops

Write ASSEMBØØ in 1963

Write B00 000 945 000 in 1990. Press C.R.

Printout AP READY

G. Summary of Operating Procedure for Assembly

Program to be assembled on 01

Tape for assembly output on 03

Error SS ON

Load program search code in 1978

Start at 0000

The preparation, starting and running of all programs will be similar in the over-all approach. Exact operating procedures for the other checkout routines can be found in the Utility Manual.

SECTION VI

CONSOLE STOP CONDITIONS

I. INTRODUCTION

A Console operator must necessarily become as familiar as possible with the various console conditions which can exist during the operation of a problem. The most common condition should, of course, be one where the problem is progressing in the automatic mode and no operator intervention is necessary. It is the duty of the operator to determine if a problem is in a "loop", in which case the problem must be stopped.

The most important conditions, however, are those in which the Computer has come to a halt. This indeed may be a programmed stop and should have been anticipated, thereby governing the actions of the operator. If, however, an unexpected halt arises, the operator is responsible for determining the reason for the halt, and taking necessary corrective steps to enable the program to restart. Surely it will not always be possible to start the program again. In this event, sufficient information should be obtained via the Flexowriter so that the programmer will be able to analyze the reason for this stop. If the halt is due to a check function, the operator is responsible for obtaining information which will enable the program to restart at the most convenient memory location.

II. OPTIONAL OR PROGRAMMED STOP

Cause: This type of stop, as implied by its name, is planned by the programmer to occur under predetermined conditions.

Console Picture: The only significant light during this type of stop is the OS (Optional Stop) light. It will be ON only when this type of stop occurs. The OP CODE lights will also display the code 15 6 (see Figure 18).

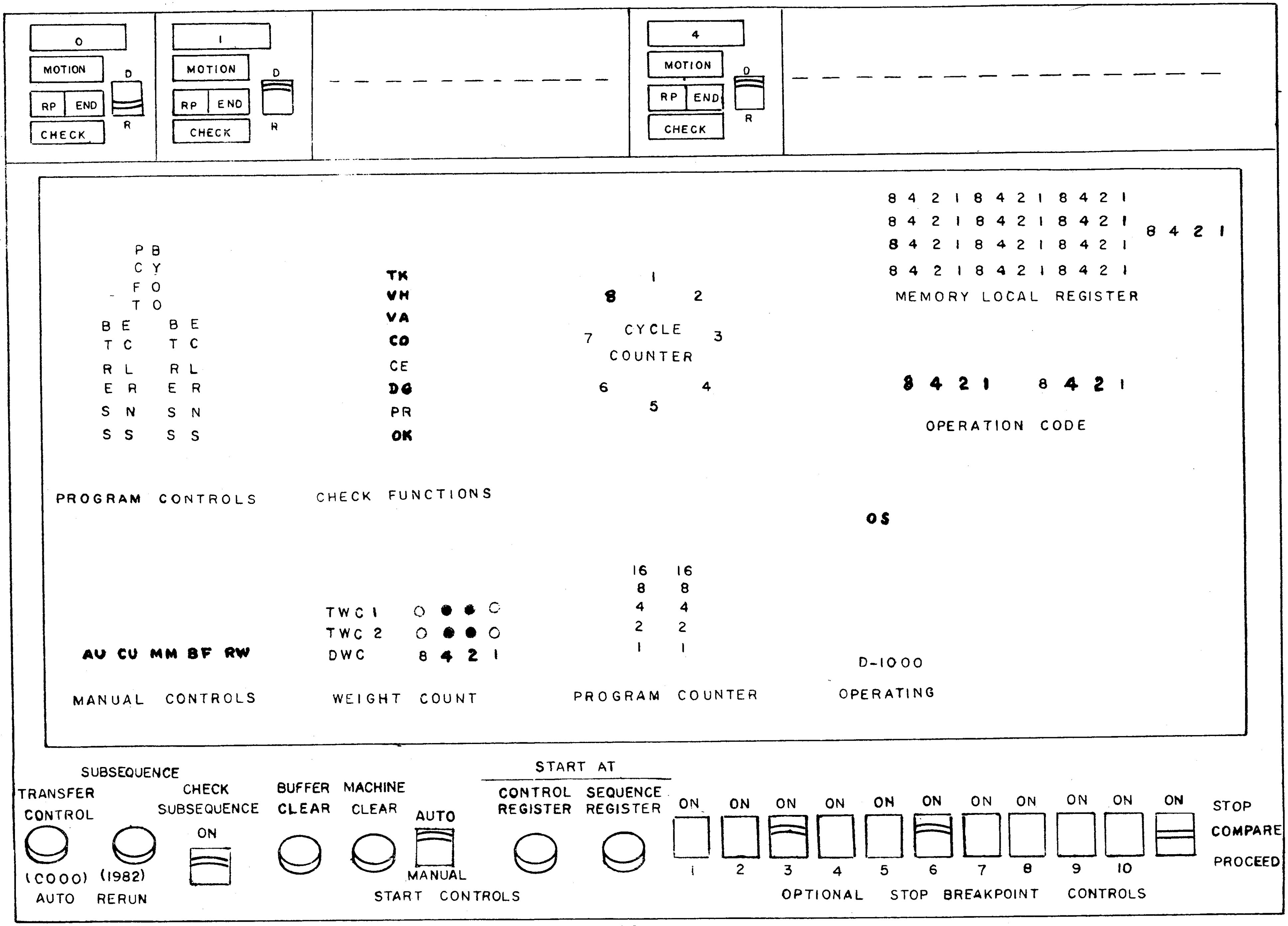

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
PB
CY
FO
TO
BE BE
TC TC
RL RL
ER ER
SN SN
SS SS
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 8 4 2 1
16 8 4 2 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER CONTROL
(C000) AUTO
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 18.

Corrective Action: The optional stop order is executed so that the next instruction to be performed will be taken either in sequence or will be a subsequence call. The particular optional stop order executed can be determined by the instruction itself and resides in either the Current Order Register (1999) or the Auxiliary Register (AR 1) depending upon whether there is a subsequence call or not. If there is a subsequence call, the instruction will stop on Cycle 8 and will be found in the Current Order Register. Likewise, if there is no subsequence call, the instruction will stop on Cycle 1 and will be found in the Auxiliary Register 1.

This instruction is used in a program for some particular reason or because of the occurrence of some special condition, etc. The operator is then called upon to take some special action. In order to minimize time at the Console, a clearly indexed sheet should be provided by the programmer which defines in a concise manner exactly what steps the operator is to carry out and the reason for the optional stop. Instructions should also be provided which define which start button should be depressed in order to resume the problem.

III. TWC CHECK STOP

Cause: There are two possible ways for the DATAmatic 1000 to stop due to a transfer weight count error. The first and most common is a transfer weight count error when entering data from the input buffer to the High-Speed Memory. The second condition occurs when an internal transfer results in a word having an incorrect weight count.

Console Picture: (see Figure 19)

D-1000 OPERATING light will be OFF.

TWC-1 or TWC-2 will have a red light in either of the two end positions.

Operation Code - code of instruction which caused check stop. If G3, instruction was a transfer from buffer; otherwise code not significant.

Cycle Counter - Cycle 3 when transferring data from buffer. Not significant otherwise.

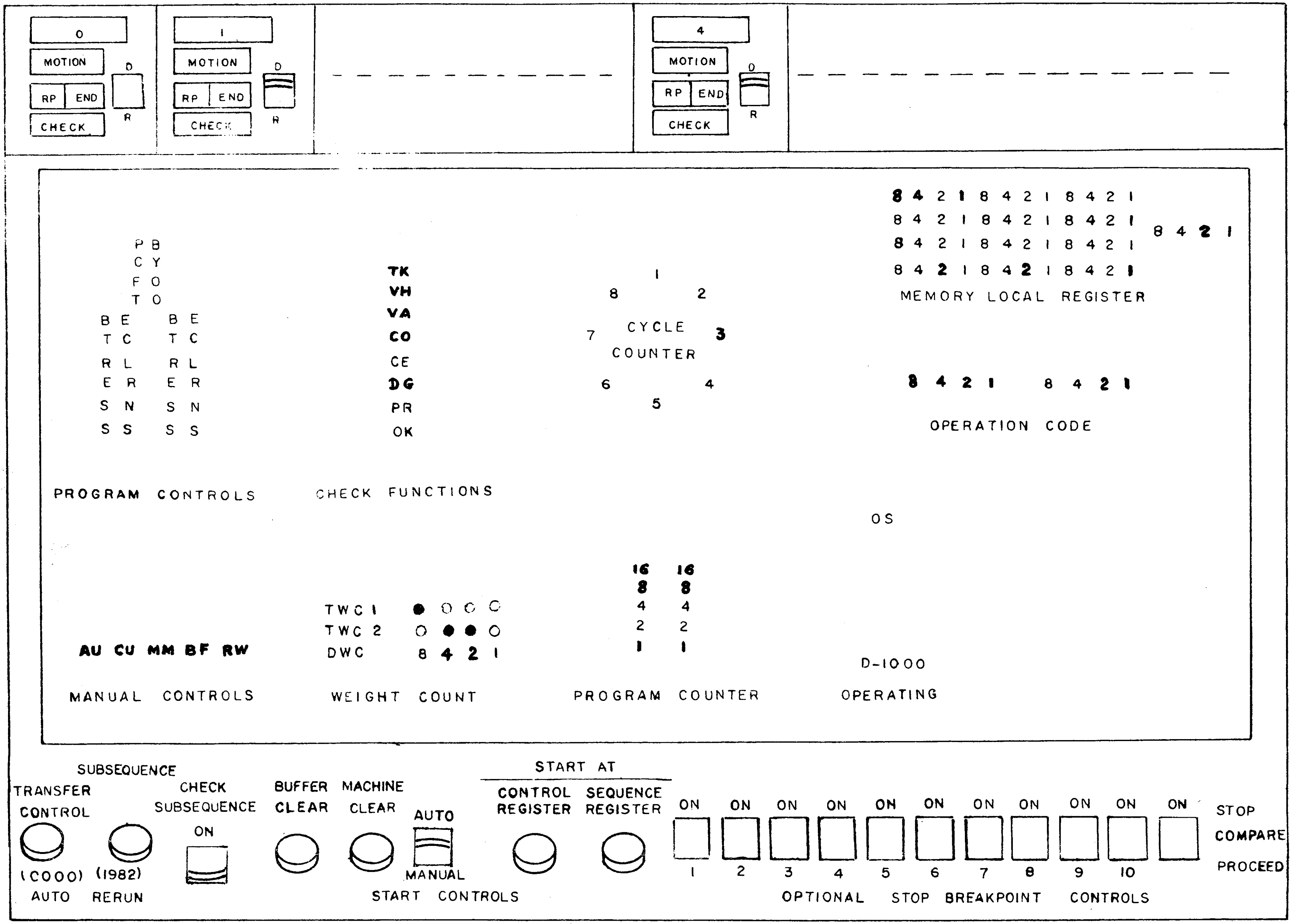

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 CYCLE COUNTER 3
6 4
5
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE TRANSFER CONTROL
(0000) AUTO
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
STOP
COMPARE
PROCEED
OPTIONAL STOP BREAKPOINT CONTROLS

Figure 19.

Corrective Action: The most common weight count errors occur when entering data from a buffer. If the ERROR SUBSEQUENCE switch is ON, an automatic subsequence call is made to 1987 where the start of a corrective routine is located. This eliminates any corrective procedure for the Console operator.

In the event a weight count error is detected in transferring data within High-Speed Memory, the Computer will always come to a halt. Experience has proved that this is an extremely rare condition and one which is not to be expected. There is no clear-cut way to re-start in the event that this happens. A possible corrective action could be the following.

Determine from SS light if stop occurred in normal or subsequence mode.

A. Normal Mode

1. Place AUTOMATIC-MANUAL switch in MANUAL if Cycle 2→8
2. Print contents of Sequence Register on Flexowriter (address of next normal instruction).
3. Set up previous address on Flexowriter
4. Write into Sequence Register
5. Re turn switch to AUTOMATIC
6. Start at Sequence Register

The above has the effect of repeating the instruction

7. If stop was on Cycle 1, simply return AUTOMATIC-MANUAL switch to AUTOMATIC and start at Sequence Register.

8. If the above does not work, it might be well to inspect the contents of the address that has the word with the incorrect weight count. This can easily be determined by printing the Control Register at time of error and noting which address appears in the A subaddress portion of the instruction. Inspect the contents of this address for incorrect weight count. If in error, the operator can follow one of two courses.

a. Re-insert the same data in the appropriate address and repeat appropriate steps to re-start.

b. If necessary, look up hard copy and re-insert correct data.

B. Subsequence Mode - SS Light ON

Cycle 2 → 7

1. Place AUTOMATIC-MANUAL switch in MANUAL.

2. Print contents of 1999 (Current Order Register) on Flexowriter. This is previous instruction of which C address is the location of instruction currently being performed.

Note: Addresses in 1999 are in form C A B

3. Print out on Flexowriter contents of C address.

4. Print on Flexowriter and write into Control Register (1990) instruction as it appears in step 3.

5. Return AUTOMATIC-MANUAL switch to AUTOMATIC.

6. Start at Control Register.

7. If stop was on Cycle 8, contents of 1999 or 1990 will contain instruction just processed. Therefore, the contents of Auxiliary Register should be typed on Flexowriter and steps 3, 4 and 5 at normal mode repeated, substituting Auxiliary Register for Current Order Register (1999) in step 3.

If the above does not correct the situation, step 8 can be repeated under Normal Mode.

Again, it should be stated that the possibility of a word with an incorrect weight count entering the system and not being detected is very remote. There is even less chance of a word with a correct weight count being transferred and the newly-generated weight count not being equal to the original.

If the occasion arises where a word with an incorrect weight count is detected, when transferring in data from a buffer and it becomes necessary to determine which word it is, the following procedure should be used.

1. Position tape
2. ERROR SUBSEQUENCE switch OFF
3. AUTOMATIC-MANUAL switch to MANUAL
4. Read block with incorrect word into a buffer
   a. Type into CR - 836 206 000 000 -- RFA 06 -- --
   b. Depress START AT CONTROL REGISTER button
5. Transfer in first 31 words from buffer. If Computer does not halt due to a TWC, transfer in next 31 words.
   a. Type in 8G3 101*01G 000 -- TIA 0101 31 --
   b. If no TWC 8G3 131*01G 000 -- TIA 0131 31 --
6. When TWC occurs type out the Control Register
   a. Assume type out is: -- 8G3 116 01G 000 -- The A address will contain the location of where the incorrect word is to go.

   (N.B. The incorrect word will not be in that location nor in the buffer.)

   b. In the above example it is apparent that the 16th word is the incorrect word on the block.

IV. <u>DWC CHECK</u>

<u>Cause:</u> If the Decoded Weight Count check function stops the Computer, it is due to the order not being executed according to the original instruction.

---

* A address should be any available area in memory

The DWC check insures that each address is properly selected, the Op Code properly interpreted and valid, and also determines if the next instruction to be selected is correct. By far the most common occurrence of the DWC check stop is due to a programming error in generating an address.

The Assembly program will catch any invalid address or Op Codes prior to a run. However, addresses which are modified by the program or instructions entered via the Flexowriter may possibly be invalid (memory location non-existent). This will result in a DWC check.

Console Picture: (see Figure 20)

D-1000 OPERATING - OFF
DWC - either the 8 or 1 or both ON
Op Code - only significant if invalid Op Code appears
CO light under check functions will be OFF if illegal Op Code

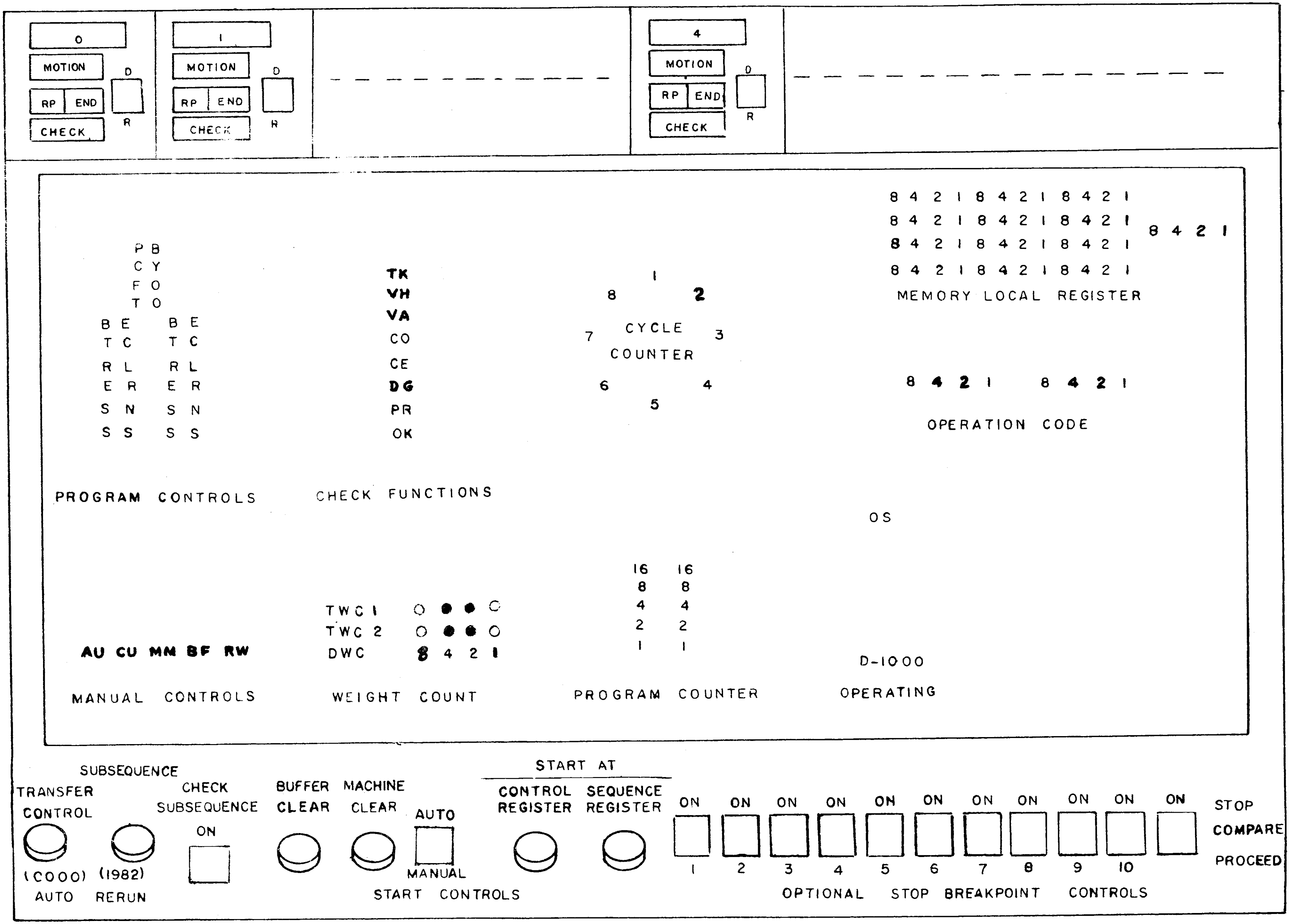
Remaining lights are insignificant.

Corrective Action: Before any corrective action can be taken, the underlying cause for the error must be understood. For this type of check stop, the following steps should be taken.

1. AUTOMATIC-MANUAL switch to MANUAL
2. Print out contents of Sequence Register
3. Print out contents of Control Register (1990)
4. Print out contents of Current Order Register (1999)

An invalid Op Code or address will show up in register 1990. An illegal Op Code will also turn the CO light OFF under check functions. If a new program and the correction is apparent, the operator should:

1. Set up the instruction via the Flexowriter and enter it into 1990.
2. Return AUTOMATIC-MANUAL switch to MANUAL.
3. Press START AT CONTROL REGISTER button.



0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
PB
CY
FO
TO
BE BE
TC TC
RL RL
ER ER
SN SN
SS SS
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 CYCLE 3
COUNTER
6 4
5
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(C000) AUTO
(1982) RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
STOP
COMPARE
PROCEED
OPTIONAL STOP BREAKPOINT CONTROLS

Figure 20.

If the original printout on the Flexowriter did not show any inaccuracy, repeat the instruction as outlined under TWC Corrective Action.

V. INVALID DATA CHECK

Cause: This type of stop occurs when attempting to perform arithmetic-type operations on unsigned quantities. This will invariably be on a newly-written program.

Console Picture: (see Figure 21)

D-1000 OPERATING - OFF

Operation Code - Arithmetic type order

[Add (14 14), Sub (15 15), Mul (14, 13), Div (13 13)]

Cycle Counter - 7

All other lights are insignificant

Corrective Action: In order to take proper corrective action when invalid data caused a machine stop, one must fully realize that immediate correction may be impossible without the source document. The following procedures could be used.

1. Put AUTOMATIC-MANUAL switch to MANUAL
2. Print out via Flexowriter the contents of:

   a. Sequence Register
   b. Control Register (1990)
   c. Addresses as dictated by 1990
   d. Current Order Register (1999)
   e. Note whether in Normal or Subsequence Mode

Conditions at the time may dictate whether or not to re-start. To restart at the following instruction, return switch to AUTOMATIC and depress SEQUENCE REGISTER button.

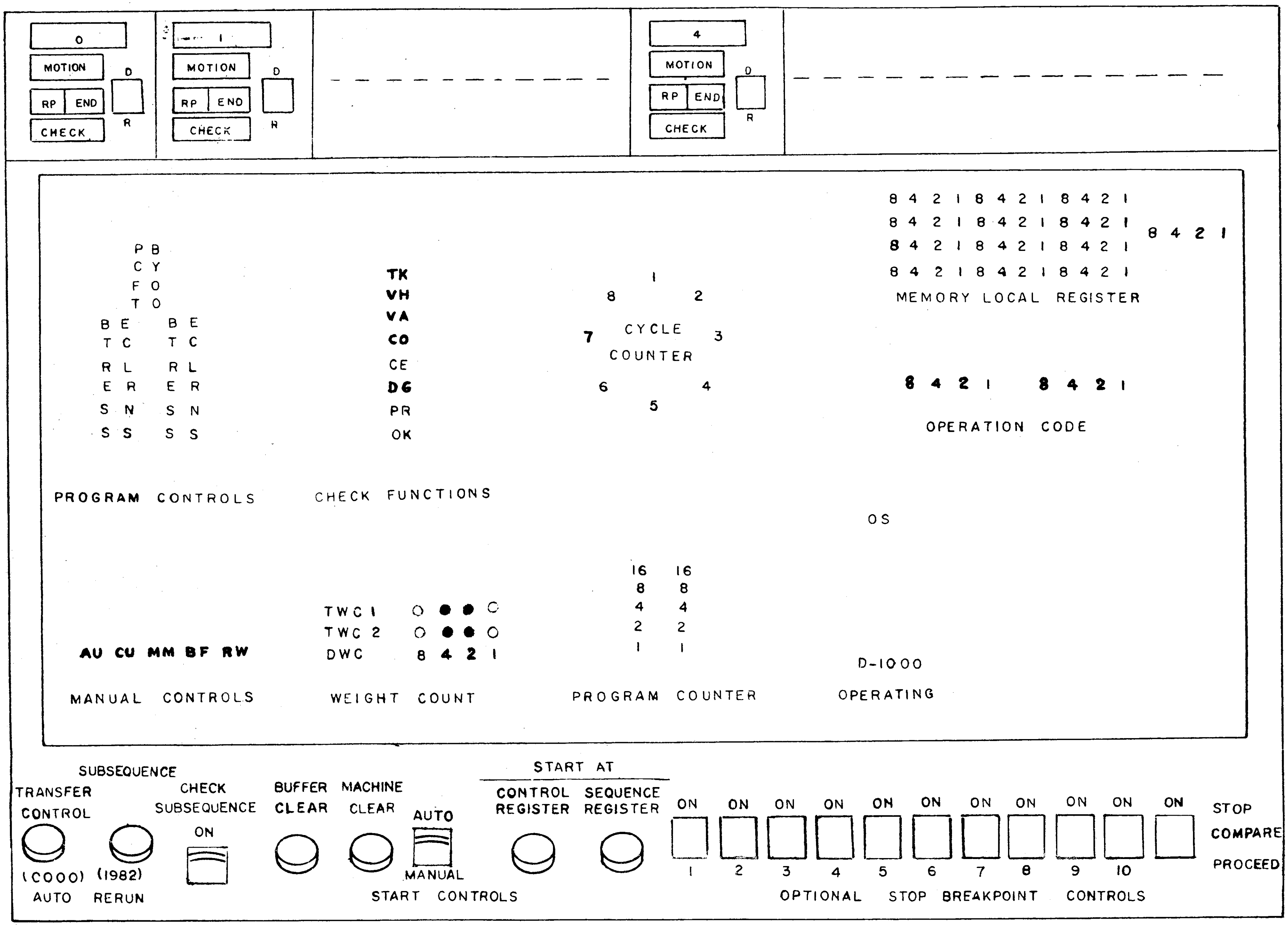

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 8 4 2 1
16 8 4 2 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER CONTROL
(C000) AUTO
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 21.

In general, it would be advisable to go to the next program until the error could be corrected.

VI. CHECK FUNCTION STOP

Cause: Computer stops of this type are brought about by the various checks within the Central Processor not being satisfied. The various check functions are described in the Section entitled Detailed Description of the Console.

Console Picture: (see Figure 22)

D-1000 OPERATING - OFF

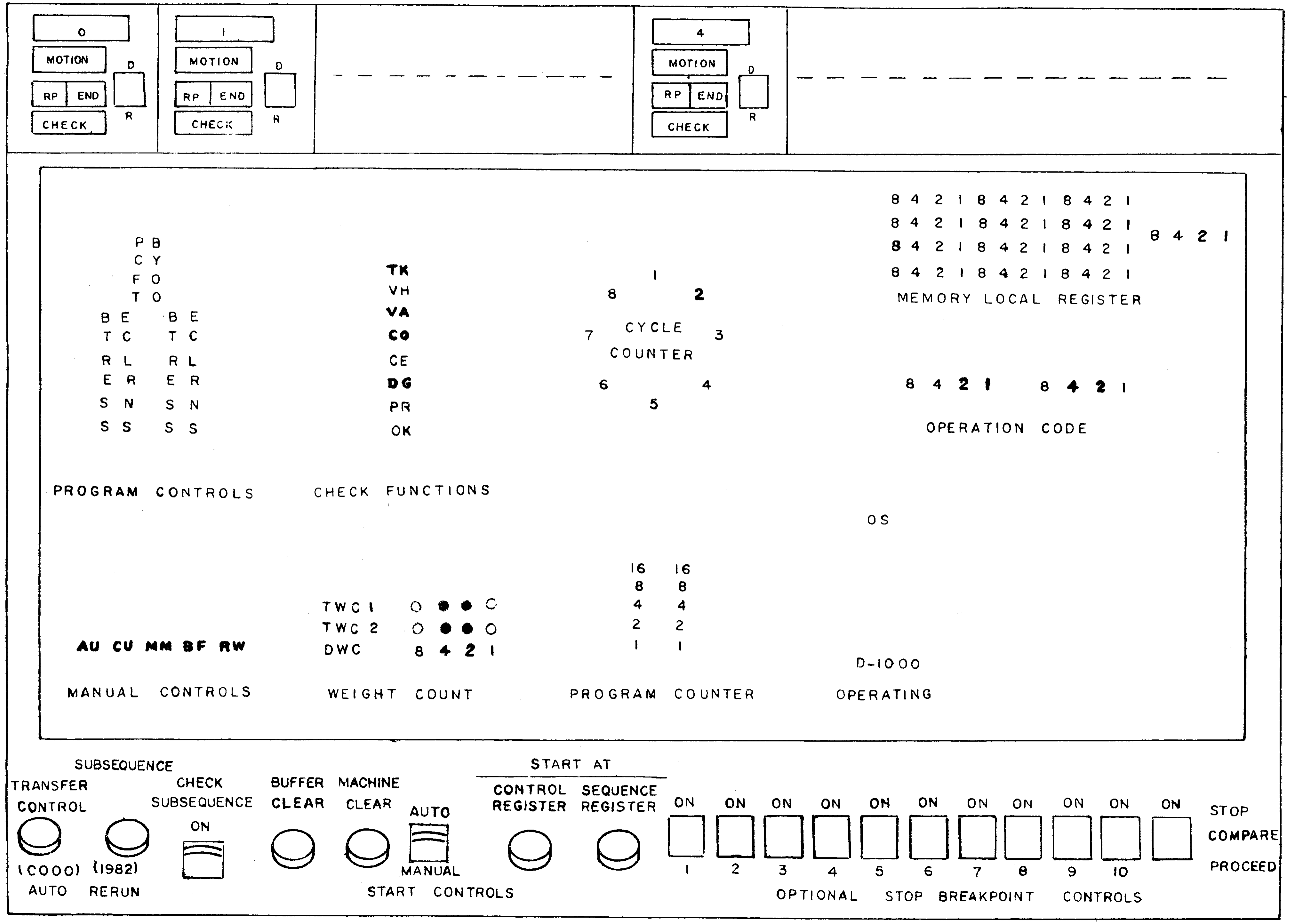
Check Functions: The light which represents the corresponding check function will not be in the normal state. This in general means that one of the check function lights will be OFF. In Figure 22 the VH light is OFF which should always be ON.

Corrective Action: Unless the check is an illegal Op Code (CO light OFF) it is usually wise to call a Field Service Engineer to correct the trouble.

VII. SUMMARY

A resume of the preceeding section will indicate that on any type of a halt, the D-1000 OPERATING light will be OFF. Therefore, the operator of the Console should constantly notice the status of this light.

While checking the D-1000 OPERATING light, the OS light can be checked simultaneously. If both lights are OFF, then a check of the TWC lights and OP CODE lights should be made. Also, if the CYCLE COUNTER reads 3, the stop is usually due to a transfer of one type or another. If the TWC lights are O K and the CYCLE COUNTER reads 7, then the error is probably one in which unsigned numbers were used in an arithmetic-type operation.



0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
PB
CY
FO
TO
BE BE
TC TC
RL RL
ER ER
SN SN
SS SS
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 3
CYCLE
COUNTER
6 4
5
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(C000) AUTO
(1982) RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 22.

As an operator becomes familiar with the Console, a quick scan of all lights should give a complete picture of the status of the Computer. Once an error has caused a halt, the Control Register (1990) and the Current Order Register (1999) are generally interrogated. After which the conditions that caused the stop dictate the corrective procedures to be taken by the operator.

SECTION VII

OPERATOR'S FUNCTION AND RESPONSIBILITY

I. INTRODUCTION

The Console operator is chiefly responsible for the efficient operation of any Computer installation. He should well realize his area of responsibility and the framework within which he operates. Efficient operation of any installation begins with efficient use of the Console, thereby making it imperative that the Console operator know exactly what procedures are to be carried out during normal operating conditions and especially during abnormal conditions. In short, the Console operator must be an expert so that he may be in a position to give on-the-spot advice to programmers who are checking out their programs on the Console, and also take immediate corrective action under any conditions that may arise. This includes knowing when the condition warrants calling in an engineer.

II. EFFICIENT OPERATION

A. Scheduling

The Console operator in general is not responsible for the scheduling or re-scheduling of problems. However, in many cases the operator assists in making out the day's schedule. Adjustments in a daily schedule have to be made in the event of unexpected Computer downtime. A small blackboard may be kept posted to inform interested persons as to how close the daily schedule is being adhered to.

B. Coordination

Becoming familiar with the programs that are run most frequently will greatly improve running efficiency in the event that unexpected conditions arise. Being able to make on-the-spot decisions in unexpected situations is a must for Console operators. The correct judgement will often save valuable computer time and in some instances, if the programmer is not present, can only be made if the operator is familiar with the most common computer runs.

[illegible]

[illegible] out [illegible]. The [illegible] as to [illegible] Computer [illegible] of time [illegible] by pro- [illegible] waiting to use the machine for new programs.

[illegible]

Probably one of the most important and easiest tasks an operator has is one of logging. Accurate logging of problem times is a must in Computer installations, and the responsibility falls completely on the operator.

Each installation will have its own method of logging. However, in general, all logging methods are basically the same. Separate logs will be maintained for the Central Processor and for each peripheral unit. Sample log sheets are shown in Figures 23 and 24. The basic rules which will generally apply to all logs are the following:

1. Log Housekeeping - An often abused rule, but its importance should be obvious. Make sure today's date appears on the sheet. Exceptions may be just after midnight when the previous day's date may be more convenient to use. Make sure enough carbon copies are made. (Installation decision.)

2. Time - Time at which run is started. In general 24-hour time is preferred in increments of 1/10 of an hour (six minutes). This eases the burden on the accounting department.

3. Problem or Job Number - This will identify to which problem the time is charged.

4. Running Time and Down Time - This section is one of the most important and generally the most often ignored column of the entire log sheet. All time on a given piece of equipment should be accounted for, be it good time or down time. Down time begins when

DATAmatic 1000

CENTRAL PROCESSOR LOG

DATE ____________________

Time Started	Operator	Affiliation	Problem or Job Number	Mag. File Units	Time Ended	Down Time	Remarks

Figure 23

DATAmatic 1000

PERIPHERAL LOG

EQUIPMENT: Input Converter ☐
Standard-Speed Output Converter with 407 Printer ☐
Standard-Speed Output Converter with 519 Punch ☐
High-Speed Output Converter with High-Speed Printer ☐

DATE ____________________

Time Started	Operator	Affiliation	Problem or Job Number	Description of Input or Output Data	Plugboards Used	Time Ended	Down Time	Remarks

Figure 24

a Service Request slip is filled out and an engineer is called in, and ends when the engineer turns the machine back to the operator. If the engineer can demonstrate that the <u>program</u> and not the <u>machine</u> was at fault, there is <u>no</u> down time.

5. Remarks - Include when applicable such comments as explanation of run, progress made, etc.

In general, during a given shift the Central Processor is booked solidly, and therefore the problem "off time" should be the same as the next problem "on time". There should be no unaccounted time between problems.

E. <u>Programmer Aids</u>

1. Assembly Routine and Utility System Routines - A good Console operator should be able to give assistance to programmers when first checking out new programs. An understanding of the usage of the Assembly Routine and the Utility System Routines available can be invaluable aids to programmers initially checking out new programs.

Typical coding errors should be checked for in new programs, especially with inexperienced programmers. Testing techniques should be fully understood by the programmer, but the Console operator is responsible for the instituting and completion of the programmer's desires.

2. Diagnostic Routines - Console operators should indeed know basic machine testing techniques in the event machine errors occur in the course of running a problem. There are numerous Diagnostic Test routines which can check out every portion of the Computer.

There are tests written to check the Arithmetic Unit, the Control Unit, the High-Speed Memory, the Input and Output Buffers, all pairs of orders, Comparison tests, Console tests, etc.

In some instances, errors occur only when a certain condition is met in the machine (i.e., a particular pair of orders occurring together or particular patterns of bits within memory). If a Console operator is able to recognize certain symptoms and properly describe them, proper use of Diagnostic Test Routines will be simplified.

F. Preventive Care

Preventive care can also be described as good operating technique. Breakdown cannot be prevented, but by proper handling of equipment and supervising the operating techniques of others, some down time will be averted. This obviously increases the efficiency of the entire Computer installation. Knowing when to call an engineer and when not to is just as important. Exercising this proper judgement shoud come with experience.

G. Responsibility

The Console operator should have authority over all peripheral units as well as the Console. It is of the utmost importance the Console operator be certain that all tape units needed for the successful operation of a problem at hand be at the Central Processor's disposal. All operators of peripheral equipment should be directly responsible to the Console operator, and any changes on any of the File Units should not be made without first obtaining authorization from the Console operator.

SECTION VIII

EXAMPLES

The following illustrative examples are designed to aid the student in determining why the Central Processor halts and what is to be done to re-start. The section following the illustrations lists the reason for the halt and the proper corrective action to be taken.

The student should study the illustrations and list the cause of the halt and the proper corrective action to take before turning to the answers.

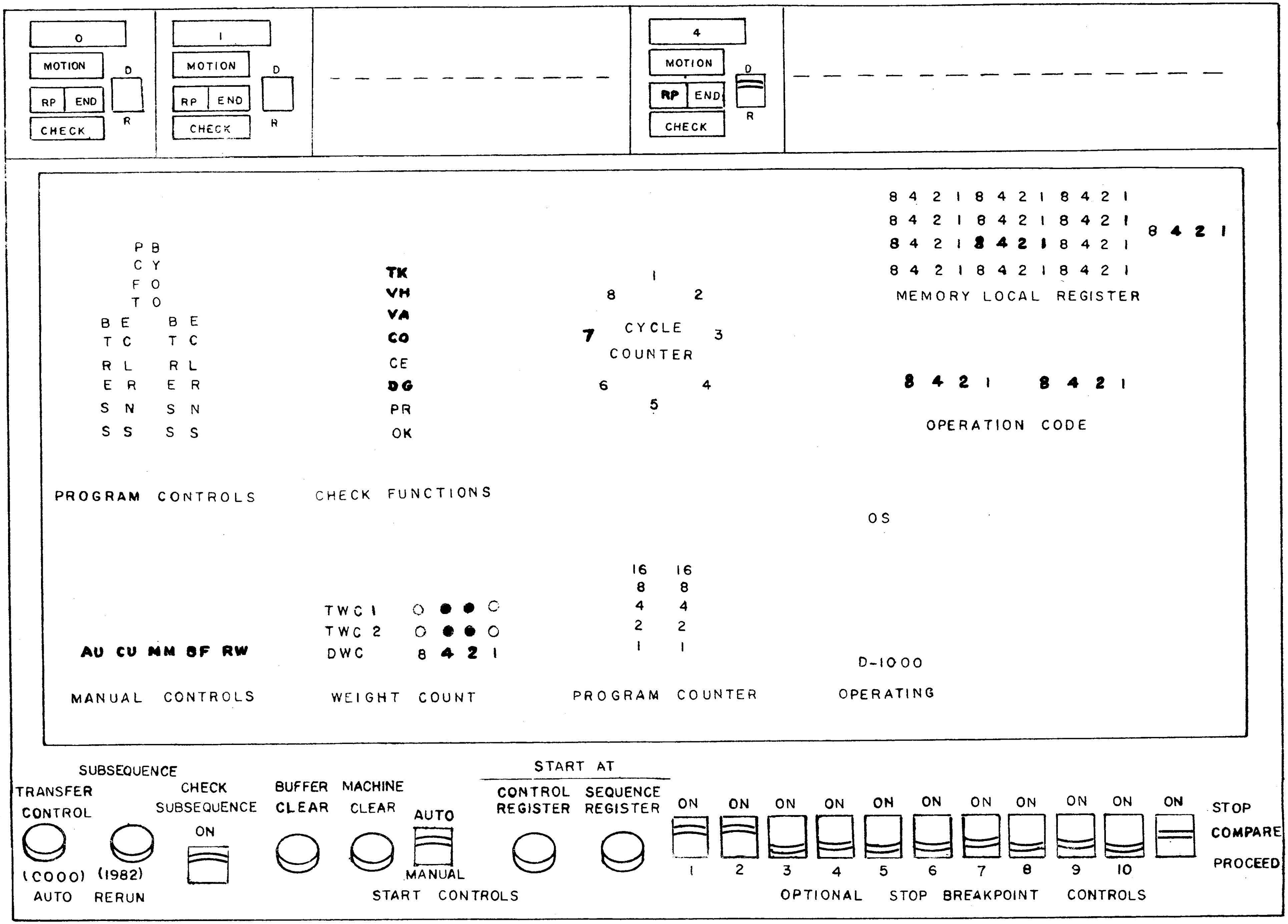

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 CYCLE 3
COUNTER
6 4
5
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(COOO)
AUTO
(1982)
RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
STOP
COMPARE
PROCEED
OPTIONAL STOP BREAKPOINT CONTROLS

Figure 25.

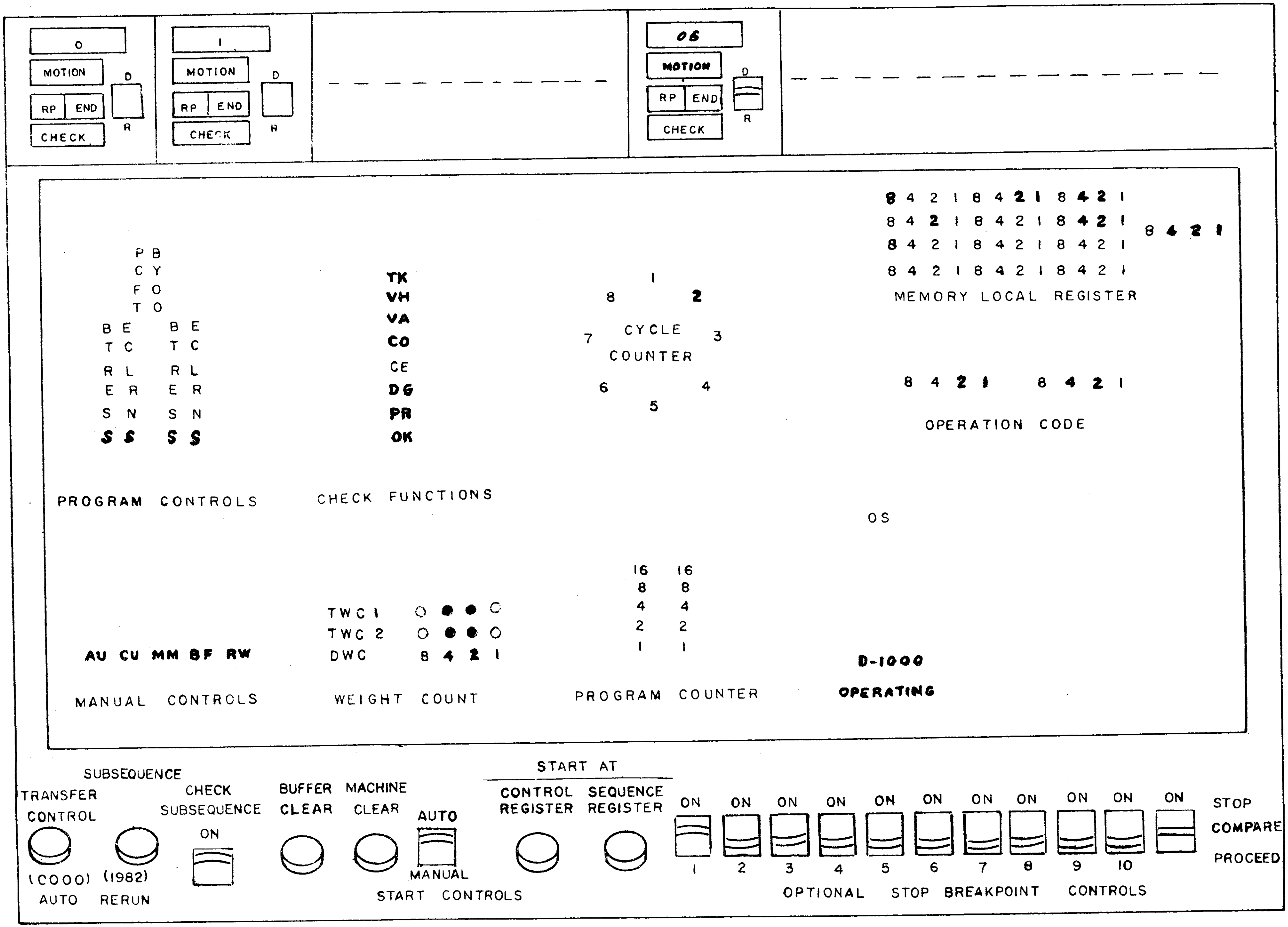

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
06
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 CYCLE 3
COUNTER
6 4
5
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(COOO)
AUTO
(1982)
RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 26.

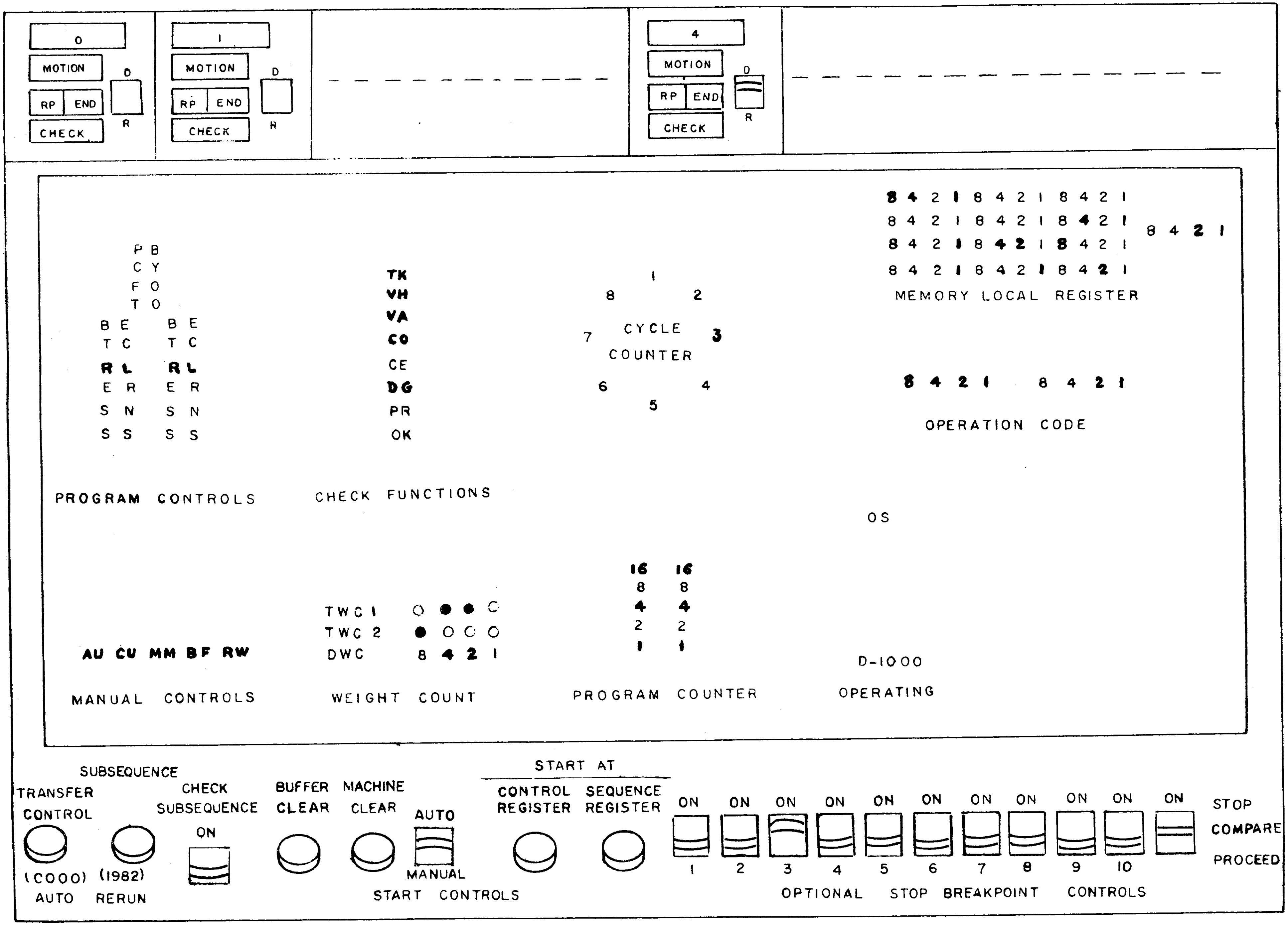

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
PB
CY
FO
TO
BE BE
TC TC
RL RL
ER ER
SN SN
SS SS
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
TRANSFER CONTROL
(COOO) AUTO
SUBSEQUENCE
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 27.

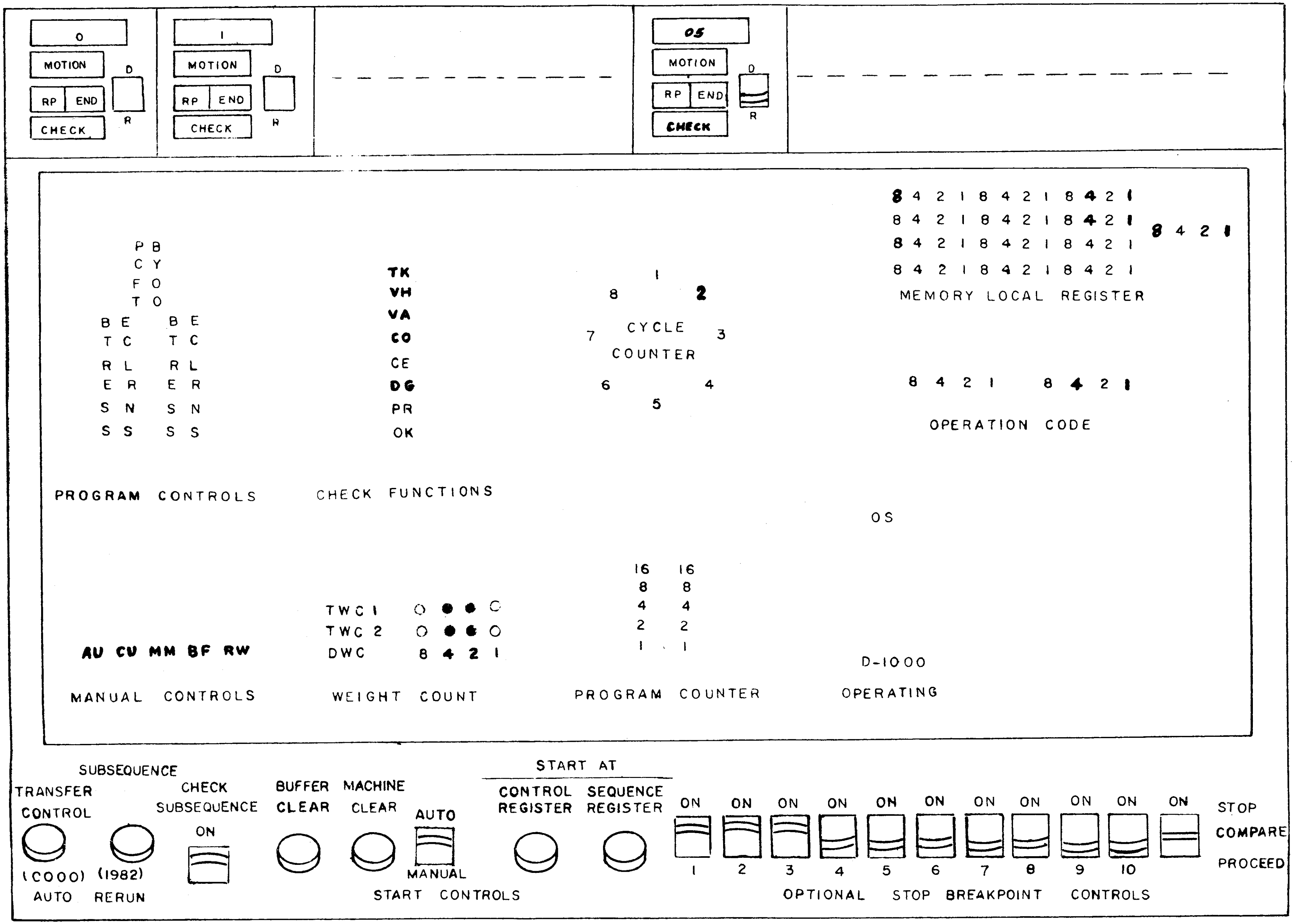

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
05
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
TK
VH
VA
CO
CE
DG
PR
OK
1 2 3 4 5 6 7 8
CYCLE
COUNTER
8 4 2 1 8 4 2 1
OPERATION CODE
PROGRAM CONTROLS
CHECK FUNCTIONS
OS
16 16
8 8
4 4
2 2
1 1
TWC 1
TWC 2
DWC 8 4 2 1
AU CU MM BF RW
D-1000
MANUAL CONTROLS
WEIGHT COUNT
PROGRAM COUNTER
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
STOP
COMPARE
PROCEED
(C000) (1982)
AUTO RERUN
START CONTROLS
OPTIONAL STOP BREAKPOINT CONTROLS

Figure 28.

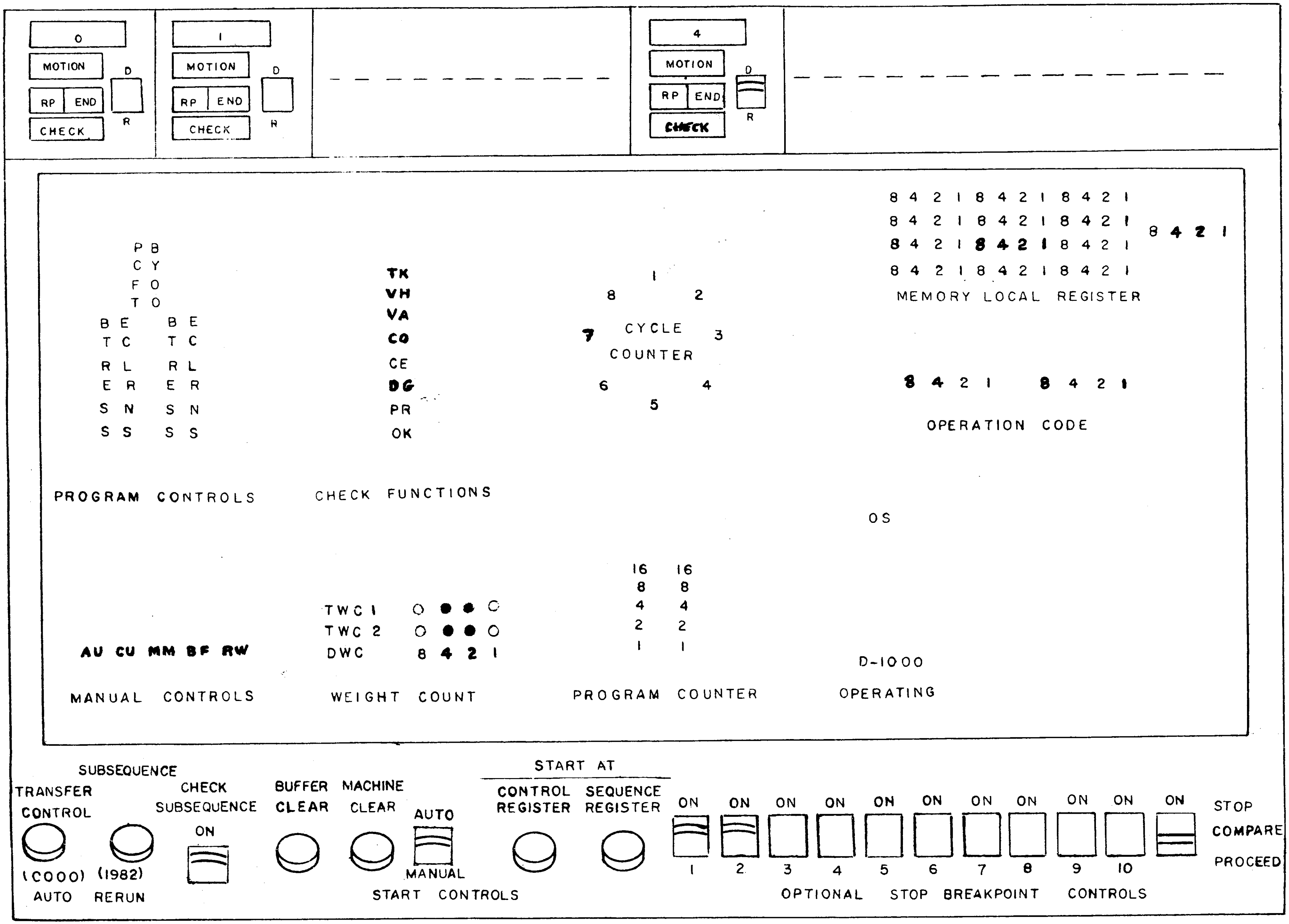

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(C000) AUTO
(1982) RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
START CONTROLS
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 29.

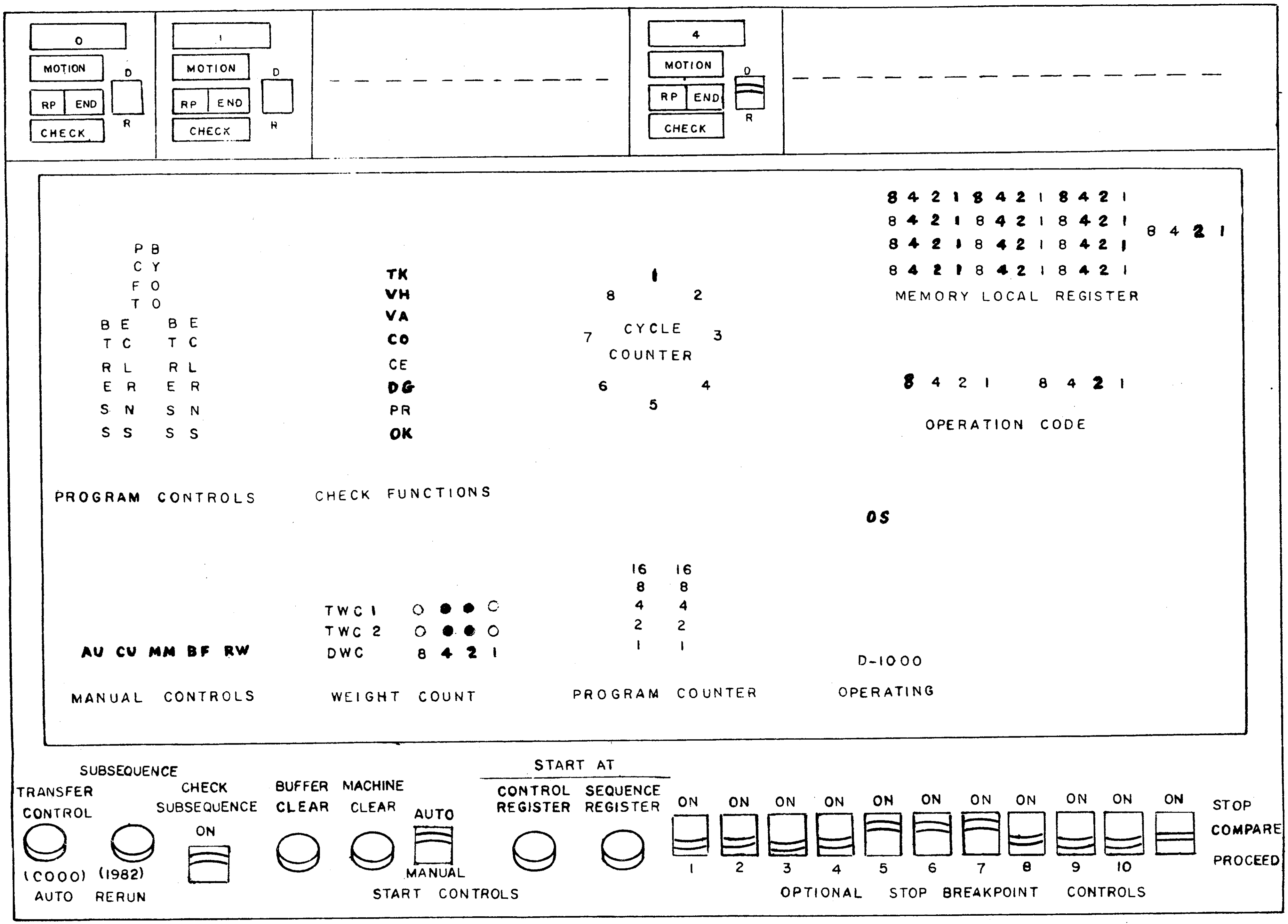

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
P B
C Y
F O
T O
B E
T C
R L
E R
S N
S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1 2 3 4 5 6 7 8
CYCLE COUNTER
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 8 4 2 1
16 8 4 2 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER CONTROL
(C000) AUTO
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START AT
CONTROL REGISTER
SEQUENCE REGISTER
START CONTROLS
ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 30.

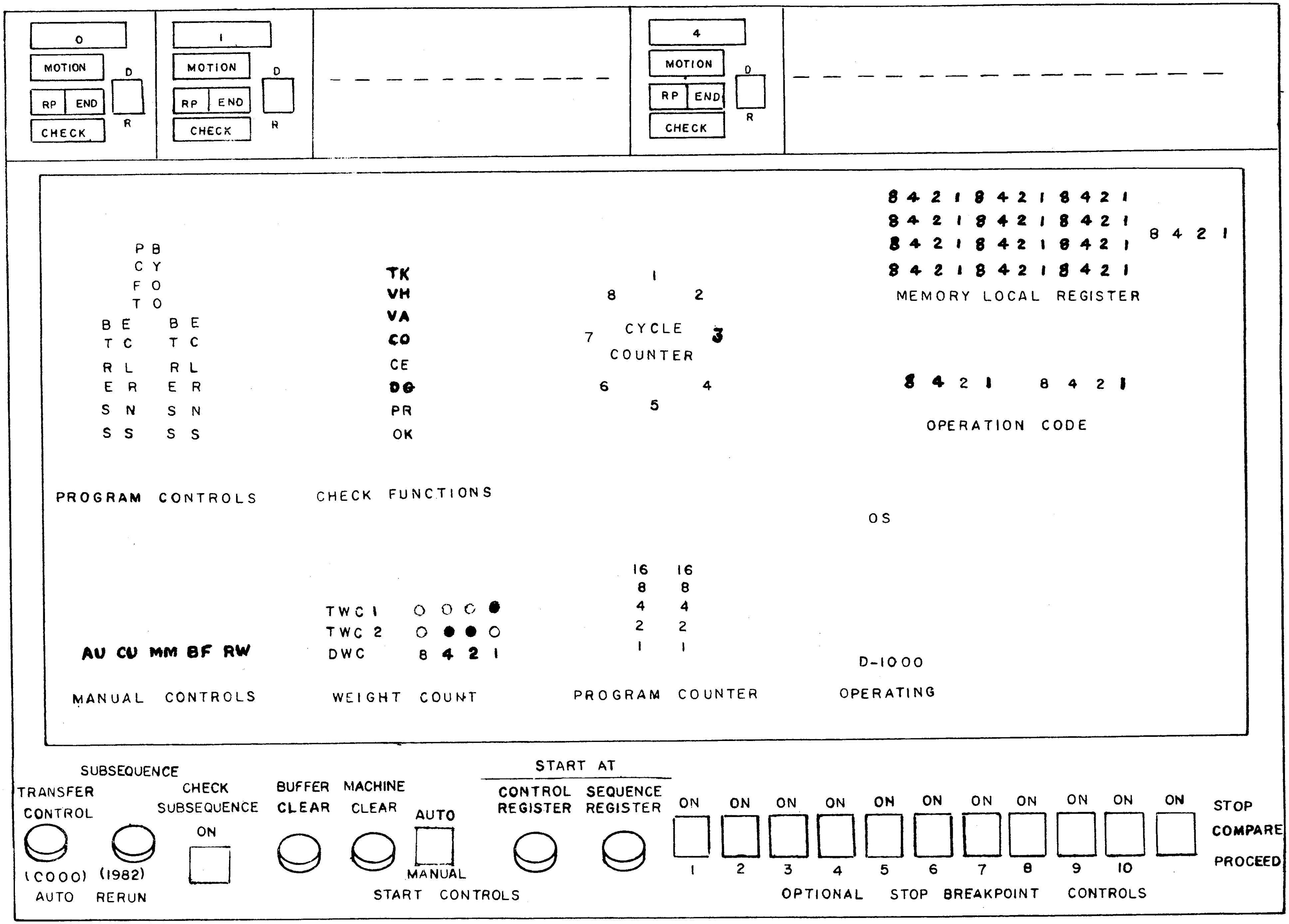

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
7 CYCLE COUNTER 3
6 4
5
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
MEMORY LOCAL REGISTER
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER CONTROL
(C000) AUTO
(1982) RERUN
CHECK SUBSEQUENCE
ON
BUFFER CLEAR
MACHINE CLEAR
AUTO
MANUAL
START CONTROLS
START AT
CONTROL REGISTER
SEQUENCE REGISTER
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 31.

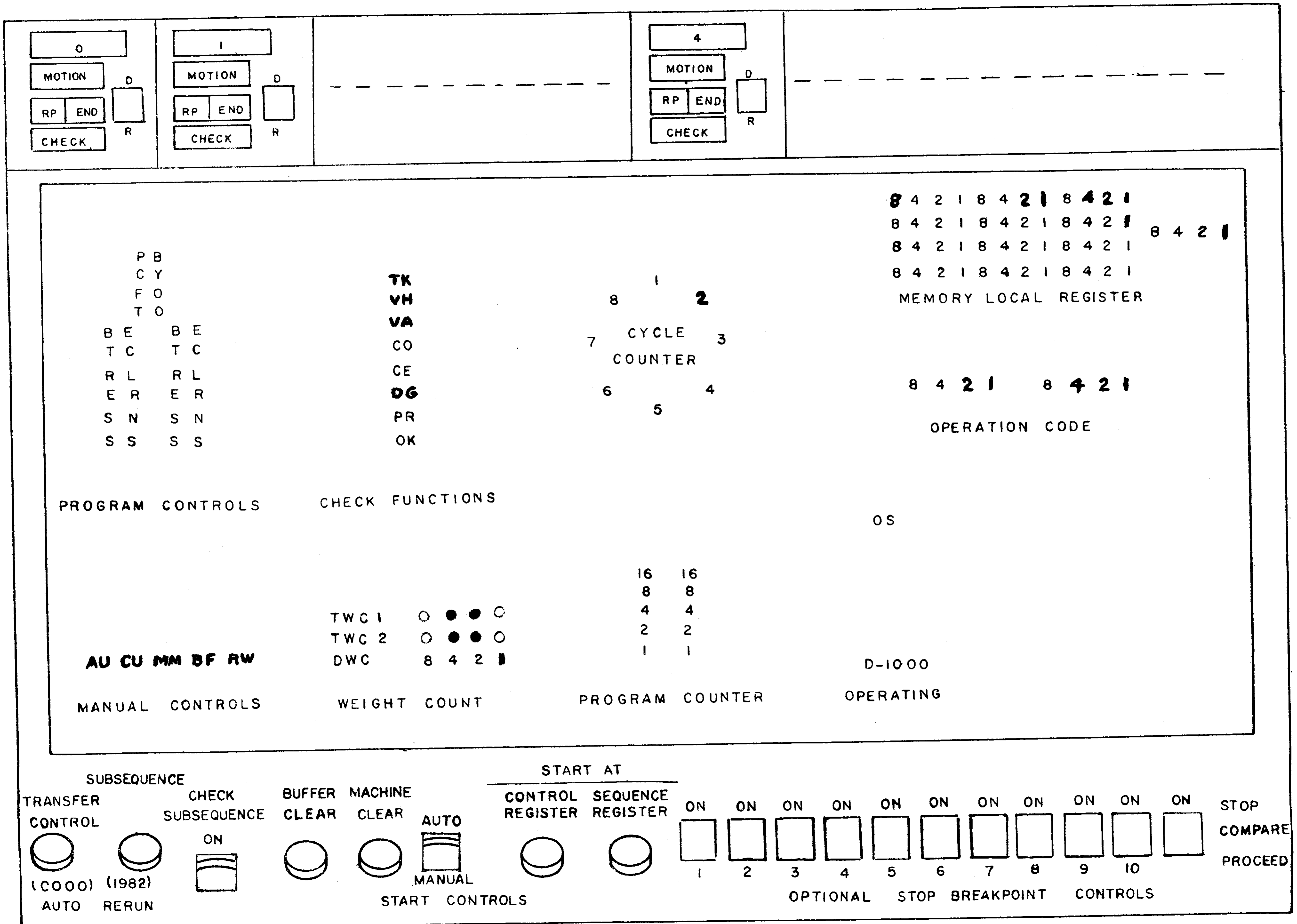

0
MOTION
RP END
CHECK
D
R
1
MOTION
RP END
CHECK
D
R
4
MOTION
RP END
CHECK
D
R
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1 8 4 2 1 8 4 2 1
8 4 2 1
MEMORY LOCAL REGISTER
P B
C Y
F O
T O
B E B E
T C T C
R L R L
E R E R
S N S N
S S S S
PROGRAM CONTROLS
TK
VH
VA
CO
CE
DG
PR
OK
CHECK FUNCTIONS
1
8 2
CYCLE
7 COUNTER 3
6 4
5
8 4 2 1 8 4 2 1
OPERATION CODE
OS
AU CU MM BF RW
MANUAL CONTROLS
TWC 1
TWC 2
DWC 8 4 2 1
WEIGHT COUNT
16 16
8 8
4 4
2 2
1 1
PROGRAM COUNTER
D-1000
OPERATING
SUBSEQUENCE
TRANSFER
CONTROL
(C000) AUTO
(1982) RERUN
CHECK
SUBSEQUENCE
ON
BUFFER
CLEAR
MACHINE
CLEAR
AUTO
MANUAL
START CONTROLS
START AT
CONTROL
REGISTER
SEQUENCE
REGISTER
ON ON ON ON ON ON ON ON ON ON ON
1 2 3 4 5 6 7 8 9 10
OPTIONAL STOP BREAKPOINT CONTROLS
STOP
COMPARE
PROCEED

Figure 32.

EXPLANATION OF ILLUSTRATIONS

Figure 25

The DATamatic 1000 has stopped on Cycle 7 of an ADD order. This is clearly indicated by the OP CODE lights and the CYCLE COUNTER.

Corrective Action

1. Depress MACHINE CLEAR button.
2. AUTOMATIC-MANUAL switch to MANUAL
3. Print out contents of 1990 on Flexowriter
   Assume printout: BFF 306 306 101
4. Print out contents of 0306 and 1101
   0306: E00 000 000 000
   1101: 800 005 000 000

Error is that contents of B address is an unsigned quantity; therefore, replace 1101 with proper sign.

5. Type in 1101 via Flexowriter
   1101: +00 005 000 000
6. Reconstruct order in 1990
   Type in 1990: BFF 306 101 306
7. Start at Control Register

Figure 26

In this example, the computer has apparently halted except that the D-1000 OPERATING light is ON. Closer inspection indicates that the OP CODE lights indicate a read instruction is taking place. The Memory Local Register shows that the instruction is RFA 06 --- ---. The MFU lights on the Console indicate that tape 06 is in motion and the cause of the apparent halt due either to a previous READ, WRITE, or REWIND order on the same drive not being completed.

Figure 27

This stop is easily recognized as a TWC check stop. The operation code and the cycle counter show that the computer stopped on Cycle 3 of a transfer-in order. The program counter indicates how many more words remain to be transferred in to complete this order. In this case, 10. The Memory Local Register contains the last valid word transferred in. The fact that the error subsequence switch if OFF prevented the invalid word from reaching the High-Speed Memory.

Corrective Action: The course of action is dependent highly upon what the programmer has in mind. The Control Register will contain, in the A address, the location where the next word is to be transferred, i.e., (1990) DG3 482 01F 000. Therefore, the invalid word would have gone to 1482 had it not been stopped. A re-start at the Control Register would place the invalid word in location 1482, but the Computer again would come to a halt. At this point the correct or any valid word could be typed in via the Flexowriter and re-starting at the Control Register again will permit the program to continue.

Note: Before any corrective action can be taken, the MACHINE CLEAR button must be depressed to reset the interlocks which stopped the machine.

Figure 28

The instruction that has stopped the Computer is a rewind instruction as seen by the OP CODE 05. A check of the MFU lights shows that the CHECK light is ON on the drive addressed as 05. Furthermore, the DEMAND-RELEASE switch is in the RELEASE position which indicates that the Console does not have control of the MFU.

Corrective Action:

Place DEMAND-RELEASE switch in DEMAND position

Depress MACHINE CLEAR and RESTART AT CONTROL REGISTER buttons.

Figure 29

The DATAmatic 1000 has stopped in the Automatic mode during the execution of an ISL instruction. The most probabl: cause for the check stop is due to an invalid C address. The contents of 1999 will yield the address of the current instruction, i.e., BOO XXX 000 000 or 800 XXX 000 000. To further qualify this, the three least-significant digits appear in the A subaddress position, while the memory designator bit is bit position 50 or D_b. Since the instruction is generally always plus, either an 8 or B will be in the leftmost position of the Flexowriter printout. Thus, if the print out were BOO 652 000 000 the register 1652 should be interrogated. The restriction placed on the ISL instruction is that the extracted digit plus the least-significant digit of the C address be less or equal to 19.

Corrective Action:

Print out 1999: BOO 652 000 000
Print out 1652: GD9 810 950 866
C address is invalid
Consult programmer on restart

Figure 30

The DATAmatic 1000 has stopped due to an optional stop instruction. A check of the Breakpoint switches indicates that any of four optional stop instructions could have halted the Computer. Since the stop occurred on Cycle 1, no subsequence call was made from the instruction. If there was a subsequence call, the stop would occur at Cycle 8.

Corrective Action:

1. All Optional Stops should be described by the programmer outlining the necessary steps needed to restart.

2. To determine which optional stop caused the halt

   a. Print out AR 1 of 1999 is stop is on Cycle 8
   Assume printout is 66 ~~882~~ 000 080 000
   866
   Printout indicates that the optional stop is due to Breakpoint switch 5 being set.

b. The following table should be useful in ascertaining which Breakpoint switch caused an Optional Stop.

Breakpoint Switch	B Address
1	800
2	400
3	200
4	100
5	080
6	040
7	020
8	010
9	008
10	004
Unconditional Stop	001

It should be further noted that at the conclusion of any Optional Stop instruction, the contents of the Control Register will be the next instruction to be performed.

Figure 31

In the condition shown, the halt is due to a weight count check function halting the Computer. The Internal Transfer instruction, as indicated in the OP CODE register, has come to a halt due to an incorrect weight count. The invalid word appears as a word of all G's and can be observed in the Memory Local Register.

Corrective Action

1. Depress MACHINE CLEAR button
2. AUTOMATIC-MANUAL switch to MANUAL
3. Print out Control Register

   Assume printout is 8E1 107 012 207

   The addresses are permuted to the form ACB during Cycle 3; therefore, the word in 0107 is invalid.

4. Check word in 0107 via Flexowriter printout

   a. If invalid, type valid word in via Flexowriter
   b. If valid, repeat instruction

<u>Figure 32</u>

The DATAmatic 1000 has halted due to a programming error as indicated by an illegal operation code in the OPERATION CODE register.

<u>Corrective Action:</u>

1. Depress MACHINE CLEAR button
2. AUTOMATIC-MANUAL switch to MANUAL
3. Print out via Flexowriter 1999

   Assume 1999: 800 352 000 000
4. Check location 0352 with printout via Flexowriter

   a. If invalid, check program sheet for correct instruction.
   b. If valid, repeat instruction and re-start.