MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Lincoln Laboratory
Lexington 73, Massachusetts

VOlunteer 2-3370

31 October 1958

To: Recipients of Whirlwind Programming Manual (2M-0277)

From: Frank E. Heart

Subject: Error Detection

An attempt has been made to maintain accuracy in this Programming Manual; however, with any document this size, it would be very surprising if errors were completely missing. It would therefore be sincerely appreciated if any recipient who detects a mistake or an omission would please communicate this fact, preferably in writing, to the undersigned.

Frank E. Heart

Frank E. Heart

FEH:bic

2M-0277

WHIRLWIND

PROGRAMMING

MANUAL

UNCLASSIFIED

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

Memorandum No. 2M-0277

To: Distribution List

From: Cynthia Muhle

Subject: Whirlwind Programming Manual

Non-Lincoln Recipients
PLEASE DO NOT RETURN
Permission has been granted by the Contracting Officer to destroy this document, when it is no longer required by the using agency, according to applicable security regulations.

ABSTRACT

This programming manual describes the current Whirlwind order code and explains the programming use of input-output equipment associated with the Whirlwind computer.

It is intended that this manual supersede prior Whirlwind programming manuals and, in particular, it is intended to supersede 6M-1623-2 and 6M-1624-2. It includes all revisions and modifications to the Whirlwind Instruction Code and the input-output system as of the date of publication.

This document has been prepared for internal use only. It has not been reviewed by Office of Security Review, Department of Defense, and therefore is not intended for public release. Further dissemination or reproduction in whole or in part of the material within this document shall not be made without the express written approval of Lincoln Laboratory (Publications Office).

Signed Cynthia J. Muhle
Author

Approved Frank E. Heart

Date 31 October 1958

CJM:bic Rec'd 31 October 1958

The research reported in this document was supported jointly by the Department of the Army, the Department of the Navy, and the Department of the Air Force under Air Force Contract No. AF 19(122)-458.

UNCLASSIFIED

ACKNOWLEDGMENTS

A number of persons actively supported the technical preparation of this document and their help is gratefully acknowledged. These included F. E. Heart, W. F. Holst, A. A. Mathiasen, H. D. Neumann, M. Curran, J. E. Drinan, H. C. Peterson and P. Stylos. Also, C. E. Brackett and a number of other Group 60 Whirlwind technicians were helpful in checking technical accuracy.

TABLE OF CONTENTS

		Page
SECTION I.....	DEFINITIONS AND INTRODUCTORY MATERIAL.....	4
SECTION II.....	WHIRLWIND I OPERATION CODE.....	12
SECTION III.....	MAJOR IN-OUT UNITS.....	25
3.1	Magnetic Drums.....	25
3.2	Magnetic Tape Units.....	32
3.2.9	Delayed Output via Magnetic Tape..... (Delayed Flexo Printing and Punching)	36
3.3	Photoelectric Tape Reader.....	39
3.4	Mechanical Tape Reader.....	43
3.5	Printer (Direct Flexo Printer).....	45
3.6	Anelex Printer.....	47
3.7	Punch (Direct Flexo Punch).....	56
3.8	Oscilloscopes.....	59
3.9	Cameras.....	64
3.10	Clock (Timing Register).....	66
3.11	Manual Intervention Registers and Indicator Lights..	68
SECTION IV.....	SPECIAL IN-OUT UNITS.....	72
4.1	Light Guns.....	72
4.2	Radar Input.....	74
4.3	Crosstell System	81
4.4	Semi-Automatic Height Finder Request Coder.....	84
4.5	Digital Output Coder (DOC System).....	87
4.6	Teletype System (Input and Output).....	92
4.7	Programmed Core Clearing.....	98
SECTION V.....	COMPUTER ALARMS.....	99
SECTION VI.....	COMMON SWITCHES AND PUSH BUTTONS.....	103
TABLE 1.....	THE FLEXOWRITER CODE (Tape Format).....	106
TABLE 2.....	THE "FL" FLEXOWRITER CODE (Alphanumerical Sequence).	107
TABLE 3.....	THE "FL" FLEXOWRITER CODE (Binary Numerical Sequence)	108
TABLE 4.....	WHIRLWIND I INSTRUCTION CODE.....	109
TABLE 5.....	DESIGNATIONS OF IN-OUT EQUIPMENT WWI (si addresses).	110

SECTION I

DEFINITIONS AND INTRODUCTORY MATERIAL

1.1 CENTRAL COMPUTER

1.1.1 General Terms

(1) Sequence

A sequence is the numerical or other arrangement of a set of words stored in or performed on the computer.

(2) Instruction

An instruction is a 16-digit binary word used to control the computer.

(3) Operation

An operation is the 5 digits of an instruction which go to the operation control switch.

(4) Command

A command is a control pulse from the control matrix.

(5) Process

A process is an automatic manipulation initiated by a command.

(6) Modulo (Abbreviated "mod")

A number p modulo q is defined as the numerator of the fractional remainder when p is divided by q.

Example 1: $60 \bmod 13.\ \frac{60}{13} = 4 + \frac{8}{13}$, hence, $60 \bmod 13 = 8$

Example 2: $1.37 \bmod 1.\ \frac{1.37}{1} = 1 + \frac{.37}{1}$, hence, $1.37 \bmod 1 = .37$

1.1.2 Computer Programs

(1) Program

A program is a sequence of actions by which a computer handles a problem. The process of determining the sequence of actions is known as programming.

(2) Flow Diagrams

A flow diagram is a series of general statements of what the computer has to do at various stages in a program. Lines of flow indicate how the computer passes from one stage of the program to another.

(3) Coded Program

Programs and flow diagrams are somewhat independent of computer characteristics, but instructions for a computer must be expressed in terms of a code. A set of instructions that will enable a computer to execute a program is called a coded program, and the process of preparing a coded program is known as coding. Individual coded instructions call for specific operations such as multiply, add, shift, etc.

1.1.3 Computer Components

(1) Registers and Words

A register has 16 binary digit positions, each able to store a one or a zero. A word is a set of 16 binary digits that may be stored in a register, representing either an instruction or a number.

(2) Arithmetic Element

Arithmetic operations take place in the arithmetic element whose main components are three flip-flop registers, the A-Register, the Accumulator, and the B-Register (AR, AC, and BR). The 16 digits positions of AR starting from the left are denoted by AR 0, AR 1, ... , AR 15. The digit positions of AC and BR are denoted in a similar fashion. Words enter AC through AR; BR is an extension of AC to the right.

(3) Primary Storage

Primary storage consists of 6144 registers of magnetic core memory (MCM) with an access time of 7 microseconds, and 32 registers of test storage. There are three shower-stalls of magnetic core memory, two of which contain 17 planes of 1024 cores each. The third and latest shower-stall contains 17 planes of 4096 cores each. The 17 planes represent the 16 binary digits of WWI register and the one digit used for checking purposes. Since WWI is designed to operate with a full complement of 2048 registers of storage, only 2048 of the 6144 available registers may be used at a time. Thus, for manipulation ease core storage is divided into six equal parts of 1024 registers, with the proviso that two parts (fields) be used at one time to provide the required 2048 registers necessary for full computer operation. The six equal parts or "fields" are numbered 0 through 5 and may be chosen by the use of the WWI instruction "change fields," cf. A control system exists which considers register addresses to be in one of two groups: Group A includes register addresses 0 - 1023 inclusive, and Group B includes register addresses 1024 - 2047 inclusive. Any combination of fields can be used with the exception that the same field cannot occupy Group A and Group B locations simultaneously. The 32 toggle-switch registers of test storage and the five flip-flop storage registers, which may be interchanged with any of the 32 toggle-switch storage registers, occupy registers 0 - 31 inclusive, of the field memory used in Group A, which are thus normally unavailable for programming

(4) Input-Output

All information entering or leaving the computer is temporarily stored in the input-output register (IOR). The computer regulates the flow of information between the internal storage and IOR, and also calls for any necessary manipulation of external units.

(5) Control element

This element controls the sequence of computer operations and their execution. The control element takes the instructions one at a time from storage in a sequence described in 1.1.6 (1) where instructions are stored as individual words.

(6) Inter-connections

The four main elements of the computer (storage, control, arithmetic, and input-output) are connected by a parallel communications system known as the bus.

1.1.4 Representation of Instructions

(1) Operation section

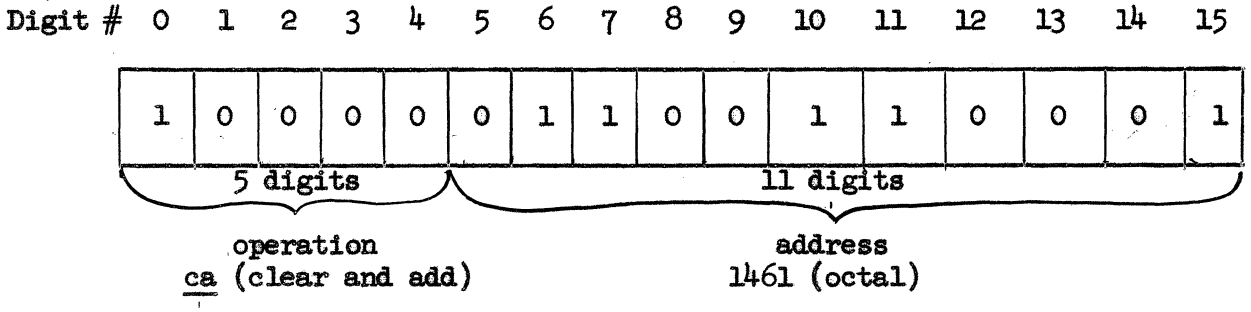
When a word is used to represent an instruction, the first (left-hand) 5 digits, or operation section, specify a particular operation in accordance with the operation code.

(2) Address section

The remaining 11 digits, or address section, are interpreted as a number with the binary point at the right-hand end. For the majority of instructions, this number is the address of the register whose contents will be referred to in the operation. In the instructions slh, slr, srh, srr, clc, and clh, the number specifies the extent of a shift, and may also specify additional variants, such as rounding-off or clearing the B-register; the address section is not used in rd; in rc the address section is referred to only when used with the character generator (see Section 3.8.3 (3)). The address section of the si operation ordinarily selects a particular in-out unit and its mode of operation. (See Sections III and IV)

(3) Example

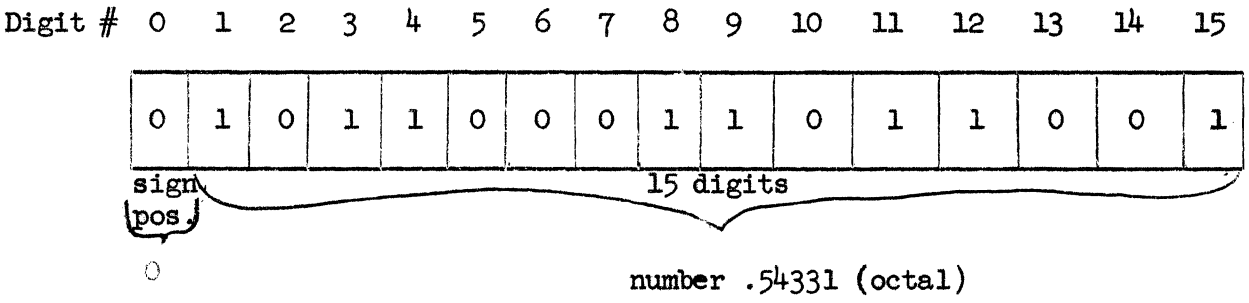
The instruction ca x has the effect of clearing AC (making all the digits zero) and then copying into AC the word that is in the register whose address is x. If q is a word in some register, the operation needed to copy q in AC is not ca q but ca x, where x is the address of the register that contains q. Suppose q is stored in register 1461 (octal):



1.1.5 Representation of Numbers

(1) Single word representations

When a word is used to represent a number, the first digit indicates the sign and the remaining 15 are numerical digits. For a positive number the sign digit is zero, and the 15 numerical digits with a binary point at their left specify the magnitude of the number. The negative of a positive number y is represented by complementing all the digits, including the sign digit, that would represent y. (The complement is formed by replacing every zero by a one and every one by a zero.) In this way a word can represent any multiple of 2^{-15} from $-1 + 2^{-15}$ to $1 - 2^{-15}$. (Numbers $\geq +1$ or ≤ -1 cannot be represented by a single word.) Zero has two representations, either 16 zeros or 16 ones, which are called +0 and -0 respectively.



The octal number above, 0.54331, could also be represented by the decimal fraction +.6941 or the decimal integer +22,745 times 2^{-15}.

(2) Overflow -- increase of range and precision

With single-word representation, the range is limited to numbers between -1×2^{-15} and $1 - 2^{-15}$ inclusive. Programs must be so planned that arithmetic operations will not cause an overflow beyond this range. Overflow will stop the computer in an arithmetic check alarm except where special provision has been made to accommodate the overflow (see sa operation). The range may be extended by using a scale factor, which can be separately stored. Precision can be increased by using two words to represent a number.

(3) Scale factors

In order that computations can be performed involving numbers outside the limited range of $-1 + 2^{-15}$ to $+1 - 2^{-15}$ all numbers with which the computer must deal are made to fall in this range by the application of scale factors. A scale factor is a multiplier, remembered by the programmer, by which a number is multiplied so that it falls within the computer's range. For example, in order to use the octal number 34567 in our computations, it must be multiplied by 2^{-15} to make the product smaller than $+1 - 2^{-15}$. Then the number is represented as $34567. \times 2^{-15}$ equalling .34567 in the computer's range with the scale factor of 2^{-15}.

Scale factors are customarily chosen to be some power of 2, principally as a matter of convenience. During the course of a program, it occasionally becomes necessary to change the scale factor of a number. Since WWI operates in the binary number system, the simple process of shifting a number to the left or right is equivalent to multiplying or dividing by a power of 2. If the number's scale factor is initially a power of 2, then shifting the number will change the scale factor to some other power of 2.

Algebraic principles demand that two numbers can be added together only if their multipliers, or scale factors, are the same; therefore, the quantities 5×2^{-6} and 2×2^{-4} cannot be directly added. First the scale factors must be made equal, for example, by multiplying the second scale factor by 2^{-2} to make it equal to 2^{-6}. In the computer this is accomplished by shifting the quantity 2×2^{-4} two places to the right, making it 2×2^{-6}. The resulting quantities (5×2^{-6}) and (2×2^{-6}) can be added.

In multiplication no adjustment of scale factors is necessary. The result of the multiplication is the product of the two original numbers times the product of the two original scale factors. For example: $(5 \times 2^{-6}) \times (2 \times 2^{-4}) = (5 \times 2) \times (2^{-6} \times 2^{-4}) = 10 \times 2^{-10}$.

Some care must be exercised in the choice of a scale factor. A number in WWI is represented by fifteen binary digits (or 5 octal digits) to the right of the point, roughly equivalent to 4 decimal digits. It is desirable, if possible, to choose the proper scale factor to cause these digit positions to contain significant bits. For example, to compute with the decimal fraction .003512 without a scale factor, only the first 4 digits to the right of the point of which 2 digits are 0 could be used, a precision of 2 digits. However, multiplying by a scale factor of 2^8, the number becomes .899072 of which the first four digits .8991 can be used giving a precision of 4 decimal digits.

1.1.6 Computer Procedure

(1) Sequence of Operations

After the execution of an instruction, the Program Counter (PC) in the control element holds the address of the register from which the next instruction is to be taken. Control calls for this instruction and carries out the specified operation. If the operation is not sp, cp or cf, the address in the program counter then increases by one so that the next instruction is taken from the next consecutive register. The sp, cp and cf instructions may permit a change in this sequential procedure.

(2) Zero

All sums and differences resulting in zero are represented as negative zero (1.111 111 111 111 111) except in the two cases: (+0)+(+0) and (+0)-(-0). The sign of a zero resulting from multiplication, division, or shifting is in accordance with the usual sign convention.

(3) Manipulation of Instructions

Words representing instructions may be handled in the arithmetic element as numbers.

(4) Procedure in the Arithmetic Element

The execution of an addition includes the process of adding in carries; this process treats all 16 digits as if they were numerical digits, a carry from AC 0 being added into AC 15. (This compensation is necessary because of the method of representing negative numbers.) A subtraction is executed by adding the complement. Multiplication, division, scale factoring, shifting (but not cycling), and roundoff are all executed with positive numbers, complementing being performed before and after the process when necessary. For roundoff, the digit in BR 0 is effectively added into AC 15.

(a) BR

In the operations sl, sr, mh, mr, dv, sf, if the initial content of the AC is negative, the AC is complemented before the operation is executed, and if the result should be negative (as in the multiplication of a negative and positive number) the content of the AC is complemented but the content of the BR is not complemented. The net effect is that a number in BR is treated as a positive magnitude, the sign of the number being indicated by the sign digit of AC. Therefore, if a number is to be recalled from BR for further operations, it is necessary to compensate for any change in the sign digit of AC which may have occurred after the number was placed in BR. No complementing of any sort occurs in the execution of the cycle instructions, during which AC and BR may be considered a closed ring of 32 digit positions.

1.2 IN-OUT SYSTEM

A flexible system of terminal devices to permit communication between the computer and both the human controllers and remote external equipment must include many separate input and output units. A centralized selection and control element which accomplishes control functions common to many of the terminal units is the IN-OUT Element. The IOE may be broken up into three major parts: (1) In-Out Switch (IOS) which selects external units and their modes of operation; (2) the In-Out Register (IOR) which provides buffer storage for data being transferred between the computer and external units; and (3) the In-Out Control (IOC).

If a number of in-out instructions are given in a program, since the in-out equipment operates at a much slower speed than the central computer in processing these instructions, some means of halting the computer until a previous in-out instruction has been performed before doing another is required. The Whirlwind computer checks a special flip-flop in the in-out system, called the interlock, whenever an in-out instruction is to be performed. If the interlock is set to a 1 at this time, the computer halts; at the end of the first in-out instruction the interlock will be cleared and the later in-out instruction can then be performed. There exist two ways at present of deciding that an in-out instruction is finished. One is to have a pulse (called a completion pulse) return from the selected equipment to indicate this fact. The other is to count a delay in the computer sufficiently long so that this operation must be completed. The interlock is then cleared at

the end of this delay. Such delay counting is done in the In-Out Delay Counter (IODC). For operations which actually require a sequence of events, such as recording a word of magnetic tape, the IODC may be used to time these various events, and the total delay will be the sum of the individual delays.

The group of computer operations termed "in-out" operations is composed of the operations involved with the transmission of Whirlwind words (instructions or numbers) into and out of the central computer (Arithmetic Element, Magnetic Core Memory, Control). These operations are si (select in-out equipment), bi (block input), rd (read), bo (block output), and rc (record). They are used both for the auxiliary storage of information by the magnetic drums and tape units and for the input of information to, or the output of information from the computer by the various pieces of terminal equipment.

1.2.1 Si Instructions

The action of the instruction si pqrs is to select a particular in-out unit and prepare it to start operation in a specified mode designated by the octal address digits pqrs. In general, q designates the class of equipment (for example: q = 1 selects the magnetic tape equipment), and r and s together designate the number of the unit and mode of operation. An si normally precedes one or more of the other in-out instructions bi, rd, bo, and rc, except in certain special cases involving the camera, clock, computer stop orders, etc., where only one si instruction is used.

Any instructions other than in-out instructions may intervene between the si and its associated bi, rd, bo, or rc, without affecting the in-out process. (However, physical motion of the selected equipment may continue which could result in a program alarm, etc.) Following an si instruction which specifies a read mode, the computer must not execute another si until the process initiated by the earlier si has ended. Since this is not insured by the computer hardware, it is necessary to program at least one rd or bi instruction after every si which selects a read mode.

1.2.2 Assigned si Addresses

All the si addresses which have been assigned functions are listed under the equipment to which they apply. A complete list of assigned si addresses is given in Table No. 5 at the end of this memorandum. Unassigned si addresses may not be used indiscriminately. Certain unassigned addresses are "illegal", that is, they may cause an in-out unit to operate in an unpredictable fashion. Other unassigned addresses are reserved for possible use at a later date.

1.2.3 Stop Instructions

The si operation is also used to provide a "stop", either to stop the computer or to stop any in-out unit which does not stop automatically (that is, a magnetic tape unit or the photoelectric reader). Si 0 will stop the computer. Si 1 will stop the computer only if the "STOP ON si 1" switch is ON. Si (any assigned address) will stop any other free running in-out unit which may be in motion, without stopping the computer; however, if no in-out unit need be selected by this si stop instruction, the unique designation si 630 (octal) or si 408 (decimal) should be used, both for program clarity and for safety of operation.

1.2.4 rd and rc Instructions

After an si instruction has been given which selects a unit to operate in the read mode, the next available word from this piece of in-out equipment is transferred from this equipment into the IOR (In-Out Register) before the action of the si instruction is regarded as complete. The following rd (or bi) instruction completes the transfer of information by bringing the word from the IOR into the Accumulator by way of the A-Register. When an si for recording on a certain piece of in-out equipment is given, the word is not transferred from the central computer until the following rc instruction is given. When this rc is given, the word is transferred from the central computer to the selected in-out equipment, by way of the IOR. The address section of the rd instruction is not used at present, and the address section of the rc instruction is used only with the vector and character generators.

1.2.5 Block Transfer Instructions

Blocks of words can be transferred from the high-speed memory of the central computer to certain pieces of in-out equipment, and vice versa, by means of the block transfer orders bi and bo. The bi order is used to transfer a block of words from certain pieces of in-out equipment into a selected position of the high-speed memory, while the bo order is used to transfer a group of words from a selected position in the high-speed memory to a certain position in the in-out equipment, or to record a block of characters on certain output units.

1.2.6 Use of AC to Specify In-Out Operation

The in-out sequences for specific in-out units are given in Sections III and IV of this memorandum. It will be obvious that, in certain cases, proper execution of the in-out instructions requires specific contents in AC at the time of the in-out instruction. Since the required AC content may be derived in different ways, instructions for inserting the necessary information into AC are not included in the sequences.

SECTION II

WHIRLWIND I OPERATION CODE

The Whirlwind I Operation Code has been rewritten to bring it up to date and to incorporate all notes, wherever possible, with the specific operations to which they apply, regardless of the undue repetition. Included under each operation are the average time of execution, the function, the contents (if altered) of AC, BR, AR, IOR, SAM (Special Add Memory), and the register whose address is x after the operation, and possible alarms.

2.1 Abbreviations

AC = Accumulator

AR = A-Register

BR = B-Register

SAM = Special Add Memory

PC = Program Counter

F{} = Fractional part of quantity in

I{} = Integral part of quantity in

IOR = In-Out Register

MCM = Magnetic Core Memory

x = Address of a storage register

n = A positive integer

C = Original contents of register ( )

2.2 Contents of Various Registers

The contents of AC, BR, AR, IOR, SAM, and the register whose address is x are undisturbed unless the contrary is stated.

2.3 Execution Times

The times given are the times for the execution of single instructions which are stored in MCM and which refer to addresses in MCM.

2.4 In-Out Operations

Operations which call for the transmission of information to and from various units of terminal equipment (termed "in-out operations") are described

briefly in the Operation Code. Details of the actual application of these operations (si, bi, rd, bo and rc) appear in Sections III and IV of this memorandum.

2.5 Three-letter Operations

The three-letter operations slh, slr, srh, srr, clc and clh utilize part of the address section of the instruction (namely, digit 6) to specify the operation. If an address is inserted in one of these instructions by a "ta" or "td" operation, care must be taken to maintain the presence or absence of digit 6 in the address of the modified instruction. The two-letter designations sl, sr, cl, are ambiguous and cannot be used in programs, but they may be used in general descriptions and comments.

2.6 Instruction Summary

A single-sheet summary of all Whirlwind I instructions is attached at the end of this memorandum.

2.7 Instructions

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
si pqr	select in-out unit/stop	00000	0.00	0	30 microsec	si

Stop any in-out unit that may be running. Select a particular in-out unit and start it operating in a specified mode, designated by the digits p q r; or, stop the computer. Si 0 will stop the computer; si 1 will stop the computer only if the "Conditional Stop" switch is ON. A program alarm may occur if the computer is not ready to receive information transmitted to it from the selected in-out unit (e.g., the use of an si address if si selects Magnetic Tape, or PETR without the necessary rd instruction following). Also, an inactivity alarm may result when using an rc or bo with an si instruction which selects the read mode. (For further details, see Section V, COMPUTER ALARMS.)

---

illegal instruction		00001	0.04	1		illegal instruction

If an attempt is made to use this operation, a check alarm will result.

---

bi x	block transfer in	00010	0.10	2	8 millisec avg., 16 millisec max. for 1st word; 16 microsec for each additional word	bi

Transfer a block of n words or characters from an in-out unit to MCM, where register x is the initial address of the block in MCM, and $\pm$ n times 2^{-15} is contained in AC. The computer is stopped while the transfer is taking place. After a block transfer, AC contains the address which is one greater than the MCM address at which the last word was placed; AR contains the initial address of the block in MCM. (For further details, see Sections III and IV with reference to particular equipments.)

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
rd x	read	00011	0.14	3	15 microsec	rd

Transfer word from IOR to AC, then clear IOR. (Wait, if necessary, for information to arrive in IOR from an in-out unit.) Contents of AR are identical to contents of AC. The address section of the instruction has no significance. (For further details, see Sections III and IV.)

---

bo x	block transfer out	00100	0.20	4	8 millisec avg., 16 millisec max. for 1st word; 16 microsec for each additional word	bo

Transfer block of n words from MCM to an in-out unit where x is the initial address of the block in MCM, and $\pm$ n times 2^{-15} is contained in AC. The computer is stopped while the transfer is taking place. After the block transfer, AC contains the address which is one greater than the MCM address from which the last word was taken and stored; AR contains the initial address of the block in MCM. (For further details, see Sections III and IV.)

---

rc x	record	00101	0.24	5	22 microsec	rc

Transfer contents of AC via IOR to an in-out unit. IOR is cleared only after an rc used as a display instruction. The address section of the instruction has no significance except when used with the character generator. (For further details, see Sections III and IV.)

---

sd x	sum digits	00110	0.30	6	22 microsec	sd

The sum of the original contents of digit i of AC and original contents of digit i of register x becomes stored in digit i of AC. The final value of digit i of AC is 0 if the values of digit i of AC and of register x are alike; the final value of digit i of AC is 1 if the values of digit i of AC and of register x are different.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
cf pqr	change fields	00111	0.34	7	15 microsec	cf

The address section does not refer to a register of storage in this instruction, but supplies information to the computer requesting a change in fields Group A and/or B. When the field to be changed contains the program, it is necessary for the cf instruction to perform like an sp instruction. Digit 7 of the cf address section causes the contents of the accumulator to be read to the program counter (PC) prior to the field change; thus, program continuity can be preserved during field changes. The A-Register will contain the original PC address plus one upon completion of the cf instruction. The digit allocation for the cf word is as follows:

digits 0 - 4 : 00111 cf order.

digit 5 : spare.

digit 6 : examine feature - causes contents of core memory Group A control and Broup B control registers to be read into the Accumulator.

digit 7 : sp enable - reads content of AC to PC to establish starting point of program in the new field.

digit 8 : change Group A field enable. (If, when the examine feature of digit 6 is requested, there is a "1" in digit 8, the content of Group A control will be changed before read-out to the A-Register takes place.)

digit 9 : change Group B field enable. (If, when the cf examination feature is requested, there is a "1" in digit 9, the content of Group B control will be changed before read-out to the A-Register takes place.)

digits 10 - 12 : contain field designation for Group A (registers 40 - 1777).

digits 13 - 15 : contain field designation for Group B (registers 2000 - 3777).

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
ts x	transfer to storage	01000	0.40	8	22 microsec	ts

Transfer contents of AC to register x. The original contents of register x are destroyed.

---

td x	transfer digits	01001	0.44	9	29 microsec	td

Transfer last 11 digits of AC to last 11 digit positions of register x. The original contents of the last 11 digit positions of register x are destroyed.

---

ta x	transfer address	01010	0.50	10	29 microsec	ta

Transfer last 11 digits of AR to last 11 digit positions of register x. The original contents of the last 11 digit positions of register x are destroyed. The ta operation is normally executed after an sp or cp instruction in connection with sub-programming; less frequently after ao, sf or other operations.

---

ck x	check	01011	0.54	11	22 microsec	ck

Operates in two modes depending upon position of console switch (at T.C.) labelled "Program Check Alarm on Special Mode." The NORMAL MODE is selected if the switch is off or down. Normal Mode compares contents of AC with contents of register x. If contents of AC are identical to contents of register x, proceed to next instruction; otherwise, stop the computer and give a "check register alarm." (Note: +0 is not identical to -0.) SPECIAL MODE is chosen if switch is on. Special Mode operates in same way as above if the numbers being checked agree. If there is disagreement, no check alarm will occur but the program counter (PC) will be indexed by one, causing the next instruction to be skipped.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
ab x	add BR	01100	0.60	12	29 microsec	ab

Add the contents of the B-Register to the contents of register x and store the result in the AC and register x. The contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $|C(x) + C(BR)| \geq 1$. (dm x or clh 16 puts C(AC) into BR.)

---

ex x	exchange	01101	0.64	13	29 microsec	ex

Exchange contents of AC with contents of register x. (Original contents of AC in register x; original contents of register x in AC and AR.) Ex 0 will clear AC without clearing BR.

---

cp x	conditional program	01110	0.70	14	15 microsec	cp

If number in AC is negative, proceed as in sp. If number in AC is positive, proceed to next instruction, clear the AR and place in the last 11 digit positions of the AR the address of this next instruction.

---

sp x	sub-program (transfer control)	01111	0.74	15	15 microsec	sp

Take next instruction from register x. If the sp instruction was at address y, clear the AR and store y + 1 in the last 11 digit positions of AR. x becomes the contents of the PC.

---

ca x	clear and add	10000	1.00	16	22 microsec	ca

Clear AC and BR, then obtain contents of SAM (+1, 0, or -1) times 2^{-15} and add contents of register x, storing result in AC. The contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $|C(x) + C(SAM)\ 2^{-15}| = 1$.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
cs x	clear and subtract	10001	1.04	17	22 microsec	cs

Clear AC and BR, then obtain contents of SAM (+1, 0 or -1) times 2^{-15} and subtract contents of register x, storing result in AC. The contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $\left|-C(x) + C(SAM)\ 2^{-15}\right| = 1$.

---

ad x	add	10010	1.10	18	22 microsec	ad

Add the contents of register x to contents of AC, storing result in AC. The contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $\left|C(AC) + C(x)\right| \geq 1$.

---

su x	subtract	10011	1.14	19	22 microsec	su

Subtract contents of register x from contents of AC, storing result in AC. The contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $\left|C(AC) - C(x)\right| \geq 1$.

---

cm x	clear and add magnitude	10100	1.20	20	22 microsec	cm

Clear AC and BR, then obtain contents of SAM (+1, 0, -1) times 2^{-15} and add magnitude of contents of register x, storing result in AC. The magnitude of the contents of register x appears in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $\left|C(x)\right| + C(SAM)\ 2^{-15} = 1$.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
sa x	special add	10101	1.24	21	26 microsec	sa

Add contents of register x to contents of AC, storing fractional result in AC and retaining in SAM any overflow (including sign) for use with next ca, cs or cm instruction. Between sa and the next ca, cs or cm instruction for which the sa is a preparation, the use of any instruction which clears SAM will result in the loss of the overflow with no other effect on the normal function of the intervening operation. The following operations clear SAM without using its contents: sd, su, sa, ao, dm, mr, mh, dv, sl, sr and sf. Ca, cs or cm clear SAM after using its contents. If the overflow resulting from the sa is to be disregarded, care must be taken to destroy it before the next ca, cs or cm instruction. The contents of register x appear in AR. SAM is cleared before, but not after, the addition is performed.

---

ao x	add one	10110	1.30	22	29 microsec	ao

Add the number 1 times 2^{-15} to contents of register x, storing the result in AC and in register x. The original contents of register x appear in AR. SAM is cleared. Overflow may occur giving an arithmetic check alarm if $C(x) + (1 \times 2^{-15}) = 1$.

---

dm x	difference of magnitudes	10111	1.34	23	22 microsec	dm

Subtract the magnitude of contents of register x from the magnitude of contents of AC, leaving result in AC. The magnitude of contents of register x appears in AR. SAM is cleared. BR will contain the initial contents of the AC. If $|C(AC)| = |C(x)|$, the result is -0.

---

mr x	multiply and roundoff	11000	1.40	24	34-41 microsec	mr

Multiply contents of AC by contents of register x. Roundoff result from BR to 15 significant binary digits and store it in AC. If BR(0) = 1, roundoff is 2^{-15}; if BR(0) = 0, roundoff is 0. Clear BR. The magnitude of contents of register x appears in AR. SAM is cleared. Sign of AC is determined by sign of product. The minimum time of 34 microseconds occurs if there are all 0's in $|AC|$; the maximum time of 41 microseconds occurs when there are all 1's in $|AC|$. The average time is 37.5 microseconds.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
mh x	multiply and hold	11001	1.44	25	34-41 microsec	mh

Multiply contents of AC by contents of register x. Retain the full product in AC and in the first 15 digit positions of BR, the last digit position of BR being cleared. The magnitude of contents of register x appears in AR. SAM is cleared. The sign of AC is determined by sign of product. Result in (AC + BR) is a double register product. The time is determined the same as for mr.

---

dv x	divide	11010	1.50	26	71 microsec	dv

Divide contents of AC by contents of register x, leaving 16 binary digits of the quotient in BR and ± 0 in AC according to the sign of the quotient. The instruction slr 15 following the dv operation will roundoff the quotient to 15 binary digits and store it in AC. Let u and v be the numbers in AC and register x, respectively, when the instruction dv x is performed. If $|u| < |v|$, the correct quotient is obtained and no overflow can arise. If $|u| > |v|$, the quotient exceeds unity and a divide-error alarm will result. If $u = v \neq 0$, the dv instruction leaves 16 "ones" in BR; roundoff in a subsequent slr 15 will cause overflow and give an arithmetic overflow check alarm. If $u = v = 0$, a zero quotient of the appropriate sign is obtained. The magnitude of contents of register x appears in AR. SAM is cleared.

---

slr n	shift left and roundoff	11011-0	1.54	27	15 + .8n microsec	slr

Shift fractional contents of AC (except sign digit) and BR to the left n places. The positive integer n is treated modulo 32; digits shifted left out of AC 1 are lost. (Shifting left n places is equivalent to multiplying by 2^n, with the result reduced modulo 1.) Roundoff the result to 15 binary digits and store it in AC. Clear BR. Negative numbers are complemented before the shift and after the roundoff; hence, ones appear in the digit places made vacant by the shift of a negative number. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction slr n must be a zero to distinguish slr n from slh n described below. The instruction slr 0 simply causes roundoff and clears BR. SAM is cleared. Roundoff (ρ) may cause overflow with a consequent arithmetic check alarm if $|F\{C(AC + BR)2^n\} + \rho| = 1$. The execution time varies according to the size of n.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
slh n	shift left and hold	11011-1	1.54	27	15 + .8n microsec	slh

Shift contents of AC (except sign digit) and BR to the left n places. The positive integer n is treated modulo 32; digits shifted left out of AC 1 are lost. (Shifting left n places is equivalent to multiplying by 2^n, with the result reduced modulo 1.) Leave final product in AC and BR. Do not roundoff or clear BR. Negative numbers are complemented in AC before and after the shift; hence, ones appear in the digit places made vacant by the shift of a negative number. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction slh n must be a one to distinguish slh n from slr n described above. SAM is cleared. The execution time depends upon the size of the n.

---

srr n	shift right and roundoff	11100-0	1.60	28	15 + .8n microsec	srr

Shift contents of AC (except sign digit) and BR to the right n places. The positive integer n is treated modulo 32; digits shifted right out of BR 15 are lost. (Shifting right n places is equivalent to multiplying by 2^{-n}.) Roundoff the result to 15 binary digits and store it in AC. Clear BR. Negative numbers are complemented before shift and after the roundoff; hence, ones appear in the digit places made vacant by the shift of a negative number. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction srr n must be a zero to distinguish srr n from srh n described below. The instruction srr 0 simply causes roundoff and clears BR. SAM is cleared. Roundoff (in an srr 0) may cause overflow with a consequent arithmetic check alarm. The time depends upon the size of the integer n.

---

srh n	shift right and hold	11100-1	1.60	28	15 + .8n microsec	srh

Shift contents of AC (except sign digit) and BR to the right n places. The positive integer n is treated modulo 32; digits shifted right out of BR 15 are lost. (Shifting right n places is equivalent to multiplying by 2^{-n}.) Do not roundoff the result or clear BR. Result is stored in AC and BR. Negative numbers are complemented in AC before and after the shift; hence, ones appear in the digit places made vacant by the shift of a negative number. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction srh n must be a one to distinguish srh n from srr n described above. SAM is cleared. The time depends upon the size of the integer n.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
sf x	scale factor	11101	1.64	29	30-78 microsec	sf

Multiply the contents of AC and BR by 2 often enough to make the positive magnitude of the product equal to or greater than 1/2. Leave the final product in AC and BR. Store the number of multiplications in AR and in last 11 digit places of register x, the first 5 digits being undisturbed in register x. If all the digits in BR are zero and AC contains ± 0, the instruction sf x leaves AC and BR undisturbed and stores the number 33 times 2^{-15} in AR and in the last 11 digit positions of register x. Negative numbers are complemented in AC before and after the multiplication (by shifting); hence, ones appear in the digit places made vacant by the shift. SAM is cleared. The time varies according to the number of 0's between the binary point and first 1 in magnitude of binary fraction represented by the number in AC and BR. The minimum time of 30 microseconds occurs when $|AC + BR| \geq 1/2$; the maximum time of 78 microseconds occurs when $|AC + BR| = 0$.

---

clc n	cycle left and clear (BR)	11110-0	1.70	30	15 + .8n microsec	clc

Shift the full contents of AC (including sign digit) and BR to the left n places. The positive integer n is treated modulo 32; digits shifted left out of AC 0 are carried around into BR 15 so that no digits are lost. No roundoff. Clear BR. With the clc operation there is no complementing of AC either before or after the shift; the actual numerical digits in AC and BR are cycled to the left. The digit finally shifted into the sign digit position determines whether the result is to be considered a positive or negative quantity. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction clc n must be a zero to distinguish clc n from clh n described below. The instruction clc 0 simply clears BR without affecting AC. The execution time depends on the size of the integer n.

---

clh n	cycle left and hold	11110-1	1.70	30	15 + .8n microsec	clh

Shift the full contents of AC (including sign digit) and BR to the left n places. The integer n is treated modulo 32; digits shifted left out of AC 0 are carried around into BR 15 so that no digits are lost. With the clh operation there is no complementing of AC either before or after the shift; the actual numerical digits in AC and BR are cycled to the left. Result is kept in AC and BR. The digit finally shifted into the sign digit position determines whether the result is to be considered a positive or negative quantity. Digit 6 (the $2^9 = 512$ digit of the address) of the instruction clh n must be a one to distinguish clh n from clc n described above. The instruction clh 0 does nothing. The execution time depends on the size of the integer n.

Operation	Function	Binary Code	Octal Equivalent	Decimal Equivalent	Time	Operation
md x	multiply digits with no roundoff (p)	11111	1.74	31	22 microsec	md

The product of the ith digit of the AC multiplied by the ith digit of register x becomes stored in the ith digit of the AC. The final value of the ith digit of the AC is 1 if the initial value of the ith digits of the AC and register x are both 1; otherwise, the final value of the ith digit of the AC is 0. AR contains the complement of the final contents of the AC.

SECTION III

MAJOR IN-OUT UNITS

3.1 MAGNETIC DRUMS

There are two magnetic drums used for storage purposes in WWI - the auxiliary drum and the buffer drum. The auxiliary magnetic drum provides 24,576 (d) registers of "intermediate speed storage," and the buffer magnetic drum provides 20,480 (d) registers of auxiliary and of buffer storage, where each register in both drums can store a 16-digit binary word (with the exception of the 8192 register buffer section of the buffer drum where only 14 digits can be stored). A word can be transferred to or read from any drum register.

The registers in each drum are divided into groups of 2048 (d) or 4000 (o) resulting in 12 auxiliary drum groups consecutively numbered from 0 to 11, and in 10 buffer drum groups numbered 0_A, 0_B, 1_A, 1_B and 2 through 7. (See Sections 3.1.2 and 3.1.3 for restrictions in the use of drum groups.)

3.1.1 Programming for the Magnetic Drums

(1) Register Numbering

A drum address is specified by a 16-digit binary word - digit 0 indicates which drum is being addressed (0 for auxiliary drum, 1 for buffer drum), digits 1 - 4 specify the group number, and digits 5 - 15 specify the storage address. Within any register group, the storage addresses are treated modulo 2048. For example, in one register group a block transfer starting at address 2047 will deal in turn with registers 2047, 0, 1, 2, etc.

(2) Access Time

To gain access to a specific register on the drum takes, on the average, 8 milliseconds, equal to the time for one-half revolution of the drum. An additional 64 microseconds delay occurs on si instructions for reading from the drum.

(3) Register Selection

The drum address to be selected is determined by the si instruction and by any necessary portions of the contents of AC at the time the si is executed. The si instruction may call for a new group number, a new initial storage address, neither, or both. When a new group number is needed, it is taken from digits 1 - 4 of AC. When a new initial storage address is needed, it is taken from digits 5 - 15 of AC. Either the group selected on the drum can remain selected until an si instruction specifically calls for a change of group or, by adding 1000 to the original auxiliary drum si, a block transfer is per-

mitted to run off the end of one drum group to the beginning of the next drum group. The next storage address selected will be one greater than the storage address most recently referred to, unless an si instruction specifically calls for a new initial storage address. To provide for all the cases above, there are eight possible ways for an si instruction to specify a register:

(a) Select no new group or initial address

(b) Select new initial storage address only

(c) Select new group only

(d) Select both new group and new initial storage address

(e) Select no new group or initial address and be able to continue on to the next consecutive groups

(f) Select new initial storage address and be able to continue on to the next consecutive groups

(g) Select new group and be able to continue on to the next consecutive groups

(h) Select both new group and new initial storage address and be able to continue on to the next consecutive groups

In addition to the above, the si instruction also selects either reading or recording with no addition.

(4) Recording on Magnetic Drums

(a) Recording Single Words

Programming for recording on the auxiliary drum is as follows:

si a — selects the specified drum and the record mode. If the si instruction calls for a new group number, it is selected in accordance with the contents of digits 1 - 4 of AC and the group change delay, if any, is counted. If the si calls for a new initial storage address, it is selected in accordance with the contents of digits 5 - 15 of AC.

rc-- — records the contents of AC at the next address called for by the si instruction, or at the next consecutive address following the last address at which a word was recorded. The computer cannot perform another in-out operation until the in-out equipment completes the recording process which takes an average of 8 milliseconds and a maximum of 16 milliseconds. An rc instruction is required for each word to be recorded. As many rc instructions as necessary may be used before the next si instruction. Any number of instructions other than in-out instructions may precede each rc.

(b) Recording by Block-Transfer Instruction

A bo instruction may be substituted for a series of rc instructions. The address of the bo must be the initial address of the block to be taken from MCM (magnetic core memory), and ±n, the number of words to be recorded, must be stored (times 2^{-15}) in AC. The block transfer will require an average of 8 milliseconds and a maximum of 16 milliseconds for the first word to be recorded, and 16 microseconds for each additional word. If the block transfer involves both registers 2047 and 0 of the same drum group, in that sequence, an additional 16 milliseconds is required to complete the transfer. Any sequence of rc and bo instructions may follow a single si for the record mode.

(5) Reading from Magnetic Drums

(a) Reading from the Drums

Programming for reading from the auxiliary drum is as follows:

si b — selects the specified drum and the read mode. If the si instruction calls for a new group number, it is selected in accordance with the contents of digits 1 - 4 of AC. If the si calls for a new initial storage address, it is selected in accordance with digits 5 - 15 of AC. Reads into IOR the word from the chosen drum address. The time required to obtain the word is an average of 8 milliseconds and a maximum of 16 milliseconds (plus any drum group change delay if a group change is necessary). One, and only one, rd instruction should intervene between this and the next si instruction.

rd-- — transfers the word in IOR to AC, then clears IOR.

(b) Reading by Block-Transfer Instruction

A bi instruction may be substituted for a series of rd instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred, and ±n, the number of words to be read, must be stored (times 2^{-15}) in AC. The block transfer will require an average of 8 milliseconds and a maximum of 16 milliseconds for the first word to be read, and 16 microseconds for each additional word. If the block transfer involves both registers 2047 and 0 of the same drum group, in that sequence, an additional 16 milliseconds is required to complete the transfer. Only one bi instruction should follow an si instruction.

(c) Zero-Length Block Transfers on bi and bo

The use of a bi instruction calling for a block transfer, zero words in length, will result in the indicated word being read but not transferred into AC. The reading of the word actually is initiated by the preceding si instruction, hence one word is already read by the time the bi is ready to be performed. If the bi calls for the transfer of zero words, the word already read is simply discarded. Zero-length block transfers on bo will always be performed correctly; i.e., no recording will take place.

3.1.2 Restrictions on the use of the Auxiliary Drum

The twelve groups of the auxiliary drum are all available to the programmer but there are restrictions on the use of drum groups 0 and 11. Drum group 0 of the auxiliary drum is ordinarily not used to store data because pushing of the read-in button causes the contents of the cores to be blocked out to drum group 0, thus destroying the information stored there. Since drum group 11 has a particular program (the "Drum Input Program"), permanently stored on it, its recording circuits are normally disabled. Thus, the programmer under ordinary circumstances cannot use drum group 11 and can use drum group 0 only if its use by the input program does not invalidate his use of it.

3.1.3 Restrictions on the use of the Buffer Drum

(1) Drum Groups 2 - 7

Of the ten drum groups of the buffer drum, only groups 2 - 7 are normally available to the programmer for auxiliary storage. These six available groups of the buffer drum, called the auxiliary section, are also used to hold some of the frequently used utility programs such as the Comprehensive System Conversion Program, Post Mortem Program, etc. These programs will be destroyed by recording on the buffer drum which, while permitted, requires that these programs be rewritten on the buffer drum when they are needed again. Thus, the buffer drum auxiliary storage groups should be used only if the auxiliary drum is not available, or if its capacity is not sufficient for the program's needs.

(2) Drum Groups 0 and 1

(a) The physical structure of drum groups 0 and 1, the buffer section of the buffer drum, is the same as the auxiliary section, except that each group is further divided into slots, each consisting of 400(o) or 256 (d) registers. (Since the buffer section of the buffer drum is used primarily for radar input, it is discussed in Section IV.) The present initial drum addresses for reading the various slots are as follows:

Slot	Drum Address	Slot	Drum Address
0	0.00000	10	0.04000
1	0.01000	11	0.05000
2	0.02000	12	0.06000
3	0.03000	13	0.07000
4	0.00001	14	0.04001
5	0.01001	15	0.05001
6	0.02001	16	0.06001
7	0.03001	17	0.07001

(b) Drum Interlace

The interlace of the auxiliary section, groups 2 - 7, permits block orders to be executed at the rate of a word every 16 microseconds; however, the operation time of block orders in the buffer section is only a word every 64 microseconds.

(c) Switching Fields

Groups 0 and 1 of the buffer drum because of their connection to the special radar input equipment have the unique feature of physically consisting of four groups called 0_A, 0_B, 1_A, and 1_B. These groups must be paired according to corresponding letters, i.e., 0_A with 1_A, and 0_B with 1_B. While one of these two pairs is connected to external equipment recording information, the other pair is connected to the computer reading or recording information. After the data in one pair of the drums is processed by the computer, a method of being able to transfer to the data coming in from the external equipment is necessary so that this new data can also be processed. This cyclical selection of first one pair of the groups, then the other, etc., is made possible by the use of the "switch fields" instruction si 734.

(d) Additional Restrictions

The auxiliary drum and the auxiliary storage section of the buffer drum consists of 16 bit words. However, groups 0_A, 0_B, 1_A, and 1_B only handle 14 bit words. Transfers to and from MCM involve bits 2 - 14 of words in MCM.

Normal use of these buffer groups involves reading into the computer from external data sources. However, in connection with special output equipments (see Section IV), or for checking purposes, it is also possible to record on these buffer groups. In recording, although only bits 2 - 14 are transferred from MCM, the sign bit (digit 0) of words in MCM must be a "one" for control purposes. (i.e., negative numbers only may be transferred to these groups.)

3.1.4 si Addresses

(1) si Addresses for Auxiliary Drum

The si addresses referring to the auxiliary drum are as follows:

(a) Read Mode

no address specified	si 700 (o) or si 448 (d)
select new initial address (angular position)	si 701 (o) or si 449 (d)
select new group	si 702 (o) or si 450 (d)
select new group and address	si 703 (o) or si 451 (d)
no address specified with ability to read around the end of the group to the next groups	si 1700 (o) or si 960 (d)

select new initial address (angular position) with ability to read around the end of the group to the next groups	si 1701 (o) or si 961 (d)
select new group with ability to read around the end of the group to the next groups	si 1702 (o) or si 962 (d)
select new group and address with ability to read around the end of the groups to the next groups	si 1703 (o) or si 963 (d)

(b) Record Mode

no address specified	si 704 (o) or si 452 (d)
select new initial address (angular position)	si 705 (o) or si 453 (d)
select new group	si 706 (o) or si 454 (d)
select new group and address	si 707 (o) or si 455 (d)
no address specified with ability to record around the end of the group to the next groups	si 1704 (o) or si 964 (d)
select new initial address (angular position) with ability to record around the end of the group to the next groups	si 1705 (o) or si 965 (d)
select new group with ability to record around the end of the group to the next groups	si 1706 (o) or si 966 (d)
select new group and address with ability to record around the end of the group to the next groups	si 1707 (o) or si 967 (d)

(2) si Addresses for Buffer Drum

(a) Read Mode

no address specified	si 710 (o) or si 456 (d)
select new initial address (angular position)	si 711 (o) or si 457 (d)
select new group	si 712 (o) or si 458 (d)
select new group and address.	si 713 (o) or si 459 (d)

no address specified with ability to read from the end of the group to the beginning of the next groups	si 1710 (o) or si 968 (d)
select new initial address (angular position) with ability to read from the end of the group to the next groups	si 1711 (o) or si 969 (d)
select new group with ability to read from the end of the group to the next groups	si 1712 (o) or si 970 (d)
select new group and address with ability to read from the end of the group to the next groups	si 1713 (o) or si 971 (d)

(b) Record Mode

no address specified	si 714 (o) or si 460 (d)
select new initial address (angular position)	si 715 (o) or si 461 (d)
select new group	si 716 (o) or si 462 (d)
select new group and address	si 717 (o) or si 463 (d)
no address specified with ability to record from the end of the drum group to the next drum groups	si 1714 (o) or si 972 (d)
select new initial address (angular position) with ability to record from the end of the drum group to the next groups	si 1715 (o) or si 973 (d)
select new group with the ability to record from the end of the drum group to the next groups	si 1716 (o) or si 974 (d)
select new group and address with the ability to record from the end of the drum group to the next groups	si 1717 (o) or si 975 (d)

(c) Switch Fields

si 734 (o) or si 476 (d)

3.2 MAGNETIC TAPE UNITS

The five magnetic tape units used with WWI will record and read 16-digit binary words. Normally these words are recorded in blocks of arbitrary lengths (set by programmer). The start of a block is identified by a block mark automatically recorded when the si instructions for recording on the tape are given. Normal running speed of the tape is 30 inches per second. When an order affecting the state of motion of the tape is given, approximately 5.5 milliseconds are required before the tape motion is affected. Thus a tape unit which is running and is deselected by some other si instruction will continue to run at normal speed for about 5.5 milliseconds, and then will decelerate to a stop in about 0.5 milliseconds (but recorded data passing under the heads during this period will not be affected). A tape unit running in a given direction and instructed to change direction will continue to move in the original direction for about 5.5 milliseconds, then decelerate until it is moving in the opposite direction at full speed, which requires about 1 millisecond. The tape units are free-running units; i.e., once started by an si instruction, they run free until stopped by another si instruction.

Of the five tape units, 0, 1, 2, 3 and 4, a maximum of four may be actually connected to the computer at any time and three to the associated printout systems A, B and C, with one tape unit always available for auxiliary storage. In normal operation, unit 0 (which stores computer utility programs) is available only for reading and is not available for delayed printout.

3.2.1 Modes of Operation of the Tape Units (Determined by si Instructions)

(1) Record

Words are recorded on the tape by proper combinations of si instructions for recording and successive rc orders. The si order for recording causes a block mark to be recorded on the tape. It is not possible to record on the magnetic tape by the block transfer instruction bo.

(2) Read

Words are read from the magnetic tape by use of the proper si instructions followed by one or more rd or bi orders. The si order for reading causes the tape unit to run until a block mark is detected before any words may be read.

(3) Re-Record

In the re-record mode, the tape unit starts in the read mode and continues to read until a block mark is detected; it then switches to the record mode.

(4) Stop

Two methods of stopping the magnetic tape units can be used. The first method is to deselect the unit by giving any si instruction not requiring that unit to move. (However, if the program does not require a specific si instruction, the unique designation si 630 (o) or si 408 (d) should be used.) When the unit is deselected in this fashion it ignores but does not disturb the data over which it passes while it is stopping.

However, if it is desired to stop the tape in a region of cleared tape, which is the normal procedure after recording (to clear old information from the tape between blocks, for example), si instructions for stopping in cleared area should be used. In this case, the selected unit switches to the record mode in the indicated direction for 14.3 milliseconds, then reverses direction, still in the record mode, and 14.3 milliseconds later the unit is deselected. Since the unit erases tape passing under the head while in the record mode, when no rc orders are given, this combination erases about 0.6 inches of tape beyond the point at which it was given, then reverses direction and finally is deselected, so that the tape stops in this cleared area just in front of the point at which the order was given. No other instructions (internal or in-out) can be performed until the deselection of the unit occurs. Another method of insuring a cleared space after a recorded block is to program a delay before stopping the unit.

3.2.2 Assembly and Disassembly of Words

A 16-digit word is in actuality recorded as eight pairs of digits on magnetic tape, the word being automatically disassembled by digit pairs in the IOR. In the read mode, the original word will be assembled properly only if the tape is running in the same direction as it was when recorded. If the tape is read in the direction opposite to that in which it was recorded, the resulting words must be unscrambled by a special subroutine.

3.2.3 Lock-in-Read Mode

To guarantee preservation of important information, the system allows any of four units (1, 2, 3 and 4) to be locked in the read mode by throwing a toggle switch on the unit's auxiliary control panel to the lock-in-read position. An indicator light on this panel, as well as one on the switch monitor panel in T.C. 3, will signify when a unit is locked-in-read. Unit zero is (semi)permanently locked-in-read.

3.2.4 Special Modes of Operation

For normal operation tape units have their power on, are attached to the computer, are in the automatic mode, do not have erase circuits connected, and are not locked in read. When all these things are true, the computer may read or record in standard fashion. Often, however, tape units are used in some special mode or are left in some special mode by accident. If an attempt is made to read or record when a unit is in a special mode, definite things will occur depending upon the particular mode selected. The following table explains the results of an attempt to read or record when a unit is in a special mode. The entries in this table are related to the following key:

(1) The status of the unit is not affected.

(2) The unit moves but there is no information transfer.

(3) The computer goes on as if the unit were operating properly.

(4) The computer hangs up and waits for the unit.

	Power is Off	Standby	Stop	Re-wind	Reverse	Forward	Attached to Print-out Units	Locked in Read	Erase
If Computer Attempts to Record	1, 3	1, 3	1, 3	1, 3	1, 3	1, 3	4	2, 3	4
If Computer Attempts to Read	1, 4	1, 4	1, 4	1, 4	1, 3	1, 3	4	Normal	4

The program itself may sense as to whether the unit is completely ready (power on, attached to the computer, automatic mode, not in erase mode, not locked in read) by examining insertion register 3 which is addressed by an si 303 followed by an rd instruction. If the unit is ready, the intervention bit will be clear; i.e., contain a 0. If unit 2 is not ready, digit 0 of the insertion register will contain a 1; if units 3 or 4 are not ready, digit 1 of the insertion register will contain a 1; if unit 1 is not ready, digit 2 of the insertion register will contain a 1.

3.2.5 Programming for Recording

si m — selects the tape unit and starts the unit in forward or reverse depending on the address m. An interblock space 14.3 milliseconds long (less any time required to accelerate the tape to normal speed) is generated on the tape, then a block mark is automatically recorded (which requires 160 microseconds). The computer cannot perform another in-out instruction until this 14.5 millisecond period has elapsed.

rc-- — records on tape the contents of AC. 2.6 milliseconds must elapse before the computer can perform another in-out instruction. An rc is required for each word to be recorded. As many rc instructions as necessary may be used before the next si instruction. Any number of instructions other than in-out instructions may precede each rc (however, the tape continues to move).

si n — stops the tape unit in cleared area. About 0.6 inches of tape are erased beyond the point at which this order was given; the tape stops, if permitted, about 0.2 inches beyond the point this order was given. This order requires about 28.6 milliseconds.

3.2.6 Programming for Re-Recording

Re-recording is similar to recording except that the tape unit starts in the read mode (which means that a 5.1 millisecond delay is counted in IODC while the unit is starting before the unit starts searching for a block mark); after a block mark is detected, the unit switches to the record mode. No new block mark is recorded - the original block mark is not erased.

The re-record instruction can perform an auxiliary function - that of making possible the skipping of any number of blocks, in either forward or reverse. Each si instruction to re-record causes the tape unit to search for the next block mark and to switch to the record mode as soon as the block mark is found. If, for example, it were desired to replace the nth block on the tape, "n-1" consecutive re-record si instruction can be used. At this point the block mark for the nth block would have just been sensed allowing the new data to be recorded. Since the re-record si instruction immediately switches to the record mode on the sensing of a block mark, these si instructions must be given close enough together to prevent erroneous erasing. (Since the record mode erases previously recorded data). The maximum permissible interval between the si to re-record and the si to switch out of the record mode is dependent on the distribution of data on the tape, but in no case will it be less than 5.1 milliseconds.

3.2.7 Programming for Reading

(1) Word-by-Word Reading

si m — selects the tape unit and starts the unit in forward or reverse depending upon the address of m. After a delay count of 5.1 milliseconds, the computer reads into IOR the first word after the next block mark. The amount of time required for this process will depend on the distance of the next block mark from the reading heads. This si instruction must not be followed by another si without at least one intervening rd or bi instruction.

rd-- transfers the contents of IOR to AC then clears IOR in preparation for receiving the next word from tape. As many successive rd instructions will be needed as there are words to be read from tape. Assuming that the words were recorded at maximum density (one word every 2.6 milliseconds), a pair of digits will be read to IOR at intervals of approximately 326 microseconds. The computer must execute an rd instruction often enough to extract a word from IOR and clear IOR before the first pair of digits of the next recorded word arrives from the tape unit; otherwise a program alarm will result. To stop reading before the end of a recorded block has been reached, give an instruction to deselect the tape unit within about 2.8 milliseconds after the last desired word has been read; otherwise, a program alarm may result. Any instructions other than in-out instructions may precede each rd.

si-- deselects the tape unit. Any si instruction which has been assigned a function will stop the tape unit, but if the program does not require a specific si instruction, use si 630 (o) or si 408 (d).

(2) Reading by Block-Transfer Instruction

A bi instruction may be substituted for a series of rd instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred, and ±n, the number of words to be read, must be stored (times 2^{-15}) in AC at the time the computer executes the bi. Any sequence of rd and bi instructions may follow the si instruction. A program alarm will occur if the unit is not deselected before the first two digits of the second word after the last one read into MCM are transferred into IOR.

No word will be transferred from IOR to AC when a bi instruction calling for the transfer of a block zero words in length is given. A previous si, rd, or bi, may have resulted in a word being transferred from the tape to IOR. If the bi calls for the transfer of no words, this word already read is left in IOR and a program alarm will occur unless another si (which clears IOR), rd or bi (for non-zero length block transfer) occurs before the first two digits of the next word on the magnetic tape are transferred to IOR (a minimum of 320 microseconds between the si, rd or bi which precedes the zero block length bi and the si, rd, or non-zero block length bi which follows it).

3.2.8 Erase System

There are four push buttons located in the center control panel of the magnetic tape equipment which can destroy all information on the magnetic tapes on units 1, 2, 3 or 4. Any unit, except unit 0, may be erased by switching it to printout, by pushing its respective erase button, and by rewinding, going forward, or going in reverse. However, if a unit is locked in read, no erase can be initiated. The erase for any unit may be terminated by again pushing the printout button, by switching the unit to computer, or by switching the unit to lock-in-read. The erase for either unit 3 or unit 4 may also be terminated by switching between units 3 and 4.

3.2.9 Delayed Output via Magnetic Tape

(1) Delayed Output Units

Where printed page or punched paper tape output is desired, computer time can be conserved by the use of the delayed output units. The binary characters which control the printer or the punch can be recorded on certain magnetic tape units by the computer, and these tapes later run through the delayed output units which read the characters, select the printer or the punch automatically, and then print or punch these characters. Flexowriter characters for delayed output can be recorded at the rate of 133 per second. A 1000-foot reel of magnetic tape can store over 50,000 characters which can be recorded in a minimum of 6.7 minutes, and which can be printed in about 90 minutes or punched in about 63 minutes.

Four of the magnetic tape units - 1, 2, 3, and 4 - are also associated with the delayed output equipment. A toggle switch located above tape unit 2 allows units 3 or 4 to be selected for computer output even though the program selects unit 2; or allows unit 2 to be selected even though the program has selected unit 3 or 4. The function of the units (computer or printout) is controlled by a set of switches and buttons located above each unit which connect the tape unit either to printout systems A, B or C, or to the computer. Similar switches are located near the delayed printout typewriters.

(2) Programming for Delayed Output

In order to record on magnetic tape a series of Flexowriter characters for later automatic printing or punching, the following conventions must be observed:

(a) A full 16-digit word is recorded on magnetic tape to store each 6 binary digit Flexowriter character and two control digits. The six binary digits corresponding to the character to be printed or punched occupy digit positions 0 through 5 of AC when the word is recorded. Digit position 7 is used to determine whether the output is to be printed or punched; a 1 in digit 7 selects the punch, a 0 selects the printer. When the punch is selected, digit position 6 determines whether or not the 7th hole is to be punched on that line of paper tape; if digit 6 is a 1, the 7th hole is punched - if a 0, it is not punched. This control information must accompany each character recorded for delayed output; neither the printer nor the punch remains selected after a character is printed or punched.

(b) The separation between characters recorded on the magnetic tape for delayed output should be not less than 7.5 milliseconds (between the same points of two successive words) when the tape is running at normal speed. The programmer may count the necessary delay (4.9 milliseconds) between words using the same si orders and programming given previously for storage purposes. However, this minimum delay can be obtained automatically by using the special si orders for"recording for delayed output" given at the end of this section. When these si orders are given, the same interblock delay (14.3 milliseconds) is counted, and a block mark recorded at the end of this period; but on each succeeding rc instruction, a 5.1 millisecond delay is counted before the word is recorded. Since the recording of the word requires 2.6 milliseconds, approximately the minimum spacing between words is obtained on the tape. Thus the programmer can use the program outlined in Section 3.2.5, substituting only the correct si instructions, to record for delayed printing. Only if the non in-out orders intervening between successive rc orders require more than 7.5 milliseconds will any unnecessary magnetic tape be used.

(c) It is advisable but not necessary to provide a Flexowriter stop character as the last character recorded so that the Flexowriter, and therefore, the tape unit, will stop after this character is read from the tape. Then the delayed output equipment may operate unattended.

3.2.10 si Addresses for Magnetic Tape Units

(1) si Addresses for Direct Use of Magnetic Tape

si Instruction	0	1	2	3 or 4*
Re-record forward	si 100 (o) si 64 (d)	si 110 (o) si 72 (d)	si 120 (o) si 80 (d)	si 130 (o) si 88 (d)
Re-record reverse	si 101 (o) si 65 (d)	si 111 (o) si 73 (d)	si 121 (o) si 81 (d)	si 131 (o) si 89 (d)
Read forward	si 102 (o) si 66 (d)	si 112 (o) si 74 (d)	si 122 (o) si 82 (d)	si 132 (o) si 90 (d)
Read reverse	si 103 (o) si 67 (d)	si 113 (o) si 75 (d)	si 123 (o) si 83 (d)	si 133 (o) si 91 (d)
Stop after record in cleared area forward	si 104 (o) si 68 (d)	si 114 (o) si 76 (d)	si 124 (o) si 84 (d)	si 134 (o) si 92 (d)
Stop after record in cleared area reverse	si 105 (o) si 69 (d)	si 115 (o) si 77 (d)	si 125 (o) si 85 (d)	si 135 (o) si 93 (d)
Record forward	si 106 (o) si 70 (d)	si 116 (o) si 78 (d)	si 126 (o) si 86 (d)	si 136 (o) si 94 (d)
Record reverse	si 107 (o) si 71 (d)	si 117 (o) si 79 (d)	si 127 (o) si 87 (d)	si 137 (o) si 95 (d)

(2) si Addresses for Delayed Printing Via Magnetic Tape

The si addresses used to obtain the correct spacing between characters recorded on the magnetic tape are as follows:

si Instruction	0 **	1	2	3 or 4*
Record forward for delayed printout	si 146 (o) si 102 (d)	si 156 (o) si 110 (d)	si 166 (o) si 118 (d)	si 176 (o) si 126 (d)
Record reverse for delayed printout	si 147 (o) si 103 (d)	si 157 (o) si 111 (d)	si 167 (o) si 119 (d)	si 177 (o) si 127 (d)

---

* These si-addresses refer to whichever unit is connected to the computer through the transfer switch.

** Unit 0 is not available for delayed printout.

3.3 PHOTOELECTRIC TAPE READER

The Ferranti photoelectric tape reader, abbreviated PETR, reads the seven-hole paper tape punched by the Flexowriter. The maximum reading speed of the PETR is between 190 and 220 lines per second. If the motor is OFF when PETR is selected by the computer, approximately 15 seconds is necessary for the reader to attain its maximum speed. After the PETR has been deselected, its motor continues to run for from 30 to 45 seconds. If PETR is reselected within this time, the reader is at full speed in 2 or 3 milliseconds. PETR is a free-running unit; that is, once selected, it continues to run until deselected by an si instruction.

There are two PETR's available in test control -- PETR A and PETR B. Either may be placed in ready condition by pushing its respective button, the Select Reader A button or the Select Reader B button; the other reader will be in a standby condition until its selection button is pushed. Pushing the STOP button while the PETR is selected will stop the tape but will not deselect the PETR.

3.3.1 Punched Paper Tape

The conventional forms of paper tape are: (1) flexowriter coded input; and (2) standard form of a 16-digit word on punched paper tape known as the "5-5-6" form. In this "5-5-6" form, the binary digits (numbered 0 through 15) are physically distributed on the tape as shown in Figure 1; where a hole in a digit position indicates that digit is a 1, no hole indicates a 0.

Hole No.	(1)	(2)	(3)		(4)	(5)	(6)	(7)
	0	1	2	.	3	4	X	•
One Word	5	6	7	.	8	9	Y	•
	10	11	12	.	13	14	15	•
				Feed Holes				

Figure 1

Positions X and Y are normally unpunched to aid in visually reading the tape. However, they may contain the same information as positions 5 and 10, respectively. The word-by-word reading modes of the in-out system are devised to correctly

assemble into a 16-digit word a word punched in 5-5-6 form. Each line of tape which contains information must have the 7th hole position punched. If this were not done, the tape reader could not distinguish a line of significant zeros (which it must read) from a line of blank tape (which it must ignore). The omission of the 7th hole then allows the feature of punched visual identification numbers which will be ignored by the reader. Holes 1 - 5 of the first line correspond to bits 0 - 4 of the computer word; holes 1 - 5 of the second line correspond to bits 5 - 9 of the computer word; and holes 1 - 6 of the third line correspond to bits 10 - 15 of the computer word.

3.3.2 Action of the Photoelectric Reader

The Ferranti photoelectric tape reader "reads" the 6-digit binary combination punched in a line of paper tape and transmits it to the right-hand six digit places of IOR. In the line-by-line mode, each reading operation reads one line of tape and forms a word of which the left-hand ten digits are zeros and the right-hand six digits correspond to the binary combination punched in the tape. In the word-by-word mode, each reading operation reads three lines of tape and assembles (by successive shifts of five to the left in IOR) a 16-digit word from the digits punched in the tape in 5-5-6 form.

3.3.3 Programming for Line-by-Line Mode

(1) Reading in the Line-by-Line Mode

si r starts PETR. Reads the first 6-digit character from tape into the right-hand six digits of IOR and places zeros in digit positions 0 - 9 of IOR. This si instruction must not be followed by another si without at least one intervening rd or bi instruction.

rd-- waits until next character arrives in IOR and then transfers contents of IOR to AC and clears IOR in preparation for receiving the next character. The contents of digits 0 - 9 of AC will be zeros, and the contents of digits 10 - 15 of AC will correspond to the binary combination read from tape. As many successive rd instructions are necessary as there are lines of tape to be read. If there are no intervening lines of blank tape, a 6-digit character will arrive at IOR every 4.5 to 5.0 milliseconds. The computer must execute an rd instruction often enough to extract a word from IOR and clear IOR before the next character arrives from the reader, otherwise a program alarm will result. On the average this means that an rd instruction must be executed oftener than once every 4.5 milliseconds or, to allow a safety margin, no more than about 120 - 130 orders between each rd. Any instructions other than in-out instructions may precede each rd.

si-- stops the reader. If PETR is deselected soon enough after a particular line is read, it can decelerate to a stop and still pick up the next line on the paper tape when the programmer again gives an si for reading from PETR. The maximum safe time between the rd instruction which reads the last desired line before the stop, and the si instruction which deselects PETR, is about 2 milliseconds which insures

that the next line on the tape will not be skipped when PETR is used again. Any si instruction which has been assigned a function other than selecting the reader will deselect and stop the reader, but if the program does not require a specific si instruction, use the conventional designation si 630 (octal) or si 408 (decimal).

(2) Normal Line-by-Line Read

In the normal line-by-line reading operation, one line of information is read in on each in-out operation. The unit comes to a stop after each read.

(3) Reading Line-by-Line by Block-Transfer Instruction

A bi instruction may take the place of a series of rd and ts instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred, and +n, the number of words to be read, must be stored (times 2^{-15}) in AC. The time required for the block transfer is the same as the total time required to perform the rd instruction it replaces. If AC contains +0, a bi will delay computer until a word arrives in IOR. Normally an si should be given after each bi and before any following bi or rd. If an si is not given, the next word transferred from IOR by an rd or bi will be +0 and not the next word on the tape. Any sequence of rd and bi instructions may follow a single si.

3.3.4 Programming for Word-by-Word Mode

(1) Reading in the Word-by-Word Mode

si r starts PETR. Reads the next three lines of tape (which must be punched in the 5-5-6 form) and assembles them into a 16-digit word in IOR. This si instruction must not be followed by another si without at least one intervening rd or bi instruction.

rd-- transfers contents of IOR to AC then clears IOR in preparation for receiving the next character. The contents of AC will correspond to the 16-digit word originally punched on tape. As many successive rd instructions are necessary as there are words to be read from tape. If there are no intervening lines of blank tape, a 6-digit character will arrive in IOR every 4.5 to 5.0 milliseconds (13.5 to 15 milliseconds per word). The computer must execute an rd instruction often enough to extract a word from IOR and clear IOR before the next character arrives from the reader, otherwise a program alarm will result. Any instructions other than in-out instructions may precede each rd.

si-- stops the reader. Any si instruction which has been assigned a function other than selecting the reader will deselect and stop the reader, but if the program does not require a specific si instruction, use the conventional designation si 630 (octal) or si 408 (decimal). As noted above, if PETR is not deselected in blank tape, the si for deselecting PETR should follow the last rd instruction within 2 milliseconds in order not to skip a line on the tape.

(2) Reading Word-by-Word by Block-Transfer Instruction

A bi instruction may take the place of a series of rd instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred, and ±n, the number of words to be read, must be stored (times 2^{-15}) in AC. The time required for the block transfer is the same as the total time required to perform the rd instructions it replaces. Any sequence of rd and bi instructions may follow a single si.

(a) Zero-Length Block Transfer on bi

The use of a bi instruction calling for the transfer of a block zero words in length will result in one word being read but not transferred. The reading of the word actually is initiated by the preceding si (or bi or rd) instruction, hence one word is already read by the time the bi is ready to be performed. If the bi calls for the transfer of no words, the word already read is simply discarded.

3.3.5 si Addresses for the Photoelectric Reader

		PETR
read line-by-line:	si 210 (o) or si 136 (d)	A
	si 211 (o) or si 137 (d)	B
read word-by-word:	si 212 (o) or si 138 (d)	A
	si 213 (o) or si 139 (d)	B

3.4 MECHANICAL TAPE READER

The mechanical tape reader is the section of the electrically operated Flexowriter which senses the 6-digit binary combination punched in a horizontal line of paper tape and then transmits the information to the right-hand 6 digit places of IOR. Since the mechanical tape reader has as its average operation speed only 3-1/3 words per second, or only 10 lines per second, it is very seldom used.

In the normal line-by-line mode, each reading operation on one line of tape forms a word of which the left-hand ten digits are zero and the right-hand six digits correspond to the binary information punched in the line of tape.

In the automatic assembly word-by-word mode, each reading operation on three lines of tape assembles (by successive shifts left in IOR) a 16-digit word from the digits punched in the tape in 5-5-6 form. (See Section 3.4.2 (1).) The mechanical tape reader does not need to be stopped by an si instruction.

3.4.1 Programming for Line-by-Line Mode

(1) Reading in the Line-by-Line Mode

si r selects the mechanical reader.

rd-- reads the next 6-digit character from paper tape into the right-hand six digit positions of AC from the IOR and clears IOR in preparation for receiving the next character. The contents of digits 0 - 9 of AC will be zeros, and the contents of digits 10 - 15 of AC will correspond to the binary combination read from tape. In this mode with the mechanical reader, the computer requires about 105 milliseconds to execute each rd instruction. As many successive rd instructions are necessary as there are lines of tape to be read. Any number of instructions other than in-out instructions may precede each rd. Since the reader automatically stops after each rd instruction, no special command is necessary to deselect the mechanical tape reader.

(2) Reading Line-by-Line by the Block-Transfer Instruction

A bi instruction may take the place of a series of rd instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred, and +n, the number of lines to be read, must be stored (times 2^{-15}) in AC. The time required to execute the block transfer is the same as the total time required to perform the rd instructions it replaces; however, the computer cannot be used for other operations when a bi instruction is given. Any sequence of rd and bi instructions may follow a single si.

3.4.2 Programming for Word-by-Word Mode

(1) Reading in the Automatic Assembly Word-by-Word

si r selects the mechanical reader.

rd-- reads the next three lines of tape (which must be punched in 5-5-6 form) and assembles them via IOR into a 16-digit word in AC and clears IOR in preparation for receiving the next word. In this mode with the mechanical reader, the computer requires about 315 milliseconds to execute each rd instruction. As many successive rd instructions are necessary as there are words to be read from tape. Any number of instructions other than in-out instructions may precede each rd.

(2) Reading Word-by-Word by the Block-Transfer Instruction

A bi instruction may take the place of a series of rd instructions. The address of the bi must be the initial address of the block of registers in MCM to which the words will be transferred and +n, the number of words to be read, must be stored (times 2^{-15}) in AC. The time required to execute the block transfer is the same as the total time required to perform the rd instructions it replaces. Any sequence of rd and bi instructions may follow a single si.

(a) Zero-Length Block Transfer on bi

Zero-length block transfers should not be used with the mechanical tape reader in either mode of operation. Such an order (bi with +0 in AC) actually reads a line or word from the paper tape which will be lost. Program alarms may also occur if zero-length block transfers are attempted with the mechanical reader.

3.4.3 si Addresses for the Mechanical Reader

The si addresses for the mechanical tape reader are as follows:

read line-by-line:	si 200 (o)	or	si 128 (d)	Console
	si 201 (o)	or	si 129 (d)	
read word-by-word:	si 202 (o)	or	si 130 (d)	Console
	si 203 (o)	or	si 131 (d)	

3.5 PRINTER

The direct Flexoprinter is an electrically operated typewriter which prints characters in response to a 6-digit binary coded input called the Flexowriter code, received directly from the computer. There are two such printers available: printer 2 in the computer room in use next to the console, and printer 3 originally located in the direction center but not in use at the present time.

3.5.1 Action of the Direct Printer

Each character to be printed or machine function to be performed (for example, carriage return) requires that the computer send to the printer a 6-digit binary character from the left-hand six digit places of AC. Each key on the printer is actuated by a unique code character. The printer utilizes only 51 of the 64 possible code combinations and it will ignore without consequence the remaining combinations. The computer-controlled printers will also ignore the "stop" code.

(1) The Flexowriter Code

The 6-digit code, known as the "FL" Flexowriter Code, is assigned arbitrarily by the manufacturer. The code is given in tables at the end of this memorandum. Table 1 explains reading of the tape, Table 2 is in alphanumerical sequence and Table 3 is in numerical sequence of binary code characters.

3.5.2 Programming for Printer Operation

(1) The following sequence of instructions causes the printing of alphanumerical characters and the performance of machine functions (i.e., tab, carriage return, etc.):

si t — selects the printer designated by the address t. The printer will remain selected until the next si instruction is executed.

rc-- — actuates the printer key corresponding to the 6-digit code character contained in digits 0 - 5 of AC. A time (listed below) equal to that required for the printer to respond to the most recent character must elapse before the computer can perform the next in-out instruction. An rc instruction is required for each character to be printed or machine function to be performed. As many rc instructions as necessary may be used before the next si instruction. Any number of instructions other than in-out instructions may precede each rc.

(2) Printing via the Block-Transfer Instruction

If the Flexowriter codes for a group of characters to be printed are stored in sequence and in the left-hand 6-digit places in a block of consecutive registers, a bo instruction may be substituted for a series of rc instructions. The address of the bo must be the initial address of the block of registers, and +n, the number of registers in the block, must be stored (times 2^{-15}) in AC at the time the bo instruction is executed. The time required for the block transfer to the printer will be the same as the total time required to execute the rc instructions it replaces. Any sequence of rc and bo instructions may follow a single si.

3.5.3 Printer Response Times

The approximate times required for the printer to carry out various processes are listed below:

Print any alphanumerical character or symbol, space, color change, upper and lower case shifts - 125 milliseconds

Back space - 180 milliseconds

Tabulation and carriage return - - - - - - - - 200 to 900 milliseconds

3.5.4 Printer for Delayed Output

To conserve computer time, information can be recorded on magnetic tape and later printed out. (See Section 3.2.9.)

3.5.5 si Address for Printer

(1) The following instructions will select the printer indicated and operate properly on an rc provided that digits 0 - 5 of the accumulator do not all contain "zero." If AC 0 - 5 all contain "zero," a completion pulse will never be received from the printer and the computer will hang up on an in-out instruction.

si 224 (o) or si 148 (d) - select printer #2 in test control by console.

si 234 (o) or si 156 (d) - select printer #3.

(2) The following instructions will select the printer indicated. The action on an rc will be the same as described in section (1) above except that a printer completion pulse will always be received.

si 225 (o) or si 149 (d) - select printer #2 in test control.

si 235 (o) or si 157 (d) - select printer #3.

(3) The following orders will select the printer indicated, but three characters will be printed on one rc instruction. The code of these characters will depend on the contents of the AC. If AC contains ABCDEFGHIJKLMNOP, the printer will attempt to print the three characters whose codes are ABCDEF, FGHIJK, and KLMNOP. If one of these groups of six contains all zeros, the computer will not receive a printer completion pulse and will hang up.

si 226 (o) or si 150 (d) - select printer #2 in test control.

si 236 (o) or si 158 (d) - select printer #3.

(4) The following orders will select the printer indicated. The action on an rc will be the same as described in section (3) above except that a printer completion pulse will always be received.

si 227 (o) or si 151 (d) - select printer #2 in test control.

si 237 (o) or si 159 (d) - select printer #3.

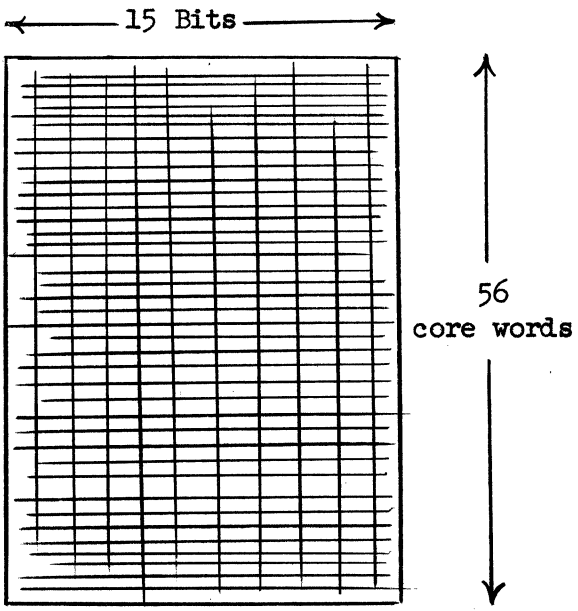
3.6 ANELEX PRINTER

3.6.1 General

The use of the Anelex Line Printer requires considerably less computer operating time than is required for either delayed printout via magnetic tape or direct Flexowriter printer. The printer offers a choice of 56 different characters with a maximum of 120 characters across the page. (See Figure 2.) The characters are spaced nominally ten-to-the-inch across the page and six-to-the-inch in the vertical direction for single spacing.

Unlike the Flexowriter which has a mechanical translator to decode a six-bit character, the Anelex printer requires the decoded information for one complete line, including blanks, to be available for each column of the maximum of 120 to be printed. This information must be broken down into eight groups of 15 columns each (see Figure 3) where only one group may be done at a single interval of time. Since only 15 columns are printed at a time, this machine is not a true "line" printer. The printer requires one revolution of the print wheel to pass all 56 characters through the printing position; therefore, eight revolutions are required to print 120 columns. With a printroll speed of 1200 RPM, and with the allowance of one revolution for advancing the paper, the maximum speed for the printing of 120 columns is 133 lines per minute. If fewer groups of columns are used, the speed can be increased to a maximum of 600 lines per minute for a 15-column printout. In order to print a full line of information with one block out instruction, 8 x 56 or 448 (d) consecutive registers are needed.

The computer core memory is used to form a "printout image." Within this image, a group of registers is allocated to each 15-column Anelex group. For such a group of 15 columns a block of 56 (d) core registers should be considered as a 56 x 15 matrix.



ORDER OF CHARACTERS ON PRINT WHEEL

1.	slant	29.	N
2.	regular minus	30.	delta Δ
3.	regular decimal or period	31.	P
4.	plus	32.	Q
5.	comma	33.	R
6.	exponent 0	34.	S
7.	exponent 1	35.	T
8.	exponent 2	36.	U
9.	exponent 3	37.	V
10.	exponent 4	38.	W
11.	exponent 5	39.	X
12.	exponent 6	40.	Y
13.	exponent 7	41.	Z
14.	exponent 8	42.	figure 0 and letter 0
15.	exponent 9	43.	figure 1 and letter I
16.	A	44.	figure 2
17.	B	45.	figure 3
18.	C	46.	figure 4
19.	D	47.	figure 5
20.	E	48.	figure 6
21.	F	49.	figure 7
22.	G	50.	figure 8
23.	H	51.	figure 9
24.	sigma δ	52.	left parenthesis
25.	J	53.	right parenthesis
26.	K	54.	exponent minus
27.	L	55.	inequality
28.	M	56.	exponent decimal

Figure 2

ANELEX GROUP BREAKDOWN

GROUP	COLUMNS
0	1 - 15
1	16 - 30
2	31 - 45
3	46 - 60
4	61 - 75
5	76 - 90
6	91 - 105
7	106 - 120

Figure 3

The presence of a "one" bit in the ith column and jth row would instruct the printer to print the jth character from the list given in Figure 2 in the ith column relative to the beginning of the currently selected 15-column Anelex group. (The 15 bits are taken from bits 1 through 15 of the core memory words; the sign bits, bit 0, are used for control purposes.) Such a group of 56 (d) consecutive registers is required for each group of 15 columns to be printed. In addition to blocks of 56 (d) words per group, additional single words are required for control purposes; i.e., an Anelex image normally consists of 56 word blocks interspersed with appropriate control words.

An si 244 (o) will select the printer and will not affect the group control; an si 245 (o) will select the printer and clear the group control; the subsequent bo instruction will block out via the input-output register the desired number of registers of the image.

Indexing of the paper is ordinarily under the control of the programmer; however, certain automatic operations, such as the single-space feed after a bo instruction to group 7 or the page feed at the end of a page, cannot be inhibited.

3.6.2 Control Registers

If the sign bit of a register in the core image is positive (contains a "zero"), that register specifies information to be printed; if the sign bit is negative, the information in that register is used for control purposes and is not printed. The programmer may control that group (0 - 7) to which the next 56 registers refer, and he may also call for a paper feed. Either or both of these functions may be controlled by the same control register. A control register may appear anywhere within the core image block; (but the next non-control register will correspond with the first character on the print wheel). However, control registers normally appear in between blocks of 56 (d) words.

If the first word of an image block is a control word, the computer will have to wait to get in synchronization with the printroll a second time. Thus a 50 millisecond delay of one printroll revolution will result if group control is changed and/or if a paper feed of less than five lines is initiated. If the paper has to advance more than five lines, a further delay will result. To conserve computer time, whenever possible the last word of a block should be a control word to specify paper feeds and group control changes, since in this case the computer proceeds immediately after initiating the control word functions.

(1) Group Selection

If the sign bit (digit 0) and bit 2 contain a "one," the anelex printer group control is set to whatever number (0 - 7) appears in bits 13 - 15 of the same register. If the programmer desires to start his printout in a group other than the group presently selected, the first word in the image should be a control word to select the desired group.

(2) Paper Feed

If bit 0 and bit 1 contain a "one," the paper will be indexed according to which format is specified in bits 7 - 12 of the control word which correspond

to formats 1 - 6, respectively. Only one bit from 7 - 12 should contain a "one." (If none of these six bits contain a "one," the computer will hang up within the block order, unless the paper feed is called for on the last word of the block. In this case, the computer would hang up when the next printer bo instruction is given.)

The format control consists of a pre-punched tape with eight channels which is mounted within the printer itself. Formats 1 - 7, of which 1 - 6 are available to the programmer, stop the paper feed when a hole is sensed by a photocell. Format 1 (digit 7) provides a single-space paper feed; format 2 (digit 8) provides double spacing; format 6 (digit 12) indexes the paper to the beginning of the next page. Format 3 is available to provide triple-space paper feed. Also, the format control automatically provides indexing from the bottom of one page to the top of the next page.

A single-space paper feed takes approximately 17 milliseconds, and each additional space (6-to-the-inch) takes 7 milliseconds if programmed by a single control register. Each separate paper feed takes a minimum of 17 milliseconds. The only time the computer does not hang up and wait for completion of a paper feed is when the last word of an image is a control word which calls for a paper feed, or when the group control is indexed from 7 to 0 at the end of a block order. Thus, to save on computer time, a paper feed should be called for at the end of a block rather than at the beginning of the next block.

(3) Combined Paper Feed and Group Selection

If bits 0, 1 and 2 contain a "one," a paper feed is initiated and group control is changed (as indicated above). The next non-control register corresponds to the first character on the printroll.

(4) Termination of Anelex Block Out Order (bo)

Another control word function of the Anelex is to allow programmers to terminate the Anelex block out order (bo) when the control word is read, even though the block length indicates that there are more words to be read. A "one" in digit 0 and a "one" in digit 3 selects this mode. The other two control functions (group change and paper feed) may be specified in the same control word as the terminate bo function.

3.6.3 Programming

When an si 245 is given, the printer group control is cleared so that the first 70 (o) registers to be blocked out after the bo is given cause printing in group 0, columns 1 - 15. After a set of 70 (o) registers is blocked out, the group control is automatically indexed to the next group. The next register to be blocked out then corresponds to the first character on the print-wheel and specifies printing in the new group. For example, the following instructions start a printout:

```
si  245
ca  RC (block length)
bo  (starting address)
```

After the specified block length is blocked out, unless terminated sooner by a special control word, the computer exits from the bo instruction and goes on to the next instruction. At the end of the bo instruction the group control is indexed, unless the last register of the image is a control register. If group control is on 7, it will go to group 0. Also, when group control is automatically indexed from 7 to 0, a single-space line feed is initiated.

In general, the block length for a printout will be some multiple of 70 (o) plus the number of control registers used within the block. To print one full line, 10 x 70 or 700 (o) registers plus one paper feed control register at the end of the block, making a total of 701 (o) registers, might be used. (However, if 700 (o) registers starting in group Ø have been blocked out, after an si 245 (o), no paper feed control word is needed since an automatic paper feed occurs.) If the information for several lines of printing is available, the block length can be increased accordingly. At the end of group 7, if there is no control word, the paper will feed one line and group control will select group 0. If, however, a control word is given, then the automatic paper feed and group index are suppressed; the control word must specify paper feed and new group if they are desired.

Although common, it is not necessary that a block length of 70 (o) be used to print each group of columns. It must be only as long as the position on the print-wheel of the last character wanted in the printout (see character ordering in Figure 2). If several groups of the same block length were printed out in this manner, a control register to specify a new group would have to be inserted wherever the information for a given group ended.

If the last word of a previous block initiated a paper feed, the computer does not wait for the paper to be advanced but is available for programming. If the line feed command occurs immediately after decimal code 56 time, 50 milliseconds must elapse before printing can take place. To operate the printer at maximum speed, the new information must thus be prepared and another printer bo given within the 50 milliseconds.

Example:

Figure 4 shows the printout image to print the words PRINTOUT IMAGE: in group 2 (columns 31 - 45). Assume that a block of registers starting at 100 (o) are available for this block. After the block is loaded, the following octal program should be used:

```
si 244
ca RC (72)
bo 100
```

The first word of the block is a control word that selects group 2. The following 70 (o) registers correspond to the characters on the printroll. Note that in none of the registers 101 - 171 (o) is a "one" in bit 9, so that this column will be blank. Another point of interest is that bit 15 contains a "one" in register 103 (o) (regular decimal point or period) as well as in

SAMPLE PRINTOUT

Register	Character	Decimal Code #	Bits 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
100			1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
101	/	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
102	-	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103	. (reg.)	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
104	+	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
105	,	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
106	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
107	1	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110	2	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111	3	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
112	4	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
113	5	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
114	6	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
115	7	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
116	8	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
117	9	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
120	A	16	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
121	B	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
122	C	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
123	D	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
124	E	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
125	F	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126	G	22	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
127	H	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
130	I	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
131	J	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
132	K	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
133	L	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
134	M	28	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
135	N	29	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
136	Δ	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
137	P	31	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
140	Q	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
141	R	33	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
142	S	34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
143	T	35	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
144	U	36	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
145	V	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
146	W	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
147	X	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	Y	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
151	Z	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
152	0	42	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
153	1	43	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
154	2	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
155	3	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
156	4	46	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
157	5	47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
160	6	48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
161	7	49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
162	8	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
163	9	51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
164	(	52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
165	)	53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
166	- (exp.)	54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
167	>	55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
170	. (exp.)	56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
171			1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
				P	R	I	N	T	O	U	T		I	M	A	G	E	:

Figure 4

register 170 (o) (exponent decimal). Both of these characters are printed, resulting in a colon. (See Section 3.6.5 for more information on this "overprinting.") Register 171 (o) is a control register which calls for a single-space paper feed but no group change. The computer will exit from the bo instruction and continue with the rest of the program while the paper is indexing.

3.6.4 Core Clearing

To prepare a block of registers for a printer image, the block might be cleared and then the printer image generated. With a simple loop program, a block of 700 (o) registers may be cleared in about 40 milliseconds, which, for maximum printer speed, would leave less than 10 milliseconds to load the block with the printer image. To save computer time, the special method of clearing cores with si 17 (o) should be used to read +0 into a block of registers. This process takes approximately 5 milliseconds for a block of 700 (o) registers. (See Section 4.7 for further information.)

3.6.5 Overprinting - Special Characters

Because of the limited number of characters (56 (d)) available on the printer, provisions have been made to form special characters by overprinting two characters on the same line and column. One way to do this is to use a second set of 70 (o) registers, referring to the same group and line as the first block of registers, to print the second part of the special characters. The disadvantages of this method are the extra time (50 milliseconds) and the extra storage (70 (o) registers) required.

It is also possible to print two characters in the same printroll cycle by placing a "one" in the two registers in the block of 70 (o) registers corresponding to the two characters which are to be overprinted (e.g., as shown in Figure 4 to produce a colon). The only restriction is that the two characters be separated by at least 28 (d) positions on the printroll.

Figure 5 shows the most common special characters that can be obtained by overprinting.

SPECIAL CHARACTERS OBTAINED BY OVERPRINTING COMBINATIONS

$\emptyset$	(using slant and 0)
$\ominus$	(using minus and 0)
$\leftarrow$	(using minus and parenthesis)
=	(using both minus signs)
:	(using both decimal points)
;	(using exponent decimal and comma)

Figure 5

3.7 PUNCH

The punch is a device physically connected to the Flexowriter and is used to transfer binary information from the computer to paper tape. It will punch any combination of seven holes with the presence of a hole being a "one" and the absence of a hole, a "zero." This punched tape has a series of small holes, "feed holes," located near the center of the tape which are used to locate a line on the tape. The holes on the tape are normally referred to as holes 1 - 7 starting at the left side, with the feed hole located between holes 3 and 4.

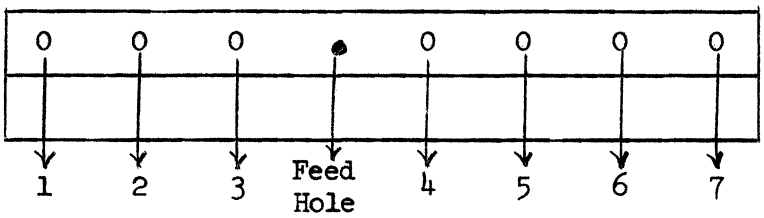


Figure 6

The "seventh hole" position indicates whether or not the line contains information pertinent to the computer. No hole in this position indicates that there is no pertinent information in that line. The other six positions transfer 6 binary digits of information into the computer.

Each line of digits to be punched on tape is transmitted to the punch from the left-hand 6 digit places of the IOR. The seventh digit is controlled by IOS and will punch or suppress the seventh hole, depending on the mode of operation of the punch selected by the si instruction. The punch requires 93 milliseconds to transmit one line of information to paper tape. In the word-by-word mode, each recording operation punches three lines of tape in 5-5-6 form (see Section 3.7.2), by successive shifts left in IOR, corresponding to the 16-digit word in the IOR.

3.7.1 Programming for Line-by-Line Mode

(1) Punching in the Line-by-Line Mode

si p selects the punch and prepares to punch or suppress the seventh hole, according to the address p, where p = 204 (o) or 205 (o). The punch will remain selected until the next si instruction is executed.

rc-- punches in one line on paper tape the 6-digit binary combination corresponding to the contents of digits 0 - 5 of AC. The seventh hole position is automatically punched, or not, according to the mode determined by the most recent si instruction. About 80 milliseconds must elapse before the computer can perform the next in-out instruction. An rc is required for each line of tape to be punched. As many rc instructions as necessary may be used before the next si instruction. Any number of instructions other than in-out instructions may precede each rc.

(2) Punching Line-by-Line by Block-Transfer Instruction

If the characters to be punched are stored in sequence and in the left-hand 6 digit places in a block of consecutive storage registers, a bo instruction may be substituted for a series of rc instructions. The address of the bo must be the initial address of the block of registers and +n, the number of registers in the block, must be stored (times 2^{-15}) in AC at the time the bo instruction is executed. The time required for the block transfer to the punch will be the same as the total time required to execute the rc instructions it replaces. Any sequence of rc and bo instructions may follow a single si.

3.7.2 Programming for Word-by-Word Mode

(1) Punching in the Word-by-Word Mode

si p selects the punch and prepares to punch or suppress the seventh hole, according to the address p, where p = 206 (o) or 207 (o). The punch will remain selected until the next si instruction is executed.

rc-- punches in 5-5-6 form (in three lines) the 16-digit binary combination corresponding to the contents of AC. The seventh hole position is automatically punched, or not, according to the mode determined by the most recent si instruction. About 240 milliseconds must elapse before the computer can perform the next in-out instruction. An rc is required for each word to be punched in three lines on tape. As many rc instructions as necessary may be used before the next si instruction. Any number of instructions other than in-out instructions may precede each rc.

(2) Punching Word-by-Word by Block-Transfer Instruction

If the words to be punched are stored in sequence in a block of consecutive storage registers, a bo instruction may be substituted for a series of rc instructions. The address of the bo must be the initial address of the block of registers, and +n, the number of registers in the block, must be stored (times 2^{-15}) in AC at the time the bo instruction is executed. The time required for the block transfer to the punch will be the same as the total time required to execute the rc instructions it replaces. Any sequence of rc and bo instructions may follow a single si.

3.7.3 Punch for Delayed Output

To conserve computer time, information can be recorded on magnetic tape and later punched out. (See Section 3.2.9.)

3.7.4 si Addresses for Punch

line-by-line normal:	si 205 (o) or si 133 (d)
line-by-line, 7th hole suppressed:	si 204 (o) or si 132 (d)
word-by-word normal:	si 207 (o) or si 135 (d)
word-by-word, 7th hole suppressed:	si 206 (o) or si 134 (d)

3.8 OSCILLOSCOPES

The display oscilloscope (scope) is basically a cathode ray tube which receives its horizontal and vertical deflection voltages and intensification signals from the computer. The display system consists of three types of controls: a selection control which chooses the scopes to display as a medium of output; a deflection control which determines the position of the display on the scope face; and an intensification control which indicates the light intensification for the selected scopes at the proper times in the display cycle.

In test control these are two 16" magnetically deflected and focused scopes for programmers' use, and one 5" electrostatically deflected and focused scope for computer maintenance purposes. One of the 16" scopes is primarily intended for photographic purposes and has a Fairchild 35mm automatic scope camera attached. These two 16" scopes have a common switch (ALL DISPLAY SWITCH) which selects all 256 categories, a category being a selected line of intensification combined with a selected mode of operation. In the direction center there are thirty 16" scopes and nineteen 5" scopes also available for programmers' use.

3.8.1 Selection of Scope Displays

The addresses of the si instructions for displays on the oscilloscopes, tabulated in Section 3.8.3 (6), specify a particular "scope intensification line" and mode of operation (point, character, or vector display). Any scope connected to the selected line will display a point or character or vector on each succeeding display instruction until a different si instruction is given. A bank of toggle switches at each scope permits the connection of that scope to one or several, but not all, of the 256 scope lines. A toggle switch on the manual intervention panel in test control connects the two 16" scopes in test control to all of the scope lines when it is on "up" and disconnects them from all lines when it is off. Of the scopes in the direction center there are two 16" scopes, the Visitors' Scopes, which by means of a dial can be connected to any scope in the direction center and can display whatever is on this scope.

3.8.2 Scope Deflection

The left-hand 11 digits of AC (including the sign digit), at the time a display instruction is given, determine the direction and amount of deflection. The positive direction of horizontal deflection is to the right and positive vertical deflection is upward. The value $1 - 2^{-10}$ or its negative will produce the maximum deflection. The center of the scope represents the origin with zero horizontal and vertical deflections.

3.8.3 Types of Displays

(1) Display of a Single Point

The display of a single point is programmed by the following instructions:

si s — selects the scope intensification line designated by the address s. Sets the vertical deflection of all scopes to a value corresponding to the contents of digits 0 - 10 of AC.

rc-- sets the horizontal deflection of all scopes to a value corresponding to the contents of digits 0 - 10 of AC. Intensifies a point on all scopes which are connected to the intensification line selected by the above si instruction. About 170 microseconds will elapse before the computer can perform the next in-out instruction. Any number of instructions other than in-out instructions may precede each rc. Each point to be displayed is programmed in a similar manner. The address of this rc instruction is immaterial.

(2) Display of Horizontal Lines (Point-by-Point)

The vertical deflection is set up by any si instruction (including those which do not refer to scopes) and remains unchanged until a new si instruction is executed. Similarly, the horizontal deflection is set up by any rc instruction (while a scope line is selected) and remains unchanged until a new rc instruction is executed. Hence a horizontal line may be displayed simply by a single si to set the vertical deflection, followed by a succession of rc instructions to set up the horizontal deflections, and display the individual points on the horizontal line. A cycle can be set up using the rc instruction where the number of times one goes through the cycle corresponds to the number of spots desired on the line. Before each successive rc the number in the AC is incremented. For distinct points the increment should not be smaller than 2^{-6}. After each rc about 170 microseconds must elapse before the computer can perform the next in-out instruction.

(3) Character Displays

(a) The Character Generator

To display alphanumerical information on the oscilloscopes, programs have been used which display such characters by plotting a number of points in their outlines. In order to save both programming and computer time and storage space, an automatic method of displaying certain types of characters on the oscilloscopes can be used. These characters are formed by intensifying the desired lines in a rectangular figure eight, as shown in Figure 7.

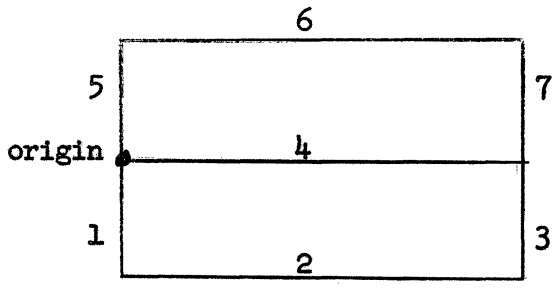


Figure 7

Character Generator Basic Figure

Any symbol which can be formed from the seven lines may thus be plotted automatically. The "origin" of the character is at the left center of the figure eight, and the position of the origin on the face of the oscilloscope is determined by the contents of AC when the si (for vertical deflection) and rc (for horizontal deflection) for displaying the character are given, as is done for plotting individual points (see above), except for the line intensification address. The lines in the figure eight which will be intensified when the rc x order is given are specified by digits 1 to 7 of register x. Each of these seven digits corresponds to one of the seven lines shown in Figure 7; a one in one of these digit positions indicates that the corresponding line should be intensified; that is, a "one" in digit position one of the code intensification line numbered one. Thus the four shown in Figure 8 is formed from lines three, four, five and seven and is specified by the binary number x.0011101xxxxxxxx in register x.

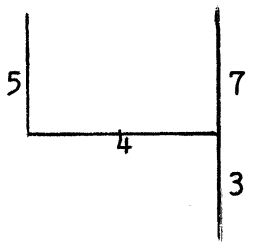


Figure 8

The Numerical Character "4"

The size of the character outline is about 7 units wide and 8-1/2 units high, where a unit is the minimum distance between two points displayed on a scope (which corresponds to a difference of 2^{-10} in the contents of AC when an si or rc order for point display is given).

It is possible also to adjust the size of the character outline so that an expanded display will be given. The adjustment is now one of a factor of $\sqrt{2}$ or 1.414, but it can be changed by equipment settings. The expanded display is available on the following scopes by changing the switch from manual to expanded display: D-12, E-12, F-11, F-13, G-11, and G-13. On all of the above scopes, except D-12 and E-12, the display can be expanded by quadrants also. This is accomplished by switching to expand 1, expand 2 expand 3 and expand 4 for the quadrant I, quadrant II, quadrant III and quadrant IV, respectively. To expand the whole display on these scopes, switch to the expand 0 position.

In order to intiate this expanded display the si 14 (o) instruction must be given, and to release or end this display the si 15 (o) instruction must be given. Either instruction must be coupled with the correct switch setting. These instructions can be given at any time except between the si and rc instructions displaying a character.

(b) Display of a Character

The display of a character is programmed by the following instructions:

si x — selects a scope intensification line for character display. Sets the vertical deflection of all scopes to a value corresponding to the contents of digits 0 - 10 of AC.

rc y — sets the horizontal deflection of all scopes to a value corresponding to the contents of digits 0 - 10 of AC. Intensifies the lines of the character according to the contents of digits 1 - 7 of register y. About 326 microseconds will elapse before the computer can perform another in-out instruction. Any number of instructions other than in-out instructions may precede each rc. Each character to be displayed is programmed in a similar manner.

(4) Vector Displays

(a) The Vector Generator

To aid in displaying alphanumerical data, drawing curves, plotting axes, etc., on the scopes, an automatic method of displaying vectors on the scopes is available. The coordinates of one end of the vector are determined by the contents of AC when the si and rc v instructions for displaying a vector are given, just as occurs when displaying an individual point. The horizontal and vertical components of the vector are determined by a pair of numbers stored in register v. Digits 0 to 5 of this register give the horizontal deflection of the vector from its origin, with maxima of ± 31 units, where a unit is equivalent to 2^{-8} spacing when displaying points; digits 8 to 13 provide a similar range for the vertical component of the vector. Digits 0 and 8 respectively are the sign digits of the two components. The largest horizontal or vertical vector is thus about 1/16th the distance across the usable portion of the scope face (for 16" diameter scopes, only the inscribed square is used, and thus for these scopes the maximum horizontal or vertical line which can be made with a single vector is about 0.7 inches long). A zero length vector appears to be a single point, but no light gun action can be taken on it.

(b) Display of a Vector

A vector is displayed by the following instructions:

si u — selects a scope intensification line for vector display. Sets the vertical deflection for all scopes to a value corresponding to the contents of digits 0 - 10 of AC.

rc v — sets the horizontal deflection of all scopes to a value corresponding to the contents of digits 0 - 10 of AC. Intensifies a vector starting at the point whose coordinates have just been established, where the sign and length of the horizontal component are given by the first six digits of v, and the sign and length of the vertical component are given by digits 8 to 13 of this register. About 166 microseconds will elapse before the computer can perform another in-out instruction. Any number of instructions other than in-out instructions may precede each rc. Each vector to be displayed is programmed in a similar manner.

(5) Oscilloscope Amplification Settings

Since the characters are of such small size that, with the usual gain settings of the test control scopes for point display, they are too small to be photographed clearly, a rotary switch has been mounted below the camera in test control so that in one position the usual gain and intensity settings for point display occur on the scope with the camera, and in the other position the gain and intensity settings have been increased so that the characters will be legible when photographed. However, with the switch in this last position, the maximum deflections for the origins of the characters cannot be used. The limits of the scope face will be reached with deflections of the character origins 1/2 to 2/3 the maximum permissible with point displays.

(6) si Addresses for Scope Lines

The si addresses referring to oscilloscope displays are as follows:

point display	from si 600 (0) to si 677 (0)
	or
	from si 384 (d) to si 447 (d)
vector display	from si 1600 (0) to si 1677 (0)
	or
	from si 896 (D) to si 959 (D)
character display	from si 2600 (0) to si 2677 (0)
	or
	from si 1408 (D) to si 1471 (D)
	from si 3600 (0) to si 3677 (0)
	or
	from si 1920 (D) to si 1983 (D)
expanded character display	si 14 (0) or si 12 (D)
release expanded character display	si 15 (0) or si 13 (D)

As indicated above, the connections of the oscilloscopes to the above lines are determined by the particular switches associated with each scope and the wiring of these switches.

3.9 CAMERAS

Two cameras are available with the in-out display system. The primary camera is a Fairchild 35mm automatic scope camera mounted to photograph one of the 16" display scopes in test control. The focus and aperture setting of this camera are controlled manually, but its film may be indexed and an exposure cycle controlled by computer orders. A second camera is available in the direction center.

3.9.1 Camera in Test Control

(1) Action of Camera

The selection of the camera by the si 4 instruction results in the following cycle of operations, termed an index cycle:

(a) Bulb lights, recording the frame number, time and card data for the previous frame.

(b) Close shutter.

(c) Advance film one frame.

(d) Open shutter.

The shutter of the camera is normally open when camera power is on, so that a display can be plotted leisurely with each dot registering on the film as it appears.

(2) Programming for the Index Cycle

The index cycle is affected by the single instruction si 4. About 500 milliseconds will elapse after this si instruction before the computer can perform another in-out instruction. If the scope is used as output, the camera should be indexed at the end of the display to eliminate a loss of data in the last frame in the case of the next user omitting an initial index.

(3) Manual Control

A push button, labeled "INDEX," on the camera control panel in test control provides for manually indexing the camera. Each time this button is depressed, an index cycle is given.

(4) Film Alarm

When the film supply in the film magazine on the camera gets low, an alarm buzzer sounds and a red light on the camera control panel goes on (labeled "FILM ALARM"). The magazine should then be replaced with one containing fresh film. When the alarm activates, however, there are still at least ten frames of film left in the camera magazine. The buzzer will be turned off when the operator presses the "ALARM ACKNOWLEDGE" button on the panel, but the light remains lit until the magazine is replaced by one with sufficient film.

3.9.2 Direction Center Camera

This camera is designed to record scan-by-scan displays of radar data during special operations.

The selection of this camera by the si 24 (o) instruction results in the same cycle of operations as the above test control camera index. This command will require approximately 30 microseconds of computer time and will permit immediate use of another part of the WWI input-output system. To permit the proper operation of the cameras, at least 500 milliseconds of program time should appear between two si 24 (o) instructions.

3.10 CLOCK (TIMING REGISTER)

The real-time clock associated with the WWI computer is a 21 binary digit counter (counting modulo 2^{21}), whose normal* input to the right-most digit is a 60-pulse-per-second source synchronized with the AC power lines. It is possible to read out (to IOR) only the first 19 digits of this counter; the last two digits provide frequency division so that the counter output can advance one count each 1/15th second. Thus, the output of the clock can indicate the time as a number of 1/15th-second intervals, counting from 0 to a maximum of $2^{19} - 1$, which corresponds to about 9.7 hours. Provisions have been made for clearing all the flip-flops of the clock, complementing the first 19, and reading out the first 19 digits in two parts. (See Figure 9.)

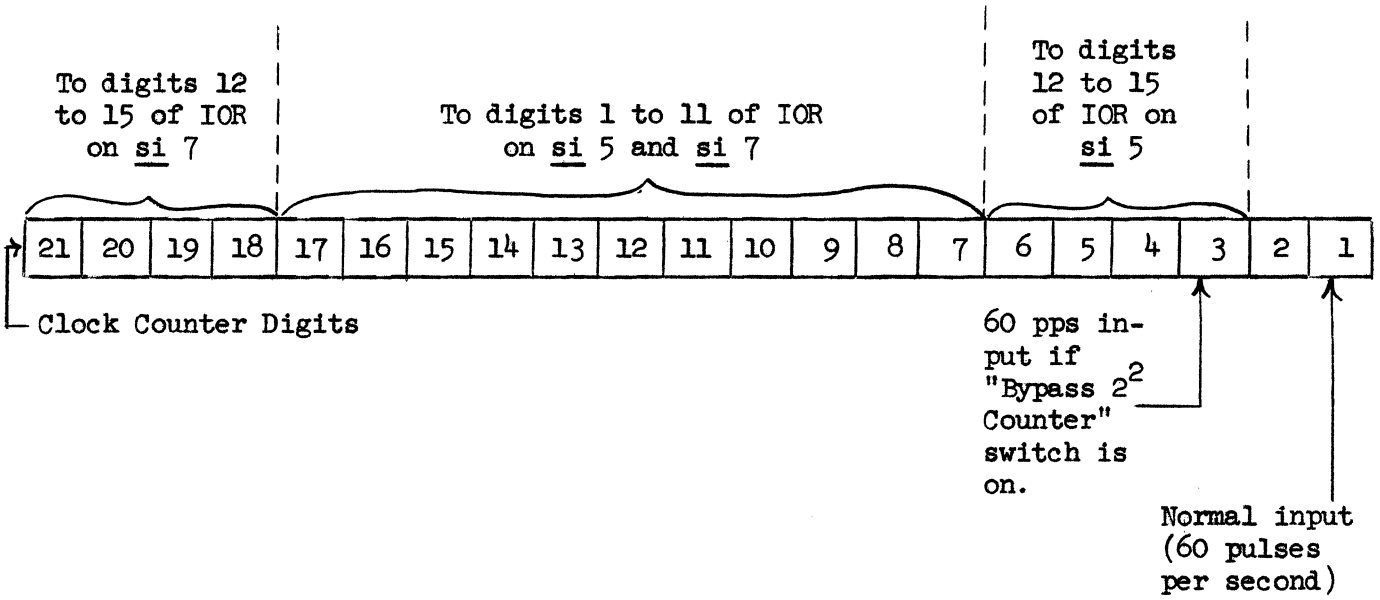


Figure 9

Clock (Timing Register)

---

* Other pulse sources can be fed into this counter. This source must be cabled into a special input of the counter and a switch (in rack E7) thrown up. When this switch is up, a light on the control panel labeled "TIMING REGISTER, EXTRAORDINARY INPUT" is lit.

3.10.1 Programming to use the Clock

(1) Fine Count

si 5 — transfers the contents of digits 17 through 3 of the clock to corresponding positions from digits 1 through 15 of IOR. Digit 0 of IOR is 0.

rd-- — transfers the contents of IOR to AC.

(2) Coarse Count

si 7 — transfers digits 17 through 7 of the clock into corresponding positions from digits 1 through 11 of IOR, just as is done on si 5 above. Digits 21 through 18 of the clock are transferred to digits 12 through 15 of IOR. Digit 0 of IOR is 0.

rd-- — transfers the contents of IOR to AC. This number must be rearranged to its original position in the clock register to provide the proper time count in increments of 1 and 1/15th seconds.

(3) Miscellaneous

si 11 (o)
or — clears all 21 digits of clock (sets all flip-flops to 0).
si 9 (d)

si 12 (o)
or — complements the first 19 digits of the clock (digits 21 through 3).
si 10 (d)

In order to provide more accurate timing for certain programs, a switch has been provided which by-passes the source of pulses around the last two flip-flops of the counter (thus it can count 1/60th second intervals). Otherwise the clock operation is unchanged. When this switch is thrown, a light on the control panel, labeled "TIMING REGISTER, BYPASS 2^2 COUNTER," remains lit.

When a switch on the console labeled "STOP ON si 11 OR si 12" is on, a light on the control panel (similarly labeled) is lit and the computer will stop if an si 11 or an si 12 instruction is given.

3.11 MANUAL INTERVENTION REGISTERS AND INDICATOR LIGHTS

Switches are provided to insert up to 480 binary digits of information into the computer via the digits of thirty 16-digit registers, the insertion registers, which may be sampled directly by the computer. Also, 32 activate buttons are available. Each activate button controls one digit of two 16-digit registers, the activate registers, which may also be sampled directly by the computer. Eight indicator light registers are available to allow computer output indications.

The system provides several sets of switch inputs. Each set may signify different types of information. There is a push button (activate button) and an activation indicator light associated with each set of switches. It is possible to use either insertion switches or activate buttons separately, assuming the program is properly designed; however, programs often use them together. In this latter mode, one would place the information into the set of switches and press the associated activate button which turns on the indicator light at the console position. The computer acknowledges "reading" the activate button by turning off the indicator light.

The insertion switches, activate buttons, and indicator lights, along with display scopes and light guns, are distributed among several console positions. Each position may be controlled by a different operator. Since it is impossible to predict exactly how the equipment should be distributed among the consoles, a distribution panel, the remote station junction box, is included in the system. The tie-in between the various units and the consoles will be done at this junction box. The junction box facilitates any desired changes in distribution of various indicators, displays and controls at each console. The connection at the junction box is semi-permanent, and because changes will affect the work of many people, any changes must be carefully planned and should not be contemplated on a day-to-day or program-to-program basis, but should be used to facilitate any general change in procedure.

3.11.1 Insertion Registers

(1) Description of Insertion Registers

A group of special registers has been provided so that asynchronous information can be given to the computer by the operator or other personnel as a program is being performed. Each such register is provided with a number of binary, octal, or decimal switches which are connected such that certain binary digits of the register may be set to 0 or 1 manually, as desired. The information given here is of a general type; thus, any specific switch locations are purposely omitted.

(2) Programming to Use the Insertion Registers

To read the contents of any insertion register, the following sequence of instructions is used:

si x — reads the contents of the insertion register indicated by the address x into IOR. Since the switches which govern this information are of the latching type, the information in the register remains there until some of the switches are changed.

rd-- — transfers the contents of IOR to AC and clears IOR.

(3) si Addresses for the Insertion Registers

The si addresses for the insertion registers (numbered 2 to 31) are as follows:

Insertion Registers 2 - 31 — si 302 (o) to si 337 (o)

or

si 194 (d) to si 223 (d)

Manual insertion registers referred to by addresses 336 (o) and 337 (o) are mounted on a panel on the console in test control, and are used by the computer operators to select special utility programs, etc.

3.11.2 Activate Registers

(1) Description of Activate Registers

Another form of manual insertion of information into the computer is provided by the activate registers. These are two registers of gas tubes (32), any one of which is fired by pressing the push button associated with that tube. When the information is read from such a gas tube, it is automatically extinguished, so that it may be fired again when its button is pushed. Also associated with each tube is a light which remains lit as long as that tube is fired. The information given here is of a general type; thus, any specific locations have been purposely omitted.

(2) Programming to Use the Activate Registers

The contents of an activate register may be read as follows:

si y — reads a word into IOR indicating the states of the 16 gas tubes which form the activate register indicated by the address y. "Ones" will be in the digit positions corresponding to gas tubes fired since this same si y instruction was given previously. These gas tubes are then extinguished.

rd-- — reads the contents of IOR into AC and clears IOR

(3) si Addresses for the Activate Registers

The si addresses referring to the activate registers are as follows:

Activate Register 0	si 300 (o) or si 192 (d)
Activate Register 1	si 301 (o) or si 193 (d)

The push buttons which fire gas tubes corresponding to digits 0 and 1 of the word obtained by using si 300 (o) to read activate register 0 are positioned on the panel on the console in test control, and are used by the computer operators with the insertion registers located on this panel.

3.11.3 Indicator Light Registers

(1) Description of Indicator Light Registers

A group of eight special registers has been provided so that the computer program can give desired information to the operators, programmers, or other personnel. Each such register represents a group of 16 gas tubes which can be fired under control of the computer. An si instruction referring to one of these registers extinguishes all the tubes in that register, and the succeeding rc instruction fires the tubes corresponding to the digit positions in AC which contain "ones." Lights associated with each gas tube of each register remain lit as long as that gas tube is fired. These indicator lights, distributed among the console positions, are provided for displaying binary information from the computer. Either "ones," "zeros," or both "ones" and "zeros" in the indicator light register may turn on lights at the console. Up to four "one" lights and four "zero" lights may be connected to each digit of each indicator light register.* The complement of the contents of the register may also be given in a parallel set of lights. The information given here is of a general type; thus, any specific locations have been purposely omitted.

(2) Programming to Use the Indicator Light Registers

To record a word in an indicator light register, the sequence of instructions used is:

si z — extinguishes all the gas tubes of the indicator light register selected by address z.

rc-- — reads the contents of AC into IOR, and wherever a "one" occurs in this word, fires the gas tube corresponding to that digit in the selected indicator light register. The lights associated with this register then show the contents of AC. Two seconds must intervene between two si instructions for the same light register in order to insure that its gas tubes are completely extinguished.

---

* Special purpose indicators, such as binary-to-octal converters and audible alarms, may also be controlled by the indicator light registers.

(3) si Addresses for the Indicator Light Registers

The following si instructions refer to the indicator light registers:

Indicator Light Registers 0 - 7	si 510 (o) to si 517 (o)
	or
	si 328 (d) to si 335 (d)

The lights for indicator light register 0, whose address is si 510 (o), are located on the panel on the console in test control for use by the computer operators and programmers.

SECTION IV

SPECIAL IN-OUT UNITS

4.1 LIGHT GUNS

4.1.1 Standard Light Guns

The light gun is a photocell device which generates a pulse if it sees a sudden change in light intensity such as occurs when a spot is intensified on a display scope. The gun contains a trigger switch which causes one pulse to be generated for each trigger depression if the in-out switch (IOS) is set to display points. The light gun action will not take effect on vectors or characters. Such a device enables the scope operator to direct the attention of the computer to information represented by certain points on the display scope.

4.1.2 Special Light Gun

A special light gun is available which operates in similar fashion to standard light guns except that only one depression of the trigger is required in order to pick up a sequence of point displays. In order to operate this special light gun, a switch must be thrown to "Light Gun" position on the side of the area discriminator console in the direction center. Since this device is seldom used, a technician should be consulted.

4.1.3 Area Discriminator

Also available is an oscilloscope called the "area discriminator" or "light cannon," which has a photocell permanently mounted above it. This cell operates similar to the light guns except that no trigger is necessary for activation. The photocell will generate a pulse for any point displayed on the scope unless a filter or mark is placed between the point and the cell.

In order to use the area discriminator, two special switch actions in the direction center are required: first, a switch on the side of the area discriminator must be thrown to "Area Discriminator" position; secondly, the display on the area discriminator is controlled by display switches located at E21-12-1 and E21-12-2. (These switches select one of two possible si display lights which are available at the area discriminator.)

4.1.4 Light Gun Wiring

Since each light gun will insert a "one" in the sign digit of the IOR and in one of the other fifteen digits when a point has been detected, a maximum of 15 unique light guns can be installed. After displaying a point, the contents of the IOR may be transferred into the AC to see if the light guns picked up the point. At the present time 13 light guns are available, and wiring is such that the IOR register which is used by the light guns is set up as shown in Figure 10.

Digit No.	Scope No.	Light Gun Station No.
0	All Light Guns use the sign bit	
1	E-12	TS
2	F-11	M #1
3	F-13	M #2
4	G-11	M #3
5	G-13	M #4
6	G-11	Area Discriminator or Special Light Gun
7	Not Used	
8	M-11	Id. S #2
9	J-11	W.D.
10	Q-11	WDM/R
11	P-11	ASC
12	I-11	Id. #2
13	H-11	I #1
14	D-12	Id. 0
15	D-11	Id. S

Figure 10

4.1.5 Programming for Light Gun Inputs

To determine if a light gun signal has occurred with the plotting of a given point, it is necessary to program an rd after the point has been displayed and before another in-out instruction. The rd will bring the contents of IOR into AC; a cp instruction may then be used to examine the sign digit to see if any signal has been received; and successive cl and cp instructions may determine which light guns generated the signals. The address part of the rd instruction is immaterial. (The rc order used to display the point clears the IOR so that a light gun return can set the appropriate digits.)

4.2 RADAR INPUT

An extensive input control system is installed at Whirlwind for the purpose of permitting automatic input of radar data from multiple radar sites. This input system involves a quantity of electronics which is comparable with the electronics of the central computer system. The main functions of this input system are to provide serial-to-parallel conversion from phone line messages to parallel digital words, and to provide buffering such that the asynchronous radar inputs may be adequately assimilated by the central computer.

Two different forms of radar data are handled by this input system: Slowed Down Video (SDV) and Fine Grain Data (FGD). The latter is commonly called an "FST-2 input" since the digital data transmission system is so designated. In handling these two inputs, the radar input system involves several major equipments:

(1) Multiple Input Terminal Equipment (MITE),

(2) The buffer section of the buffer drum,

(3) Mappers.

This memorandum will avoid any description of the actual input system equipment and will primarily stress the format of data and the programming considerations in using this data once it has actually been deposited on the buffer section of the buffer drum.

4.2.1 Slowed Down Video (SDV)

In the SDV system of data transmission, the presence or absence of a target is reported as the presence or absence of a particular pulse in a serial train of input phone line pulses. This implies that equipment at the receiving end must count pulses in order to determine the numerical azimuth and range of a target. The equipment required to do this is called Multiple Input Terminal Equipment (MITES). The existing equipment at the Whirlwind computer can handle up to four simultaneous SDV inputs. These input MITES are numbered for convenience in accordance with the place (slot) on the input buffer drum to which they are connected. The present connections as of this memorandum date are as follows:

Drum Slot	MITE	Radar
Group 0, Slot 4	4	Chestnut Hill
Group 0, Slot 7	7	Derry
Group 1, Slot 4	14	Halibut Point
Group 1, Slot 7	17	Scituate

A particular SDV message consists of an azimuth and a range and is deposited in a single 14-bit register of the buffer drum (14-bit buffer drum registers correspond to bits 2 through 15 of standard Whirlwind words). The azimuth is

recorded in bits 2 through 9 and the lowest order bit is equivalent to 1/256 of a revolution. Range is recorded in bits 10 through 15 and the lowest order bit is equivalent to 3/4 of a nautical mile. Reference: Drawing SC-81263.

4.2.2 FST-2 Inputs

In the FST-2 system of data transmission, the presence of a target is reported as a group of digital binary numbers transmitted serially over phone line circuits. Since the information is already in the form of digital numbers, no "counting" is required, but a serial-to-parallel conversion is performed by the phone line input equipment. Messages in the FST-2 system may take one of several forms; e.g., search radar returns, Mark X returns, or height finder reports. The format of the phone line messages is shown in Figure 11. Unlike SDV, these messages require more than one 14-bit register of the buffer drum per message; in fact, three registers per message are required and such a grouping of three registers is, for convenience, called a "super-register." In addition, the input data rate is such that two buffer drum slots have been allocated per FST-2 input phone line. Furthermore, under certain conditions, a particular radar may require two phone lines to handle the expected data rate.

Equipment is available at Whirlwind to handle up to four FST-2 phone line inputs, thus eight slots of the buffer drum are allocated for this purpose. This arrangement permits inputs from either two radars (at two phone lines per radar) or four radars (at one phone line per radar). The actual connection of particular radars to particular slots is flexible and subject to change; however, a particular common connection is listed below for illustrative purposes:

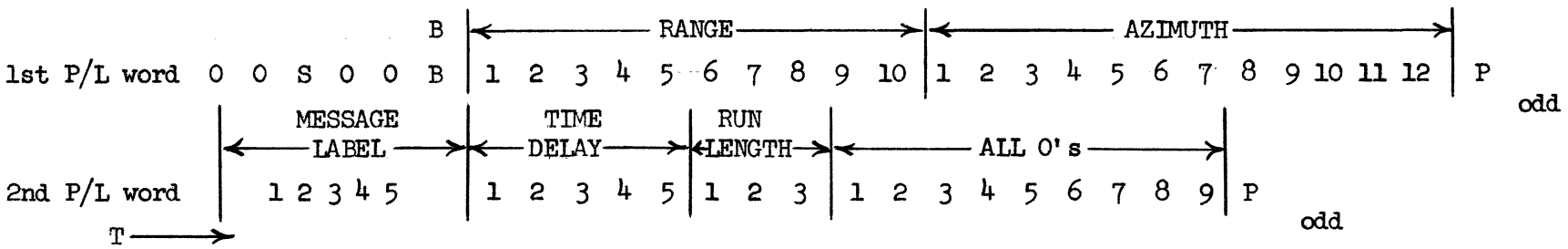
Drum Slot	Radar
Group 0, Slots 0 and 1	Bath, Channel A
Group 1, Slots 0 and 1	Bath, Channel B
Group 0, Slots 2 and 3	Truro, Channel C
Group 1, Slots 2 and 3	Truro, Channel D

Reference: Drawing SC-81263.

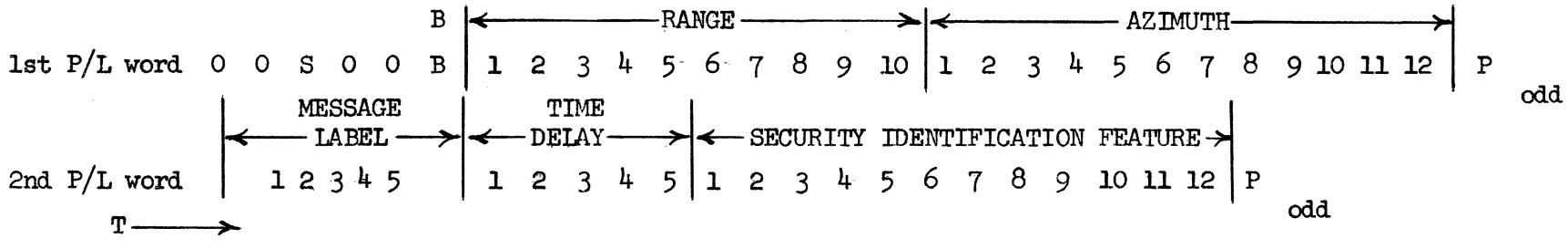
In a particular super-register the FST-2 messages are stored as indicated in Figure 12. Many of the quantities shown on this Figure are subject to changes in definitions as a function of particular applications, and a discussion of these definitions is beyond the scope of this memorandum; however, a few of the more common quantities are defined below:

(1) The Time of Arrival bits will have the contents of the WWI clock, bits 3, 4, 5 and 6, at the time the message is recorded on the drum. The time of recording will be a maximum of 17 milliseconds after the time of arrival. The least significant bit is added to approximately every 1/2 second (actually 8/15 second). If when the FST-2 input data is read from the drum, the WWI clock is read and stored in register X with bit 3 of the clock in bit 15 of the register X, and if the time recorded with message M is in register T, then the following three orders will give the length of time this message was on the drum:

PHONE LINE SIGNAL, SEARCH RADAR



PHONE LINE SIGNAL, MARK X WITH SIF



PHONE LINE SIGNAL, HEIGHT FINDER

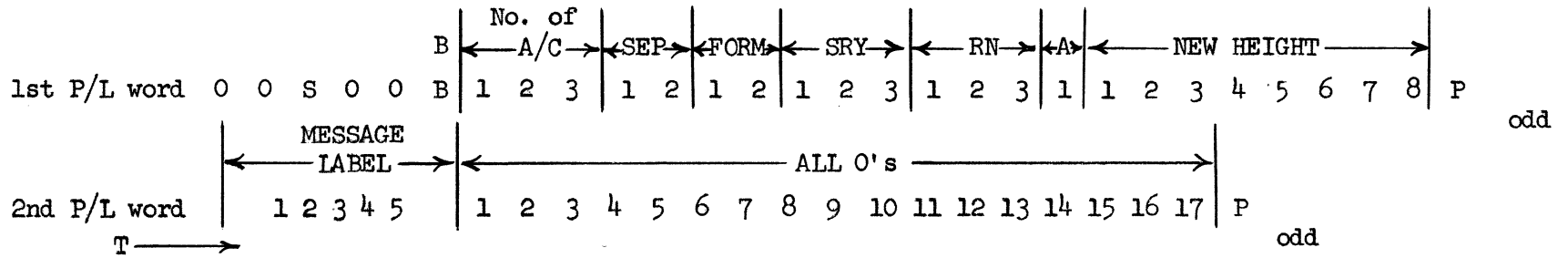
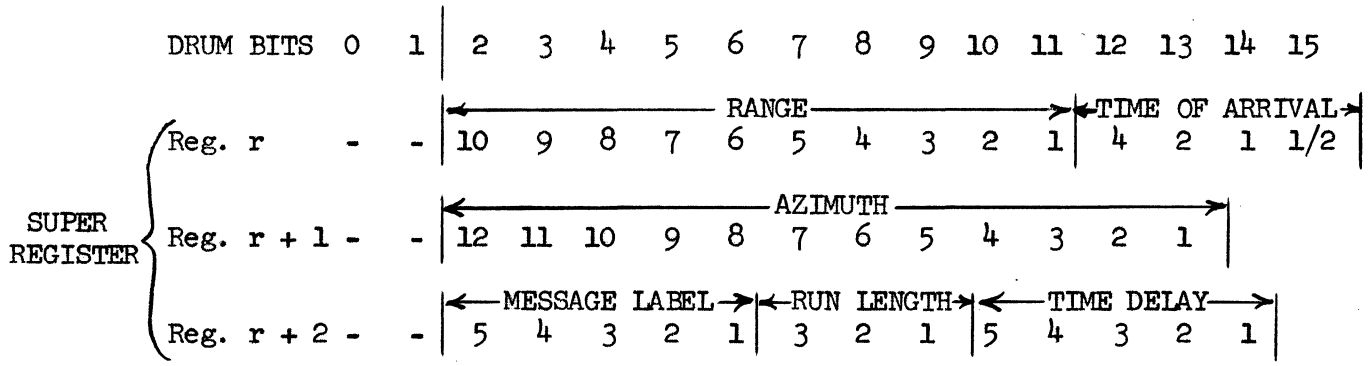
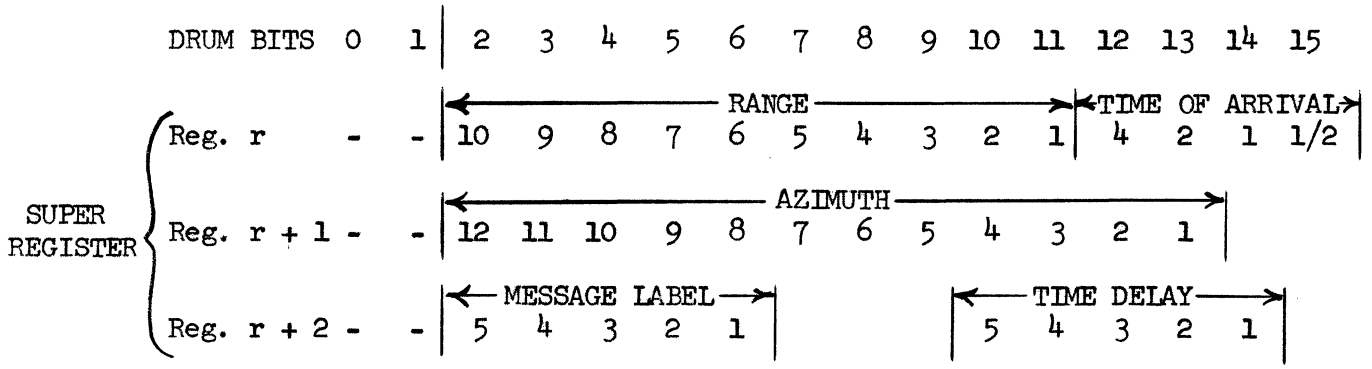


Figure 11

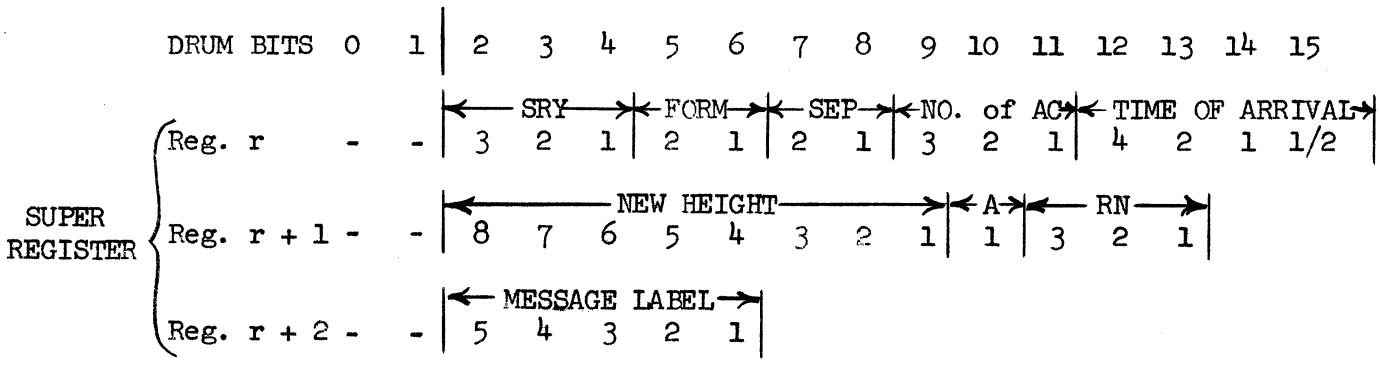
SEARCH RADAR



MARK X



HEIGHT FINDER



FST-2 INPUT DATA AS STORED ON THE BUFFER DRUM

Figure 12

```
ca  X
su  T
md  Y          Y contains 0.00017
```

This procedure assumes that the WWI clock did not end-carry between the time message M arrived and the time it was read from the drum, in which case the answer would be off by 1/15 second.

(2) The Range bits of the search radar and Mark X returns will have the range to the nearest 1/4 nautical mile with a maximum range of 255-3/4 miles.

(3) The Azimuth bits of the search radar and Mark X returns will have the azimuth to the nearest 1/4096 revolution which is approximately 0.088 degrees.

(4) Some Message Label definitions are tabulated below:

Bit No. 5	4	3	2	1	Meaning
0	0	0	0	0	No data
0	0	0	0	1	Search Radar
1	0	0	0	1	Height Finder
0	1	0	0	1	Mark X
0	0	1	0	1	Clutter Map

(5) The Time Delay bits of the search radar and Mark X returns will have the length of time, to the nearest 1/4 second, between the occurrence of the data and its transmission on the phone line.

4.2.3 Buffering and the Switch Fields Technique

The primary reason for existence of the buffer drum is to store asynchronously arriving information until the central computer can absorb it in large blocks. Incoming radar data must be continuously recorded as it arrives, and at the same time the central computer must be able to read the buffer drum essentially at its convenience. In order to permit this operation, the four groups of the buffer section of the buffer drum are divided into two fields such that at any given time two groups are connected to the external equipment and two groups to the computer. In particular, the four groups are designated 0_A, 0_B, 1_A, and 1_B. Group 0_A and Group 1_A is designated Field A; likewise, Group 0_B and Group 1_B is designated Field B. When Field A is connected to the computer, Field B is connected to the external equipment, and vice versa. In normal operation, once the central computer has absorbed the information from the field to which it is currently connected, it can switch the fields and begin reading the newly deposited information. The field switching is accomplished by a particular si instruction (si 734 (o)), thus immediately after

reading all the data from a particular field, the program would normally switch fields. The equipment is so designed that reading data from the buffer section of the buffer drum clears the groups which have been read, preparing them for the recording of additional data.

4.2.4 Programming Considerations

The buffer section of the buffer drum is arranged such that each group consists of eight slots and each slot consists of 400 octal registers. The registers in a particular slot are interleaved in such a fashion that the use of a block-in order starting at a particular place on the buffer section of the buffer drum would bring in consecutive registers which are physically eight registers apart on the drum surface. This fact, however, does not require any consideration by the programmer since it happens quite automatically in the drum circuitry. For programming purposes, the eight slots of each group may be considered separate entities. It is normal to read one entire slot at a time and then to go on reading any other slots with individual block-in instructions.

To read from any given slot, the following program may be used:

ca RC (starting drum slot address)

si 713 (o)

ca RC (number of words to be read $\leq$ 377 (o))

bi RC (initial address of block of registers in MCM to which the contents of the slot will be transferred)

In order to read in SDV information it is clear that one is only required to read one slot per radar; however, for FST-2 inputs one may be required to read up to four slots per radar in order to accept all incoming information from that radar.

4.2.5 Mapping

Data from radar sources often contain considerable clutter and noise in addition to returns from aircraft. It is desirable to eliminate such information, if possible, prior to recording on the buffer drum. In order to accomplish this a technique called "mapping" is available. Data is displayed on a cathode ray oscilloscope and a human operator attempts to map (cover up) that data which he feels is undesirable or meaningless. A photocell mounted above this oscilloscope then only transmits to the rest of the system that data which was not mapped. Mapping in connection with FST-2 inputs is normally done at radar sites, therefore, it does not require consideration in connection with use of the Whirlwind facility. However, mapping in connection with SDV inputs is normally done at the computer.

Mapper scopes are therefore available in the Whirlwind direction center for each of the SDV inputs, and in using SDV inputs care must be taken that the mapper scopes are properly set up and operating.

4.2.6 Programmed Checking of Buffer Drum

Under normal operation, recording in the buffer section of the buffer drum is done only by input equipment, and the computer only reads these groups; however, it is occasionally necessary for maintenance purposes to record on these groups, and this is possible with specially constructed programs. Discussions of such programs are beyond the scope of this memorandum.

4.3 CROSSTELL SYSTEM

4.3.1 General

The Whirlwind In-Out System includes a complex and flexible group of equipment to permit ground-to-ground crosstell of various sorts. These equipments permit both input and output crosstell operations and, in some cases, permit simultaneous crosstell input and crosstell output. Some of the crosstell equipment uses the buffer drum as a buffer storage medium; other crosstell techniques permit direct communication with the in-out element of the computer. Persons planning to use crosstell equipment should consult Whirlwind engineers or technicians concerning details of operation. In many cases, use of crosstell equipment interferes with other equipment operations and, in all cases, use of crosstell equipment requires special switching operations as well as possible cabling changes. This memorandum will describe the basic techniques, but will omit detailed equipment description and detailed format considerations.

The basic techniques include the following:

(a) Input crosstell of 85-bit (SAGE format) messages via the buffer drum.

(b) Crosstell output on a bit-by-bit basis under direct control of central computer.

(c) Crosstell output of 85-bit messages via a special shift register in the in-out system. This facility is normally called the Height Coder and is described in Section 4.4. It should be noted, however, that it may equally well be used for crosstell output in SAGE format.

(d) Crosstell output of special messages in connection with time division data link studies. This system is called the Digital Output Coder, or DOC system, and is described in Section 4.5. This system employs portions of FST-2 channels which are then unavailable for normal radar input.

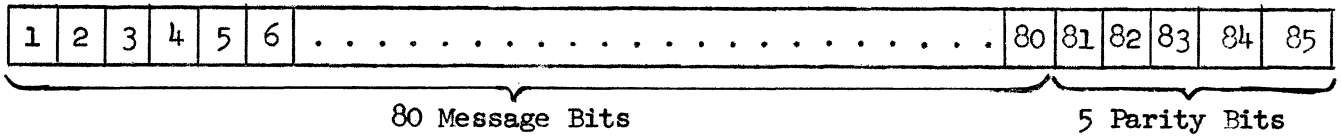
This section of the memorandum (Section 4.3) will describe (a) and (b) above.

4.3.2 Crosstell Input

The crosstell input system at Whirlwind normally accepts 85-bit messages over incoming phone lines, checks parity bits within this message, and splits the message into parts for storage in consecutive registers of the buffer section of the buffer drum. Crosstell messages arrive asynchronously and the crosstell input system can handle input messages at the maximum phone line rate.

The technique for splitting the 85-bit word and depositing it on the buffer drum is rather complex (in order to minimize equipment) and the format arrangement is shown in Figure 13. The message is split into essentially ten parts and five consecutive registers are used in each of two slots of the buffer drum. In order for the central computer to read these messages, the program would normally be required to read two slots of the buffer drum and then reconstruct the proper message from the scrambled bits. Reading from the buffer drum slots for crosstell

CROSSTELL MESSAGE (85 BITS)



BUFFER DRUM, SLOTS 6 AND 16

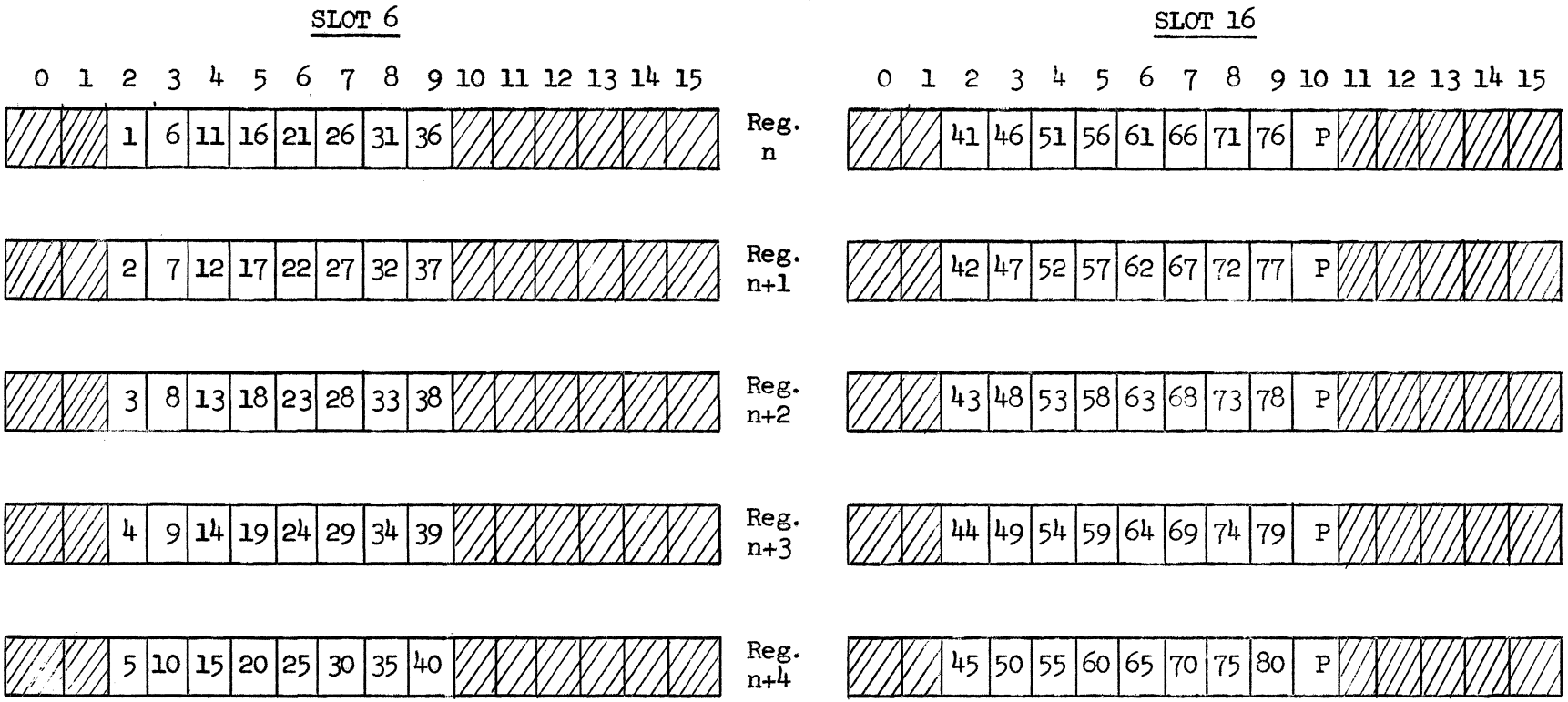


Figure 13

input is accomplished in normal fashion, as described in the Radar Input section of this memorandum (Section 4.2). It is suggested, however, that after each "read" a parity check and a "count of five" check be made. Since the parity of each individual word is not checked by the input equipment, the program should check them before they are used. To insure that all five words of each message are recorded, the total number of "reads" should be checked. If this total is not a multiple of five, the program could recognize that an error has occurred.

In normal fashion, the reading of continuous crosstell input would involve field switching. Each time the fields are switched, however, it should be remembered that two slots must be read to bring in all crosstell input data.

4.3.3 Crosstell Output (Bit by Bit)

The Whirlwind Crosstell System includes a facility for producing output crosstell messages on a bit-by-bit basis under direct control of the central computer. Several si orders are reserved for specifying the value of output bits. In particular, the execution of an si 410 instruction transmits a single "1" bit on the phone line, and the execution of an si 400 instruction transmits a single "0" on the phone line. In actually using this equipment, it is appropriate, however, to first properly synchronize the system and clear any old information. A normal program would then look as follows:

```
si 420    Synchronize system
si 401    Clear equipment
   .
   .
   .
(followed by strings of si 400 and si 410 instructions)
   .
   .
   .
```

Bits are transferred by the output equipment at a 1300 cps rate, and in using this system the central computer waits for a completion pulse in the operation of each si instruction. Therefore, the programmer does not have to worry about any timing considerations, but may merely string together appropriate si instructions, either in a long string or a cyclic loop, without timing difficulties (as long as the cyclic loop does not include more than several hundred microseconds of time).

With this technique, messages of any desired format may be generated; however, it is to be noted that for normal applications, format restrictions on the receiving end must be carefully followed.

4.4 SEMI-AUTOMATIC HEIGHT FINDER REQUEST CODER

A Height Request Output Coder (HT CODER) is part of the special input-output system at WWI. The HT coder design requires the use of the present crosstell input and output system (XT) for its operation. Therefore, when the HT coder is in use, the XT is disabled (with a switch in TC). When the HT coder is not in use, the XT will function normally. (See Section 4.3.1.)

Each height request is initiated by an in-out order, si 6. This order will select the HT coder and prepare it to receive information from the computer. The desired information (5 words) may then be transferred to the HT coder by a series of 5 rc orders or by a single bo order. The following is a sample program with rc orders:

```
si 6
ca wl      RC first word
rc
ca 1wl     RC second word
rc
ca 2wl     RC third word
rc
ca 3wl     RC fourth word
rc
ca 4wl     RC fifth word
rc
```

The following is a sample program with bo order:

```
si 6
ca cl      RC + 5
bo wl      RC first word
```

The HT coder will process the information immediately after it has received the fifth word. One si 6 order must be given for each height request. The new coder will process and transmit the height requests according to the SAGE format, using a five-bit sync system. The most significant bits of the five interlaced words are transmitted first. Although there are five interlaced words in a message, only the first four words may contain the height request information. The fifth word must contain all zeros. In addition, certain bits in the first four words must contain zeros. Otherwise, the entire message may be rejected by the

height finder equipment at the site. A sketch showing the format and those bits that must contain zeros is shown in Figure 14. The insertion of the parity bit is made by the coder automatically.

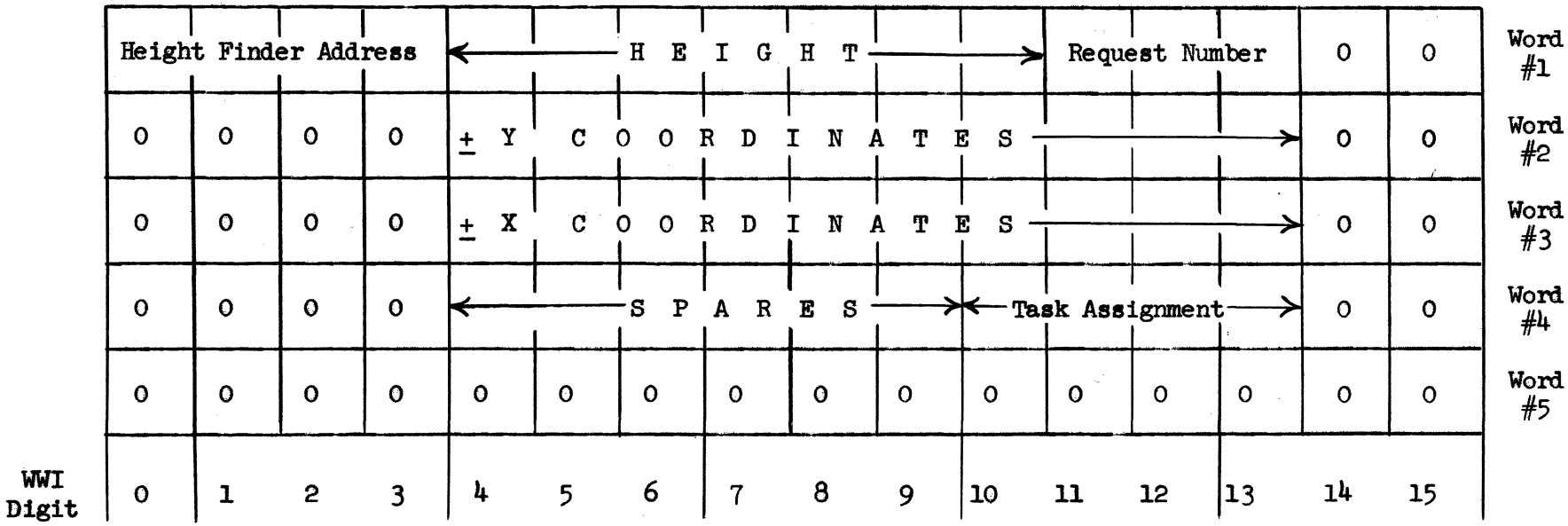
The programmer should allow sufficient time (70 - 140 milliseconds) for the transmission of HT request before attempting to use the HT coder again. Otherwise, the computer will be stopped on the second si 6 to wait for the completion of transmission of the first request. No inactivity alarm will result, but 70 milliseconds of computer time will be wasted. The computer time required to initiate one HT request is approximately 270 microseconds when rc instructions are used, and 150 microseconds when bo is used.

If the si 6 is followed by more than five rc orders, the computer would hang up on the sixth rc order and an inactivity alarm will result. If less than five rc orders are given, then that request is ignored and the next si 6 will clear the HT coder for a new request.

In using the HT coder it is important to properly synchronize and clear the output Crosstell equipment just before the first transfer of a message. This is normally done by preceding the above programs with the following two instructions:

```
si 420      Synchronize system
si 401      Clear equipment
```

Although these two instructions may not always be necessary, it is preferable to employ them for safety purposes.



HEIGHT REQUEST WORD FORMAT

Figure 14

4.5 DOC SYSTEM

The Whirlwind in-out system includes a special set of equipment which was designed for studies of the time division data link. This equipment is basically a form of crosstell output equipment and permits the generation of output messages as a function of combinations of normal Whirlwind words. When this system is in use, several other portions of the in-out system may be disabled. See Section 4.3.1. This section of the memorandum (Section 4.5) is abstracted from 6M-5551.

4.5.1 General

The Whirlwind DOC Output System employs two shift registers and two phone lines which simultaneously transmit each half of the DOC message. Four 16-bit words per message (2 per phone-line) must be transferred from the computer to the shift registers. Although each message contains five parity bits, the programmer can control only one of these; namely, the sub-parity on phone line A. The other four parities are automatically inserted by the computer at the time of transfer to the shift register. The "sync" and timing pulses are supplied by the output coder which is used for both DOC and crosstell. In order that the receiving equipment at the site will see a good message every "sync box," a recycle mode has been incorporated in the output system which will cause the last message transmitted to repeat indefinitely or until the programmer inserts a new message into the system.

In order to conveniently use the DOC system for operating equipment which is distant from Whirlwind, it is helpful if the output can be monitored at the computer. This has been made possible by the installation of a DOC input system which permits output messages to re-enter the computer and thus check on proper functioning of the output equipment. This input system is normally used only for such system checking.

4.5.2 Format and Programming

The phone line message format is shown in Figure 15 along with the correspondence between core memory digits and phone line digits. All parities are even except the sub-parity on phone line A which is odd and is determined by the programmer. Note that parity bit, P_2, on phone line A is calculated on the basis of the preceding 16 digits, including the sub-parity, P_s.

Programming for the DOC output system is normally accomplished in the following manner:

si 420	Delays transfer until sync pulse arrives (synchronization)
si 422	Permits transfer to shift Register A
ca (A_1)*	(See Figure 15 for definition of A_1)
rc	Initiates transfer
si 412	Permits transfer to shift Register B
ca (B_1)	(See Figure 15 for definition of B_1)
rc	Initiates transfer
si 400	Causes a high-speed shift of 17 digits to take place to make room for next transfer

* ( ) = Register containing

si 422
ca (A_2)
rc
si 412
ca (B_2)
rc

} Same as above, only second half of each phone line message

The even parities, P_1, P_2, P_3 and P_4, are automatically inserted by the computer based on the programmer's choice of A_1, A_2, B_1 and B_2. The only restriction imposed upon the choice of these four 16-bit words programwise is that of satisfying the sub-parity, P_s, (Label Parity), on phone line A. The first six digits of A_2 must contain an odd number of "ones" (actually digits 10 through 15 in core memory). Whether or not P_s should be a "zero" or a "one" can either be determined by programming or predetermined at the time the constants corresponding to the various A_2's are inserted into the program.

Executing the instruction si 420 before attempting a computer to shift register transfer, prohibits the destruction of any previous message that might be in the process of shifting out of the shift register. All transfers are performed during the 1540 microsecond period (2 x 770 μsec) after "sync," where 770 μsec is the time between pulses from the shift register. The transmission time for an entire message is approximately 30 milliseconds (40 x 770 μsec). Thus, approximately 29.3 milliseconds are at the disposal of the programmer between successive messages. As shown in Figure 15, the 15th bit of each word is the first to appear on the phone line, and so on down to the 0th bit or sign bit.

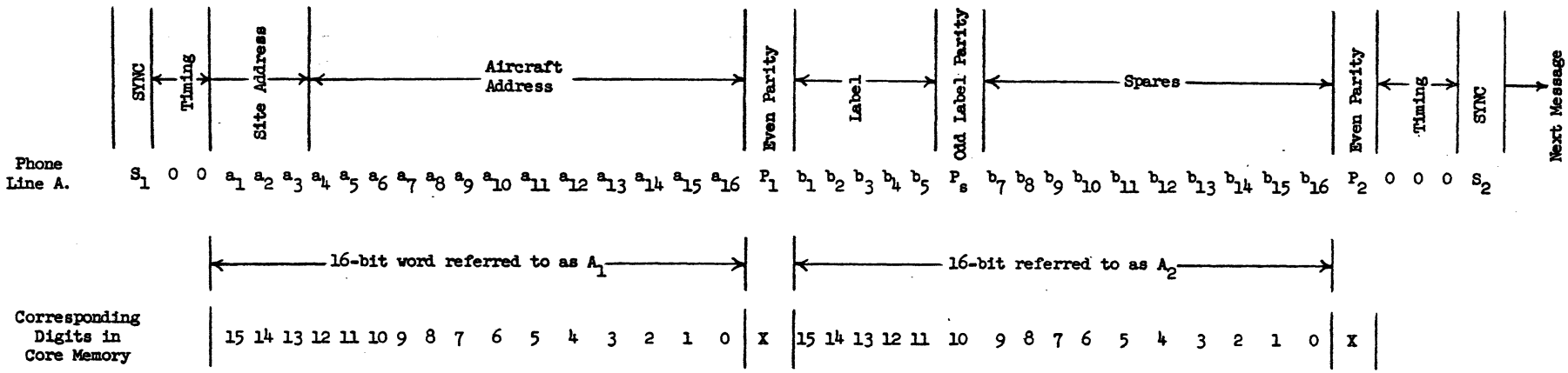
The recycle mode, unless suppressed, will normally be in operation, thereby maintaining phone line saturation regardless of whether or not the DOC system is being continuously driven by the computer. Since a computer-generated message will always take precedence over the recycle mode, no computer-generated messages will ever be destroyed by the recycle mode, and therefore it is not necessary to suppress the recycle mode to avoid interference with computer-generated messages.

The DOC output system normally uses two FST-2 channels (C and D) which are thus normally not available for radar input when the DOC system is in use.

4.5.3 The DOC Input System

Programming for the DOC Input System is essentially the same as programming for reading the drums in connection with the FST-2 system. The input system can record on the following slots:

0	0.00002	DOC	(FST-2)
1	0.01002	System "A"	(System B)
10	0.04002	DOC	(FST-2)
11	0.05002	System "B"	(System A)


SYNC
Timing
Site Address
Aircraft Address
Even Parity
Label
Odd Label Parity
Spares
Even Parity
Timing
SYNC
Next Message
Phone Line A.
S1 0 0 a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15 a16 P1 b1 b2 b3 b4 b5 Ps b7 b8 b9 b10 b11 b12 b13 b14 b15 b16 P2 0 0 0 S2
16-bit word referred to as A1
16-bit referred to as A2
Corresponding Digits in Core Memory
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 X
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 X

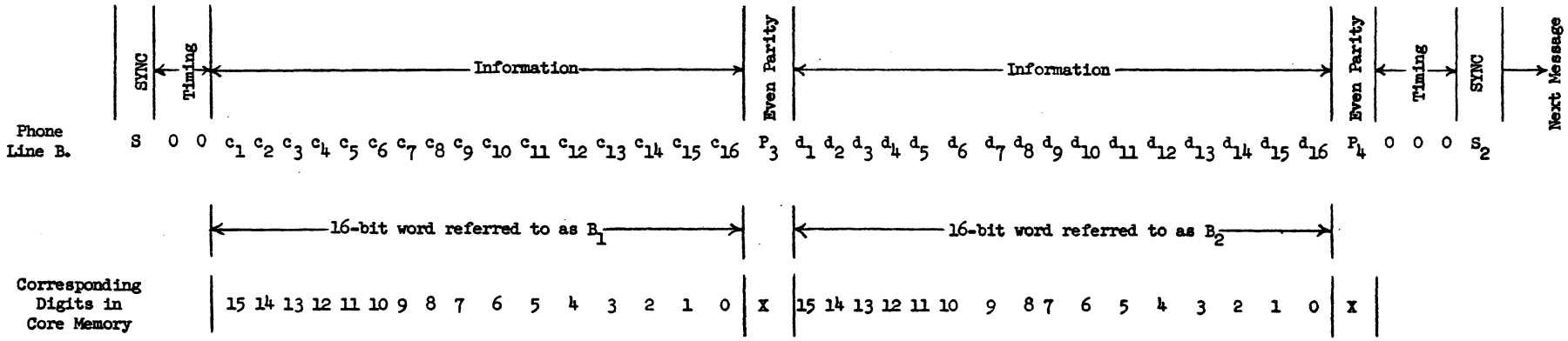

SYNC
Timing
Information
Even Parity
Information
Even Parity
Timing
SYNC
Next Message
Phone Line B.
S 0 0 c1 c2 c3 c4 c5 c6 c7 c8 c9 c10 c11 c12 c13 c14 c15 c16 P3 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 P4 0 0 0 S2
16-bit word referred to as B1
16-bit word referred to as B2
Corresponding Digits in Core Memory
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 X
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 X

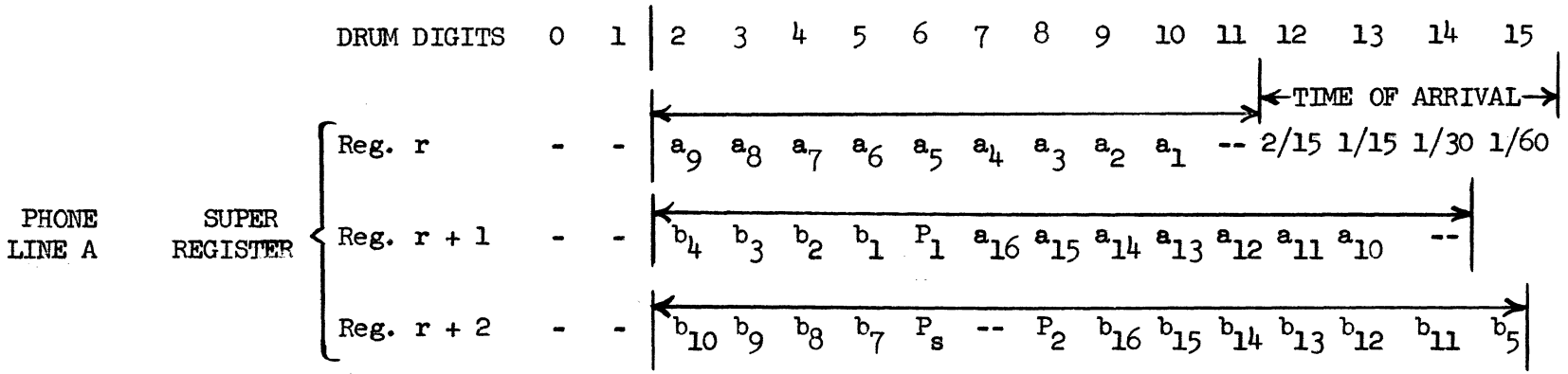
Figure 15

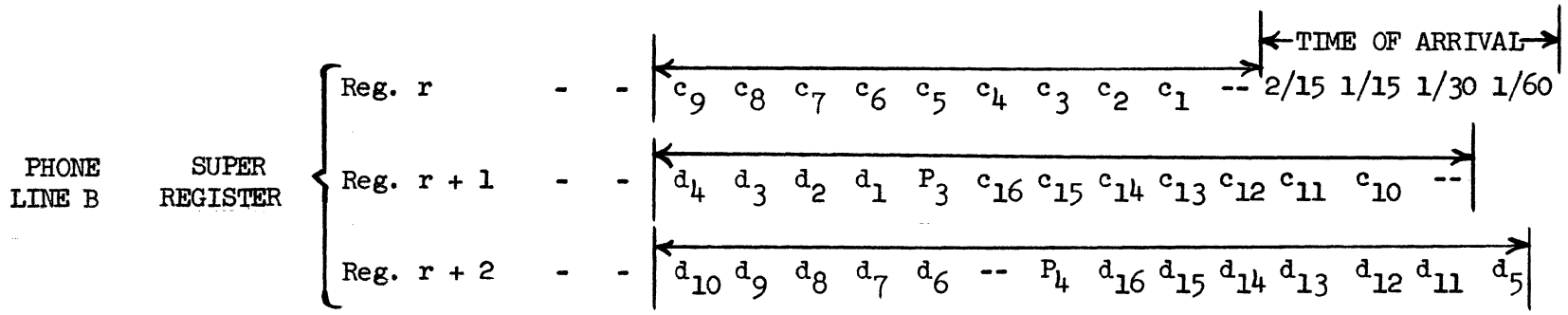
Since the actual recording on the drums is more or less random, depending upon the number of empty registers in each slot and the position of the drum at the time the record is initiated, a particular message may be recorded on one of two slots and in any empty group of three registers. If one starts initially with all slots cleared and records two messages, it is evident that either or both messages may be recorded on a particular slot and that the first message may be recorded in a higher register address than the second message. Thus, under the most general case, a programmer would have to read all four slots in order to be positive of finding both or either of these messages. After both messages are read, the problem of determining which was first recorded (in time) still exists. Since each slot has a capacity of 85 messages, it is not necessary that two slots per phone line be used with the DOC system. The drum reading time can be cut in half if the programmer initially fills up, say, slots 1 and 11 in both field A and B and does not read these two slots for the remainder of the test. This procedure will force the recording of all messages on slots 0 and 10.

Four digits of the Whirlwind Timing Register (clock) will be recorded along with each message, as shown in Figure 16. If the clock is cleared (si 11) before sending each burst of messages, the time tag will indicate which message was recorded first, and so on. The four Time-of-Arrival digits (T/A) correspond to 2/15, 1/15, 1/30, and 1/60 of a second. Bursts of greater than eight messages may result in ambiguous time tags, since an end carry will cause the clock to return the four digits to the original cleared state.

It should be recalled that if a burst of messages is being recorded on, say, Field A, a switch fields (si 734) must be programmed in order to read these messages from Field A. Also, while one is reading Field A, the messages being sent via the recycle mode are being recorded on Field B. Figure 16 shows the correspondence between phone line digits and drum digits.

On the assumption that the DOC system was to be used entirely for sub-systems testing, it was decided to cause all messages to be recorded on the drums regardless of whether or not the parity requirements were satisfied.





Lower Case letters are defined in Figure 15.

Figure 16

4.6 TELETYPE SYSTEM

There are two teletype systems at Whirlwind I -- the Teletype Input System and the Teletype Output System. The Input system transmits information to the computer via the input buffer drum. The Output system uses a direct connection between the Whirlwind In-Out register and the teletype control system.

The teletype systems being used operate at a maximum rate of 60 teletype words per minute; in teletype terminology a word is defined as averaging six teletype characters, and each teletype character requires five bits.

4.6.1 Teletype Input System at WWI

The computer input system permits up to 255 (d) teletype characters to be written on one slot of the buffer drum. Input teletype utilizes slot 5 of groups 0_A and 0_B (address 1.01003) of the input buffer drum. Since a teletype word is defined as six teletype characters, the slot will be full unless read in 255/360 minutes or 42.5 seconds with continuous transmission at the maximum rate. Buffer drum field switching must be considered in normal fashion; i.e., when the input teletype equipment is connected to Group 0_A the computer is connected to 0_B, and vice versa. To read continuous teletype input it would be necessary to switch fields after reading one full group, etc.

Each character of the teletype message appears in the sequence in which the characters were transmitted in separate registers of the drum slot. With a clear drum (be sure that the drum is cleared before attempting to transmit any information), the first character received appears in register 1, the next in register 2, the next in register 3, etc., (of the slot). Each succeeding teletype character takes the first empty register starting from register 1. The teletype character appears in the last five bits of each register in teletype code. Since a teletype character may have two distinct meanings as a function of whether there has been a previous "letter shift" or "figure shift," the input teletype equipment "remembers" the most recent figure shift or letter shift and adds an extra bit to the five-bit teletype message to permit the program to distinguish the otherwise ambiguous five-bit codes. The five-bit teletype code shown in Figure 17, is converted to a six-bit code before being deposited on the buffer drum (Figure 18).

To read slot 5 for the input teletype, the following sample program may be used:

ca	RC (initial slot drum address 1.01003)
si 713	Read from the buffer drum
ca	RC (number of words recorded $\leq$ 377)
bi	RC (initial core address)

FIVE DIGIT TELETYPE CODE

Teletype Code Digit Numbers 1 2 3 4 5	Letters	Figures
1 1 0 0 0	A	-
1 0 0 1 1	B	?
0 1 1 1 0	C	:
1 0 0 1 0	D	$
1 0 0 0 0	E	3
1 0 1 1 0	F	!
0 1 0 1 1	G	&
0 0 1 0 1	H	#
0 1 1 0 0	I	8
1 1 0 1 0	J	' (apostrophe)
1 1 1 1 0	K	(
0 1 0 0 1	L	)
0 0 1 1 1	M	. (period)
0 0 1 1 0	N	, (comma)
0 0 0 1 1	O	9
0 1 1 0 1	P	∅ (zero)
1 1 1 0 1	Q	1
0 1 0 1 0	R	4
1 0 1 0 0	S	Bell and line feed
0 0 0 0 1	T	5
1 1 1 0 0	U	7
0 1 1 1 1	V	;
1 1 0 0 1	W	2
1 0 1 1 1	X	/
1 0 1 0 1	Y	6
1 0 0 0 1	Z	"
0 0 0 0 0	Blank (no action)	
1 1 1 1 1	Letters shift	
0 0 1 0 0	Space and letters shift	
0 1 0 0 0	Line feed only	
0 0 0 1 0	Carriage return and line feed	
1 1 0 1 1	Figures shift	

Figure 17

Binary Representation of Bits 10 - 15	Octal Representation of Bits 10 - 15	Character	Binary Representation of Bits 10 - 15	Octal Representation of Bits 10 - 15	Character
000000	00	Unassigned (unused)	100000	40	Unassigned (unused)
000001	01	T	100001	41	5
000010	02	Carriage Return (see also 42)	100010	42	Carriage Return
000011	03	O (letter O)	100011	43	9
000100	04	Space	100100	44	Space
000101	05	H	100101	45	# (unused)
000110	06	N	100110	46	7/8 (unused)
000111	07	M	100111	47	. (period) (unused)
001000	10	Line Feed (see also 50)	101000	50	Line Feed
001001	11	L	101001	51	3/4
001010	12	R	101010	52	4
001011	13	G	101011	53	& (unused)
001100	14	I	101100	54	8
001101	15	P	101101	55	Ø (zero)
001110	16	C	101110	56	1/8 (unused)
001111	17	V	101111	57	3/8 (unused)
010000	20	E	110000	60	3
010001	21	Z	110001	61	" (quote) (unused)
010010	22	D	110010	62	$ (unused)
010011	23	B	110011	63	5/8 (unused)
010100	24	S	110100	64	Bell
010101	25	Y	110101	65	6
010110	26	F	110110	66	1/4
010111	27	X	110111	67	/
011000	30	Z	111000	70	- (hyphen)
011001	31	W	111001	71	2
011010	32	J	111010	72	' (apostrophe) (unused)
011011	33	Figure Shift (see also 73)	111011	73	Figures Shift
011100	34	U	111100	74	7
011101	35	Q	111101	75	1
011110	36	K	111110	76	1/2
011111	37	Letter Shift (see also 77)	111111	77	Letters Shift

Figure 18

4.6.2 Automatic Teletype Output System

The automatic teletype output system is connected by means of a multi-point phone line to a number of receive-only teletypewriters and to a monitor located at the Barta Building.

The teletype output system accepts one word at a time from the IOR and interprets the right-hand 15 digits as a set of three teletype characters (for convenience, this group of three characters is defined as a teletype syllable). The program must take responsibility for sending properly structured teletype syllables to the output system in approximately the proper time sequence.

On the output of the IOR a fifteen-digit buffer register transfers five digits (i.e., one teletype character) at a time to a teletype transmitter distributor. The transmitter distributor adds two teletype synchronizing digits ("start" and "stop") per character, and processes the character onto the phone line in serial fashion at a 60-teletype-word-per-minute rate. (A teletype word is defined as six teletype characters and thus is equivalent to two fifteen-digit WWI words.) After the first teletype character has been transmitted, the two remaining five-digit characters in the buffer register are transferred successively to the transmitter distributor. Control circuitry controls the shifting in the buffer register and synchronizes the output system for proper operation. A typing reperforator is installed in the Barta Building for the purposes of monitoring the output teletype messages.

In order to avoid the necessity of precise program timing, the teletype output system informs the program when a new teletype output request is being made too soon. If one syllable (syllable A) is being processed by the output system and the program attempts to send a new syllable (syllable B) to the system, syllable B is rejected and the program counter is indexed by two.

To transmit a single teletype syllable, the following short program may be used:

```
ca RC (syllable)
si 402
rc
```

In order to make use of the facility for determining if the output system is ready for a new syllable, the following programming technique may be used:

Register	Contents
51	ca RC (syllable)
52	si 402
53	rc
54	ao 51
55	sp

Assuming a group of syllables are stored in sequential registers when control is transferred to register 51, a syllable is transmitted on the rc instruction to the teletype output system and the ao instruction then indexes register 51 so that the next time through the program, the next syllable will be procured. However, if the teletype output system were not prepared for the new syllable, the rc instruction is started but not completed. The PC is indexed by an additional "one" and instruction 54 is omitted. Under these circumstances, the address section of register 51 is not indexed and the program will still try to send the same syllable, next time through.

To illustrate the format of a teletype syllable, the three-character word "THE" is set up pictorially in a register in Figure 19. The "T" is contained in digits one through five, the "H" in digits six through ten, and the "E" in digits eleven through fifteen.

PICTORIAL REPRESENTATION OF A REGISTER CONTAINING A TELETYPE SYLLABLE

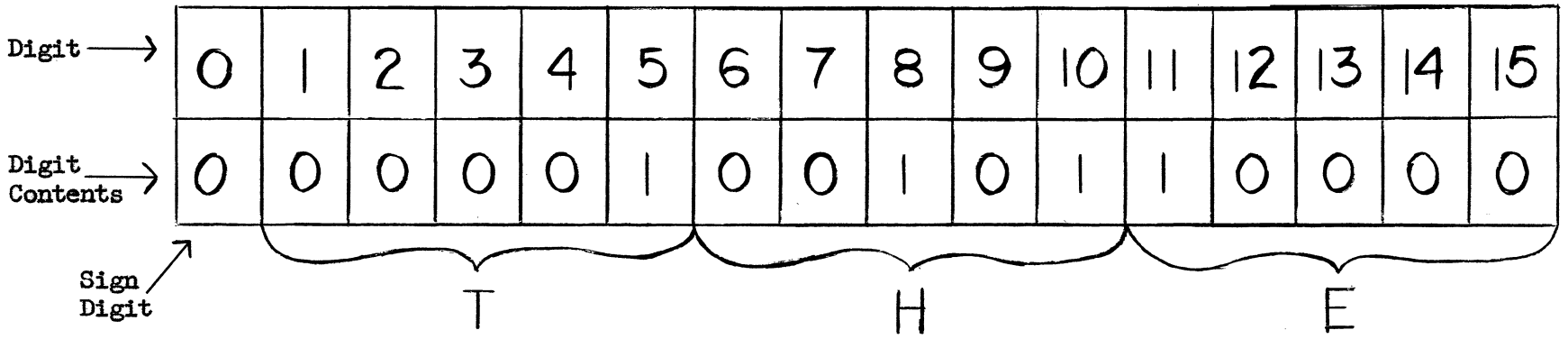


Figure 19

4.7 PROGRAMMED CORE CLEARING

A special mode of clearing magnetic core memory can be selected by the use of the instruction si 17 (0) which initiates the process of reading +0 into a block of registers, thereby clearing them. A bi y, where y is the initial address of the block, then clears the block length determined by ±n in the accumulator, where n is the number of registers to be cleared. For example, the following octal program:

```
si 17
ca RCn          (n is number of words in AC)
bi RCy          (y is initial register address of the block)
```

Only a certain number of registers from a particular address will be cleared. If these registers appear consecutively in the two fields being used, only one of the above sequences of instructions is necessary. For instance, if the last registers of the first field being used and the first registers of the other field being used are to be cleared, only one sequence of instructions is necessary. However, if the registers are not consecutive, the above sequence of instructions must be repeated.

SECTION V

COMPUTER ALARMS

In connection with program or equipment malfunction, special indicators called "Alarms" are available in the form of bright red panel lights. The more common alarms are listed below with an indication of the conditions under which they may occur.

5.1 ARITHMETIC CHECK ALARM OR OVERFLOW ALARM

Misuse of any one of the following nine WWI instructions would result in the arithmetic check alarm. The occurrence of this alarm due to programming can be caused only as specifically noted.

ab: if the absolute value of the sum of the contents of register x and the contents of the B-Register is greater than or equal to 1; i.e.,

$$|C(x) + C(BR)| \geq 1$$

ca: if the absolute value of the sum of the contents of register x and the contents of special add memory times $2^{-15} = 1$; i.e.,

$$|C(x) + C(SAM)\ 2^{-15}| = 1$$

cs: if the absolute value of the difference between the contents of special add memory times 2^{-15} and the contents of register x equals 1; i.e.,

$$|-C(x) + C(SAM)\ 2^{-15}| = 1$$

ad: if the absolute value of the sum of the contents of the accumulator and the contents of register x is greater than or equal to 1; i.e.,

$$|C(AC) + C(x)| \geq 1$$

su: if the absolute value of the difference between the contents of the accumulator and the contents of register x is greater than or equal to 1; i.e.,

$$|C(AC) - C(x)| \geq 1$$

cm: if the absolute value of the contents of register x plus the contents of special add memory times 2^{-15} equals 1; i.e.,

$$|C(x)| + C(SAM)\ 2^{-15} = 1$$

ao: if the sum of the contents of register x and 2^{-15} equals 1; i.e.,

$$C(x) + (1 \times 2^{-15}) = 1$$

slr: if the absolute value of the fractional contents of the sum of the contents of the accumulator and the contents of the B-Register, times 2 to the nth power plus the roundoff from the B-Register, equals one; i.e.,

$$|F\ \{C(AC + BR)\ 2^n\} + \rho| = 1$$

srr 0: if the contents of the accumulator plus the roundoff from the B-Register are equal to 1.

5.2 DIVIDE ERROR ALARM

This alarm occurs because of programming only if the absolute value of the contents of the accumulator is greater than the absolute value of the contents of register x; i.e.,

$$|C(AC)| > |C(x)|$$

5.3 DRUM INTERLACE ALARM

This alarm occurs when information has been requested from the drums and none has been received after two complete drum revolutions. Although this alarm can be caused by programming, its occurrence usually indicates a computer malfuntion.

5.4 DUPLICATE FIELD ALARM

This alarm occurs upon simultaneous use of a field of core storage by Group A and Group B control. The reading in of duplicate numbers into Group A control and Group B control stops the computer and lights the "Field Duplication" alarm light.

5.5 ILLEGAL FIELD ALARM

This alarm occurs when a group control register requests the non-existing group 6 or 7.

5.6 ANELEX ALARM

This alarm light on TC3 will be lit by each one of the following alarms which will cause alarm lights to be illiminated at the Anelex machine itself and on the Anelex control panel on TC1.

5.6.1 Paper Alarm

(1) This alarm may occur because of failure to receive a start pulse if the programmer fails to select one of the six format positions. The computer hangs up within the bo instruction unless the control register appears as the last word in the block, in which case it hangs up at the beginning of the next Anelex printer bo instruction.

(2) This alarm also occurs on a paper feed if the previous paper feed failed. The computer stops and printing ceases.

(3) This alarm occurs if the printer is allowed to run out of paper.

5.6.2 Ribbon Alarm

If the Anelex ribbon fails to advance due to equipment malfunction, an alarm is given and the computer stops when the Anelex is selected.

5.6.3 Format Alarm

This alarm occurs when an illegal format or a multiple format is selected. Only one format can be selected at a single time.

5.6.4 Alarm Counter

If the programmer fails to call for a paper feed and continues to print information for several lines, data would be lost and the printer might be damaged. To avoid this every time a control register is blocked out or group control is indexed, a counter is automatically also indexed. If the counter reaches eight, which can be seen in the Anelex flip-flop control panel in TC1, an alarm is given and the computer is stopped. If a paper feed is initiated before the alarm point of eight is reached, the counter is cleared.

5.7 CHECK REGISTER ALARM

A check alarm may occur as a result either of computer malfunction or deliberate programming.

5.7.1 Equipment Malfunction

If the check alarm occurs on time pulse 4 or time pulse 7 of any order, or on time pulse 1 of any order except ck, it may indicate an actual equipment malfunction, resulting in a transfer check alarm.

5.7.2 Programmed Check Alarm

If the check instruction is operated in normal mode, a check alarm including a computer halt may be used as a normal planned procedure (see Section II). The check order in this mode is commonly used by the Utility System to check information on paper tape. If, for example, a check alarm occurs during read-in with PC = 104, it most likely indicates an error in the tape has been uncovered by a sum check.

5.8 PROGRAM ALARM

While an in-out unit is operating in the read mode, the program must complete rd instructions to take information from the IOR before any more information is deposited there by the in-out unit. Should another word arrive at IOR before the first word has been read from IOR, a program alarm will occur. This situation can arise only with the free-running units (the magnetic tape and photoelectric tape reader). In general this alarm indicates information arriving in the IOR when the computer is not ready for it, and since it is not known whether or not this is an error, the program alarm is given.

Should the Ferranti reader or magnetic tape be selected with too much delay between the si and rd instruction, a program alarm results. The IOR is not cleared and information is read from the Ferranti photoelectric tape reader or magnetic tape over information already in IOR.

If the computer is stopped while the Ferranti reader is in use, the Ferranti is not de-selected, and if the tape is pulled forward manually a program alarm will be generated.

A zero length block transfer should not be used with the mechanical tape reader because such an order (bi with ± 0 in AC) actually reads a line or word, which will be lost, from the paper tape. A program alarm may occur.

5.9 INACTIVITY ALARM

If the computer is in "stop clock" for more than about 1/2 second, an inactivity alarm is generally given for computer has waited too long for completion of an operation. However, this alarm will be suppressed when we are performing in-out instructions connected with the magnetic tape equipment, the paper tape equipment, the camera, or Ferranti photoelectric tape reader, since the computer can legitimately be in "stop clock" for periods of this length when these units are being used. Frequently this alarm results from a programming error such as using rc or bo with an si which should be used with an rd.

5.10 MASTER ALARM

This alarm occurs whenever any alarm occurs (except a continuous alarm caused by a stop on an si 0, an Anelex alarm, or a core memory clear alarm).

5.11 CORE MEMORY CLEAR ALARM

This alarm occurs when either one of the two protected push buttons, clear core fields, and clear Group B, is depressed.

5.12 MULTIPLE SELECTOR ALARM

This alarm occurs when two orders are selected at once due to computer failure.

5.13 PARITY ALARM

A parity check system is used with storage on magnetic core memory and magnetic drums. An additional parity digit is recorded with each register on the drum and the cores, and whenever a word is read from the drum and cores, a parity check is performed and a parity alarm given if parity does not check. This alarm usually indicates a computer malfunction but may also occur as a result of an attempt to "bi" or "rd" from an illegal drum group.

5.14 CORE MEMORY CHECK ALARM

This alarm occurs with a failure of the control system of core memory, an equipment malfunction.

5.15 CONTINUOUS ALARM

This alarm indicates that other alarms are being given in a continuous fashion. It normally only occurs in conjunction with machine malfunction or operation in some special machine mode (e.g., with other alarms suppressed by special switch action).

SECTION VI

COMMON SWITCHES AND PUSH BUTTONS

The Whirlwind computer, unfortunately, contains a very large number of switches, buttons, knobs and keys. Most of these are meant to be left alone but a few common ones are intended for programmer use, and the most common of these are listed below with an indication of function.

Type of Switch	Name	Location	Function
P.B.	Erase Cores	Console, TC5	Changes whole surface of every register in the cores to plus zero. Do not press unless computer is stopped.
P.B.	Read In	Console, TC3 + TC6C	Stops computer, if running. Resets PC to beginning of read-in program register 32 (o). Restarts computer which then reads in the tape through PETR. The Flexo-reader may be used by putting a special read-in program in Toggle Switch Storage. At the end of read-in the program starts, or the computer stops if the STOP on si 1 switch is on.
P.B.	Start Over	Console, TC3 + TC6C	Clears most flip-flops in the computer, resets Flip-Flop storage, resets PC to number held in PC Reset Switches, and restarts the computer.
P.B.	Start Over from Rm. 222	Console, TC6C	Enables START OVER button on E-31 to initiate the above action.
P.B.	Start at 35	Console, TC3 + TC6C	Same as START OVER except resets PC to 35 (o).
P.B.	Start at 40	Console, TC3 + TC6C	Same as START OVER except resets PC to 40 (o).
P.B.	Clear Alarm	Console, TC3 + TC6C	Clears alarm indication (lights) any time and that is all.
P.B.	Stop	Console, TC3 + TC6C	Stops the computer from any state.
P.B.	Restart	Console, TC3 + TC6C	Restarts the computer from P.B. mode. Does not change PC.

Type of Switch	Name	Location	Function
P.B.	Order by Order	Console, TC5	If computer is stopped, restarts it. Lets computer run to next TP5 and stops.
P.B.	Examine	Console, TC5	Starts over and runs to TP5, enabling you to inspect in the PAR the contents of the register whose address is in the PC Reset switches.
P.B.	Reset ALL FF's	TC5	If computer is stopped, resets all digits of all Flip-Flop Storage registers to numbers specified by Flip-Flop Reset switches.
P.B.	Clear	TC3	Clears all flip-flops; i.e., sets them to +0.
P.B.	Complement	TC3	Complements contents of all flip-flops.
P.B.	Index Camera	Camera, Control Panel, TC6C	Initiates the index cycle which indexes the camera one frame independent of the computer.
P.B.	Scope Illumination	Camera, Control Panel, TC6C	Illuminates the scope located on TC6C for the length of time that the button is depressed.
T.S.	Program Check on Special Mode	TC6C	See Section 2.7 ck instruction.
T.S.	Stop on si 1	Console, TC6C	If switch is on (UP), lets si 1 stop the computer.
T.S.	Stop on si 11 and si 12	Console, TC6C	If switch is on (UP), lets both si 11 and si 12 stop the computer. See Section 3.10.1.
T.S.	Stop on Selected Pulse	Console, TC4	When UP, this switch enables the computer to stop on any order or orders he may desire and on any particular time pulse of those orders. The desired order or orders must also be selected in the adjacent set of tiny toggle switches. 32 such toggle switches are available, one for each instruction. Also on this panel 8 push buttons are available to select the particular time pulse on which a stop is desired. The order stopped on will always be in the control switch except on TP4 when the control

Type of Switch	Name	Location	Function
			switch contains si instruction, and on TP5 when the control switch contains the next order. This does not apply to the bi and bo instructions which can be stopped on any time pulse.
T.S.	Stop on Selected Register	Console, TC4	When UP, this switch enables the computer to stop on any order or orders with a particular address. This technique is extremely convenient in trying to stop at a particular place in the program. A set of tiny address toggle switches must be used to request the particular address, and one or more of the tiny instruction selection toggle switches must be used to select the order or orders. Due to close timing, a re-start after using this technique may skip a register; therefore, this technique must be used with some degree of caution.
T.S.'s	FFS Reset Switches	with each set of FFS indicators in TC5	Specify number to which FFS registers will be reset if FFS reset is called for.
T.S.'s	PC Reset Switches	TC5	Specify number to which PC will be reset on START OVER or EXAMINE.

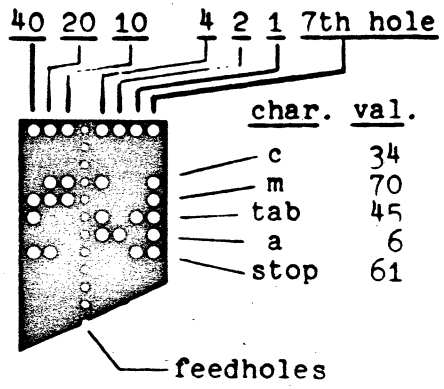
TABLE 1

THE FLEXOWRITER CODE

Alphanumerical Sequence

Low. Case	Up. Case	Code Value	Low. Case	Up. Case	Code Value
a	A	6	0	°	76
b	B	62	1	¹	25
c	C	34	2	²	17
d	D	22	3	³	7
e	E	2	4	⁴	13
f	F	32	5	⁵	23
g	G	64	6	⁶	33
h	H	50	7	⁷	27
i	I	14	8	⁸	3
j	J	26	9	⁹	66
k	K	36	+	/	15
l	L	44	-	_	35
m	M	70	.	)	21
n	N	30	,	(	31
o	O	60	=	:	11
p	P	54	|	_	5
q	Q	56	space bar		10
r	R	24	carr. ret.		51
s	S	12	tab.		45
t	T	40	up. case		71
u	U	16	low. case		75
v	V	74	nullify		77
w	W	46	stop code		61
x	X	72	back space		43
y	Y	52	color sft.		20
z	Z	42			

The values of such characters are equal to the sum of the weights appearing at the head of each column in which a hole is punched. The seventh hole, which comprises the right-most column in the sketch, is used only for control purposes. The seventh hole must be punched as part of each character which is to be read in by the computer.



Coded Value Sequence

Code Val.	Low. Case	Up. Case	Code Val.	Low. Case	Up. Case
2	e	E	35	-	-
3	8	⁸	36	k	K
5	|		40	t	T
6	a	A	42	z	Z
7	3	³	43	back space	
10	space bar		44	l	L
11	=	:	45	tabulation	
12	s	S	46	w	W
13	4	⁴	50	h	H
14	1	I	51	carr. ret.	
15	+	/	52	y	Y
16	u	U	54	p	P
17	2	²	56	q	Q
20	color shift		60	o	O
21	.	)	61	stop code	
22	d	D	62	b	B
23	5	⁵	64	g	G
24	r	R	66	9	⁹
25	1	¹	70	m	M
26	j	J	71	up. case	
27	7	⁷	72	x	X
30	n	N	74	v	V
31	,	(	75	low. case	
32	f	F	76	0	°
33	6	⁶	77	nullify	
34	c	C			

TABLE 2

THE "FL" FLEXOWRITER CODE

Alphanumerical Sequence

Lower Case	Upper Case	Character 123456	Decimal Value	Octal Value
a	A	000110	6	6
b	B	110010	50	62
c	C	011100	28	34
d	D	010010	18	22
e	E	000010	2	2
f	F	011010	26	32
g	G	110100	52	64
h	H	101000	40	50
i	I	001100	12	14
j	J	010110	22	26
k	K	011110	30	36
l	L	100100	36	44
m	M	111000	56	70
n	N	011000	24	30
o	O	110000	48	60
p	P	101100	44	54
q	Q	101110	46	56
r	R	010100	20	24
s	S	001010	10	12
t	T	100000	32	40
u	U	001110	14	16
v	V	111100	60	74
w	W	100110	38	46
x	X	111010	58	72
y	Y	101010	42	52
z	Z	100010	34	42
0	0	111110	62	76
1	1	010101	21	25
2	2	001111	15	17
3	3	000111	7	7
4	4	001011	11	13
5	5	010011	19	23
6	6	011011	27	33
7	7	010111	23	27
8	8	000011	3	3
9	9	110110	54	66
|	_	000101	5	5
space bar		001000	8	10
=	.	001001	9	11
+	/	001101	13	15
color change		010000	16	20
.	)	010001	17	21
,	(	011001	25	31
-	"	011101	29	35
back space		100011	35	43
tabulation		100101	37	45
carr. return		101001	41	51
stop		110001	49	61
upper case		111001	57	71
lower case		111101	61	75
nullify		111111	63	77

TABLE 3

THE "FL" FLEXOWRITER CODE

Binary Numerical Sequence

Decimal Value	Octal Value	Character 123456	Lower Case	Upper Case	Decimal Value	Octal Value	Character 123456	Lower Case	Upper Case
0	0	000000	not used		32	40	100000	t	T
1	1	000001	not used		33	41	100001	not used	
2	2	000010	e	E	34	42	100010	z	Z
3	3	000011	8	8	35	43	100011	back space	
4	4	000100	not used		36	44	100100	l	L
5	5	000101	|	_	37	45	100101	tabulation	
6	6	000110	a	A	38	46	100110	w	W
7	7	000111	3	3	39	47	100111	not used	
8	10	001000	space bar		40	50	101000	h	H
9	11	001001	=	:	41	51	101001	carr. return	
10	12	001010	s	S	42	52	101010	y	Y
11	13	001011	4	4	43	53	101011	not used	
12	14	001100	i	I	44	54	101100	p	P
13	15	001101	+	/	45	55	101101	not used	
14	16	001110	u	U	46	56	101110	q	Q
15	17	001111	2	2	47	57	101111	not used	
16	20	010000	color change		48	60	110000	o	0
17	21	010001	.	)	49	61	110001	stop	
18	22	010010	d	D	50	62	110010	b	B
19	23	010011	5	5	51	63	110011	not used	
20	24	010100	r	R	52	64	110100	g	G
21	25	010101	1	1	53	65	110101	not used	
22	26	010110	j	J	54	66	110110	9	9
23	27	010111	7	7	55	67	110111	not used	
24	30	011000	n	N	56	70	111000	m	M
25	31	011001	,	(	57	71	111001	upper case	
26	32	011010	f	F	58	72	111010	x	X
27	33	011011	6	6	59	73	111011	not used	
28	34	011100	c	C	60	74	111100	v	V
29	35	011101	-	—	61	75	111101	lower case	
30	36	011110	k	K	62	76	111110	0	0
31	37	011111	not used		63	77	111111	nullify	

DISTRIBUTION LIST

Lincoln

Division II

C. R. Wieser
S. H. Dodd

Group 21

P. Rosen
O. V. Fortier
L. C. Wilber
P. J. Harris
D. A. Spaeth
F. E. Heart (5)
R. M. Martinson
B. J. Benn
A. E. Budd
W. Crowther
J. E. Drinan
A. A. Mathiasen
C. J. Muhle (2)
H. D. Neumann
G. D. Stephenson
R. J. Saliga

Group 22

V. A. Nedzel (5)
J. F. Nolan
T. C. Bartee
M. M. Daley
W. F. Holst
W. Z. Lemnios
W. Lone, Jr.
D. L. Lovenvirth
E. J. Magee
M. C. Rubin
R. W. Sittler
C. W. Uskavitch
N. L. Daggett
P. R. Bagley

Group 25

H. Sherman (3)

Group 28

J. A. Arnow
H. C. Peterson

Division III

Group 31

H. G. Weiss

Group 32

E. W. Bivans

Group 33

J. H. Chisholm

Group 34

O. G. Selfridge

Group 35

D. T. Stevenson

Group 36

W. E. Morrow

Group 37

J. W. Meyer

Group 38

F. C. Frick
C. Forgie

Group 39

E. W. Blaisdell

Group 310

H. C. Gatos

Group 312

D. E. Dustin
L. J. Sullivan
H. L. Kasnitz
P. Willmann

Group 314

P. B. Sebring

Division IV

Group 42

F. R. Naka

Group 47

T. C. Bazemore

Group 48

R. C. Butman

Division VI

R. R. Everett

Group 60

A. M. Falcione
A. R. Curtiss (25)

Group 61

K. E. McVicar

Group 63

W. N. Papian
W. A. Clark, Jr.
J. M. Frankovich
J. W. Forgie
J. T. Gilmore, Jr.
R. H. Gould

Group 64

E. S. Rich

Group 66

D. R. Israel (10)
D. L. Bailey

Group 67

C. A. Zraket

DISTRIBUTION LIST (Continued)

Other

R. K. Bennett (2 copies)
Data Processing
8 Pelham Road
Natick, Massachusetts

MIT - Cambridge

	Room No.	
M. Ayvazian	41-219	
W. Blackstock	41-219	
E. S. Criscione	41-219	
N. P. Hobbs	41-219	
E. A. Witmer	41-219	
M. N. Springer	41-203	
S. Namyet	1-271	
J. E. Cunningham	26-340	
E. J. Campbell	26-352	
G. W. Hughes	10-215	
W. A. Youngblood	26-343	
D. T. Ross	32-100	(10 copies)
L. C. Allen	6-320	
M. M. Saffren	6-320	
J. H. Wood	6-320	
R. E. Watson	6-320	
D. N. Arden	26-269	(3 copies)
C. J. Maletskos	NW-12	
L. J. Gitten	10-339	
E. A. Kelley	24-510	
R. W. Wylie	24-506	
B. Shorr	24-506	
P. A. Coggio	NW-12	
J. Belanger	W2-218	

TABLE 4

WHIRLWIND I INSTRUCTION CODE

The complete WWI instruction code is given below in tabular form.

The notations used in the table, together with their meanings, are as follows:

—— = remains unchanged
IOR = In-Out Register
PC = Program Counter
AC = Accumulator, AC(i) = digit i of AC, $0 \leq i \leq 15$
HR = B-Register, HR(i) = digit i of HR, $0 \leq i \leq 15$
AR = A-Register, AR(i) = digit i of AR, $0 \leq i \leq 15$
ρ = Round-off from HR; if HR(0) = 1, $\rho = 2^{-15}$; if HR(0) = 0, $\rho = 0$
CM = Core memory

x = Address of a storage register $0 \leq x \leq 2047$
n = Positive integer $0 \leq n \leq 511$ mod 32
SAM = Special Add Memory
C( ) = Original contents of register ( )
C_i( ) = Original contents of digit i of register ( )
F{ } = Fractional part of the quantity in { }
I{ } = Integral part of the quantity in { }
---> = Becomes stored in
y = Address of instruction executed

x(i) = Digit i of register x, $0 \leq i \leq 15$
x(k-m) = Digits k thru m of register x, $0 \leq k < m \leq 15$
AC+HR = The composite 32 digit register (including sign) composed of the AC and HR taken in that order
(AC+HR)(i) = Digit i of (AC+HR), $0 \leq i \leq 31$
$\oplus$ = Boolean "exclusive or" operator
$\otimes$ = Boolean "and" operator

Instruction	Function	Binary Code	Dec. Equiv.	AC	Final contents of HR*	Final contents of AR	SAM	x	Comments	Time
si pqr	select in-out unit or stop the computer	00000	0	——	——	——	——	——	The unit selected is designated by the digits pqr, and is started. si 0 will stop the computer. si 1 is a "conditional" stop. Program alarm possible if si selects Magnetic or Photo Electric Tape Reader without necessary rd's. See D-55565-3.	30 μsec.
		00001	1	——	——	——	——	——	Illegal instruction	
bi x	block transfer in (n words to CM)	00010	2	x+n	——	x	——	first word of block	$+n \cdot 2^{-15}$ must be stored in AC at the time the computer executes the bi. If n=0 one word will be read but not transferred. It is simply discarded.	For drum 8 msec. average and 16 msec. max. for first word. 16 μsec. for each additional word.
rd x	read	00011	3	C(IOR)	——	C(IOR)	——	——	After word is transferred from IOR to AC, the IOR is cleared. The address of rd has no significance.	15 μsec.
bo x	block transfer out (n words from CM)	00100	4	x+n	——	x	——	——	$+n \cdot 2^{-15}$ must be stored in AC. If n=0, no recording will take place.	same as for bi.
rc x	record	00101	5	——	——	——	——	——	If rc is used as a display instruction, the IOR is cleared.	22 μsec.
sd x	sum digits	00110	6	C_i(AC) $\oplus$ C_i (x) ---> AC(i) i = 0, ..., 15	——	C(x)	clear	——	Adds digits without carry $\oplus$ | 0 1 0 | 0 1 1 | 1 0	22 μsec.
cf pqr	change fields	00111	7	——	——	y + 1 (digits 5-15) If dig. 7 is 1 GP. SW.-10-15 If dig. 6 is 1	——	——	With one in digit 8, y(10, 11, 12)--->Group A's field selection switch. With one in digit 9, y(13, 14, 15)--->Group B's field selection switch. With one in digit 7, AC(5-15)--->PC., 0 ---> AC Group A and/or B cannot be changed to refer to same field.** With one in dig. 6, GP. Switches ----> AC 10-15.	15 μsec.
ts x	transfer to storage	01000	8	——	——	——	——	C(AC)		22 μsec.
td x	transfer digits	01001	9	——	——	——	——	x(0-4) unchanged x(5-15) = AC(5-15)		29 μsec.
ta x	transfer address	01010	10	——	——	——	——	x(0-4) unchanged x(5-15) = AR(5-15)	ta normally follows an sp, cp, sf, or ao	29 μsec.
ck x	check	01011	11	——	——	——	clear	——	Normal Mode: Computer stops on "check-register alarm" if C(AC) ≠ C(x). (Note that +0 ≠ -0.) Special Mode: PC indexed by one if C(AC) ≠ C(x). (No "check-register alarm.")	22 μsec.
ab x	add HR	01100	12	C(HR) + C(x)	——	C(x)	——	C(HR) + C(x)	Possible Arith. Overflow Alarm if $\lvert C(x) + C(HR) \rvert \geq 1$ dm x or clh 16 puts C(AC) into HR	29 μsec.
ex x	exchange	01101	13	C(x)	——	C(x)	——	C(AC)	ex 0 will clear AC without clearing HR	29 μsec.
cp x	conditional transfer control (conditional program)	01110	14	——	——	y + 1 (digits 5-15)	——	——	If C(AC) ≥ +0 proceed to next instruction. If C(AC) ≤ -0, execute sp x. y is location of cp x.	15 μsec.
sp x	transfer control (sub-program	01111	15	——	——	y + 1 (digits 5-15)	——	——	Take next instruction from register x. x--->PC. y is location of sp x.	15 μsec.
ca x	clear and add	10000	16	$C(x) + C(SAM)2^{-15}$	clear	C(x)	clear	——	Possible Arith. Overflow Alarm if $\lvert C(x) + C(SAM)2^{-15} \rvert = 1$	22 μsec.
cs x	clear and subtract	10001	17	$-C(x) + C(SAM)2^{-15}$	clear	C(x)	clear	——	Possible Arith. Overflow Alarm if $\lvert -C(x) + C(SAM)2^{-15} \rvert = 1$	22 μsec.
ad x	add	10010	18	C(AC) + C(x)	——	C(x)	clear	——	Possible Arith. Overflow Alarm if $\lvert C(AC) + C(x) \rvert \geq 1$	22 μsec.
su x	subtract	10011	19	C(AC) - C(x)	——	C(x)	clear	——	Possible Arith. Overflow Alarm if $\lvert C(AC) - C(x) \rvert \geq 1$	22 μsec.
cm x	clear and add magnitude	10100	20	$\lvert C(x) \rvert + C(SAM)2^{-15}$	clear	$\lvert C(x) \rvert$	clear	——	Possible Arith. Overflow Alarm if $\lvert C(x) \rvert + C(SAM)2^{-15} = 1$	22 μsec.
sa x	special add	10101	21	$F\{C(AC) + C(x)\}$	——	C(x)	I{C(AC)+ C(x)}; +1 or 0	——	Sign of SAM determined by sign of overflow. Previous contents of SAM cleared without alarm.	26 μsec.
ao x	add one	10110	22	$C(x) + (1x2^{-15})$	——	C(x)	clear	$C(x) + (1x2^{-15})$	Possible Arith. Overflow Alarm if $C(x) + (1x2^{-15}) = 1$	29 μsec.
dm x	difference of magnitudes	10111	23	$\lvert C(AC) \rvert - \lvert C(x) \rvert$	C(AC)	$\lvert C(x) \rvert$	clear	——	If $\lvert C(AC) \rvert = \lvert C(x) \rvert$ result is -0	22 μsec.
mr x	multiply and round-off	11000	24	$C(AC) \cdot C(x) + \rho$	clear	$\lvert C(x) \rvert$	clear	——	Sign of AC is determined by sign of product	34-41 μsec.
mh x	multiply and hold	11001	25	C(AC) · C(x) (digits 1-15)	$\lvert C(AC) \cdot C(x) \rvert$ (digits 16-30) HR(0) - HR(14) HR 15 = 0	$\lvert C(x) \rvert$	clear	——	Sign obtained same as for mr. Result in (AC + HR) is a double register product.	Same as for mr
dv x	divide	11010	26	± 0 (sign of quotient)	$\left\lvert \frac{C(AC)}{C(x)} \right\rvert$ (16 digits)	$\lvert C(x) \rvert$	clear	——	Divide Error Alarm if $\lvert C(AC) \rvert > \lvert C(x) \rvert$. Arith. Overflow Alarm if $\lvert C(AC) \rvert = \lvert C(x) \rvert \neq 0$ and dv x is followed by slr 15. If C(AC) = C(x) = 0, the quotient is 0.	71 μsec.
slr n	shift left and round-off (n places)	11011 0	27	$F\{C(AC + HR)2^n\} + \rho$ (n taken mod 32)	clear	——	clear	——	Possible Arith. Overflow Alarm if $\lvert F\{C(AC + HR)2^n\} + \rho \rvert = 1$. The sign digit is not shifted. Digits shifted out of AC 1 are lost. Negative numbers are complemented in AC before shifting and after rounding off. Digit 6 of slr n must be zero.	15 + .8n μsec.
slh n	shift left and hold (n places)	11011 1	27	$F\{C(AC + HR)2^n\}$ (n taken mod 32)	$F\{\lvert C(AC + HR)2^n \rvert\}$ (digits 16-(31-n)) HR(j)--->0 j=16-n, ...,15 n>0	——	clear	——	The sign digit is not shifted. Negative numbers are complemented in AC before and after the shift. Digit 6 of slh n must be a one.	Same as for slr
srr n	shift right and round-off	11100 0	28	$C(AC)2^{-n} + \rho$ (n taken mod 32)	clear	——	clear	——	Possible Arith. Overflow Alarm on srr 0 (this instruction simply causes roundoff and clears HR). The sign digit is not shifted. Negative numbers are complemented in AC before shifting and after rounding off. Digit 6 of srr n must be a zero.	Same as for slr
srh n	shift right and hold	11100 1	28	$C(AC)2^{-n}$ (n taken mod 32)	$\lvert C(AC + HR)2^{-n} \rvert$ (digits 16-31)	——	clear	——	The sign digit is not shifted. Negative numbers are complemented in AC before and after the shift. Digit 6 of srh n must be a one.	Same as for slr
sf x	scale factor	11101	29	$C(AC + HR)2^n$ (digits 1-15)	$C(AC + HR)2^n$ (digits 16-(31-n)) HR(j)--->0 j=16-n, ...,15 n 0	$n \cdot 2^{-15}$	clear	$n \cdot 2^{-15}$ x(5-15)	x(0-4) unaffected. n is such that $\frac{1}{2} \leq \lvert C(AC + HR)\, 2^n \rvert < 1$; if C(AC + HR) = 0, then n = 33. Negative numbers are complemented in AC before and after the multiplication.	30-78 μsec.
clc n	cycle left and clear	11110 0	30	$(AC + HR)(n + i)_{32}$ ---> AC(i) i=0, ...,15	clear	——	——	——	Sign digit is shifted with all other digits. Digits shifted left out of AC 0 are carried around into HR 15. No roundoff. No complementing of AC either before or after the shift. Digit 6 of clc n must be a zero. Instruction clc 0 clears HR without affecting AC.	Same as for slr
clh n	cycle left and hold	11110 1	30	$(AC + HR)(n + i)_{32}$ ---> AC(i) i=0, ...,15	$(AC + HR)(n + i + 16)_{32}$ ---> HR(i) i=0, ..., 15	——	——	——	Sign digit is shifted with all other digits. Digits shifted left out of AC 0 are carried around into HR 15. No complementing of AC either before or after the shift. Digit 6 of clh n must be a one. Instruction clh 0 does nothing.	Same as for slr
md x	multiply digits	11111	31	C_i(AC) $\otimes$ C_i (x) ---> AC(i) i=0, ...,15	——	-(Final AC)	——	——	Multiplies digits with no carry $\otimes$ | 0 1 0 | 0 0 1 | 0 1	22 μsec.

* In operations mr, mh, dv, slr, srr, srh, sf, the C(HR) is assumed to be the magnitude of the least significant part of AC + HR. For the ab and dm operations, the HR is treated just as any storage register.

**Core memory now contains 6 fields numbered 0 through 5. At any given time only two of these fields are active. These fields are designated Group A $(40 \text{ through } 1777)_8$ and Group B $(2000 \text{ through } 3777)_8$. Field duplication alarm will occur if same field in both groups. Illegal field alarm will occur if field 6 or 7 specified.

UNCLASSIFIED

UNCLASSIFIED

TABLE 5

DESIGNATIONS OF IN-OUT EQUIPMENT, WWI
(si addresses for all units)

UNIT	MODE OF OPERATION	si ADDRESS			
		MT0	MT1	MT2	MT3
MAGNETIC TAPE (BLOCK OPERATION°)	RE-RECORD, FORWARD	100(o) 64(D)	110(o) 72(D)	120(o) 80(D)	130(o) 88(D)
	RE-RECORD, REVERSE	101(o) 65(D)	111(o) 73(D)	121(o) 81(D)	131(o) 89(D)
	READ, FORWARD	102(o) 66(D)	112(o) 74(D)	122(o) 82(D)	132(o) 90(D)
	READ, REVERSE	103(o) 67(D)	113(o) 75(D)	123(o) 83(D)	133(o) 91(D)
	STOP AFTER RECORD, FORWARD	104(o) 68(D)	114(o) 76(D)	124(o) 84(D)	134(o) 92(D)
	STOP AFTER RECORD, REVERSE	105(o) 69(D)	115(o) 77(D)	125(o) 85(D)	135(o) 93(D)
	RECORD, FORWARD	106(o) 70(D)	116(o) 78(D)	126(o) 86(D)	136(o) 94(D)
	RECORD, REVERSE	107(o) 71(D)	117(o) 79(D)	127(o) 87(D)	137(o) 95(D)
	RECORD**, FORWARD FOR PRINT-OUT	146(o) 102(D)	156(o) 110(D)	166(o) 118(D)	176(o) 126(D)
	RECORD**, REVERSE FOR PRINT-OUT	147(o) 103(D)	157(o) 111(D)	167(o) 119(D)	177(o) 127(D)

		PTR0 (MECH. READER)	PTR1 FERRANTI
PAPER TAPE READERS (PTR)	READ, NORMAL°° L x L°	200(o) 128(D)	-----
	READ AUTO. ASSEMBLY°° L x L°	202(o) 130(D)	-----
	READ, NORMAL°° BLOCK°	-----	211(o) 137(D)
	READ, AUTO. ASSEMBLY °° BLOCK °	-----	213(o) 139(D)

PUNCH	RECORD, NORMAL°° 7TH HOLE SUPPRESSED	204(o) 132(D)
	RECORD, NORMAL°° 7TH HOLE PUNCHED	205(o) 133(D)
	RECORD AUTO.ASSY.°° 7TH HOLE SUPPRESSED	206(o) 134(D)
	RECORD AUTO. ASSY.°° 7TH HOLE PUNCHED	207(o) 135(D)

		PRINTER 2 (COMPUTER RM.)	PRINTER 3 (NOT IN USE)
PRINTERS	RECORD	225(o) 149(D)	235(o) 157(D)
	SELECT ANELEX PRINTER, SAME PRINTER GP. CONT.	244(o)	
	SELECT ANELEX PRINTER, CLEAR PRINTER GP. CONT.	245(o)	

		REGISTERS 0 & 1 (ACTIVATE)	REGISTERS 2 TO 31 (INSERTION)
INTERVENTION REGISTERS	READ	300 & 301(o) 192 & 193(D)	302 TO 337(o) 194 TO 223(D)

MITE BUFFER STORAGE	SET ALL TIMING SYNC.	403 (O)
	CLEAR ALL FF's	401(o) 257(D)
	COMPLEMENT ALL COUNTERS AND REGISTERS	433(o) 283(D)
	SET ALL REF. MEMORIES	413(o) 267(D)
	SET ALL DATA MEMORIES	423(o) 275(D)
	SET "RECORD WHILE READING" SYNC.	411(o) 265(D)
	SET RECORD SYNC. (SUCCEEDING SLOT 7 PULSE SETS ALL MEMORIES)	421(o) 273(D)
	SET RECORD SYNC. (SUCCEEDING SLOT 7 PULSE SETS ALL DATA SYNCS.)	431(o) 281(D)
TELETYPE	RECORD TT BUFFER REG.	402(o)

REAL-TIME CLOCK (TIMING REGISTER)	FINE COUNT	005
	COARSE COUNT	007
	CLEAR	011(o) 009(D)
	COMPLEMENT	012(o) 010(D)

INDICATOR LIGHT REGISTERS	RECORD	REGISTERS 0 TO 7 510 TO 517 (o) 328 TO 335 (D)

DISPLAY SCOPES	DISPLAY POINTS	0600 TO 0677(o) 384 TO 447(D)
	DISPLAY VECTORS	1600 TO 1677(o) 896 TO 959(D)
	DISPLAY CHARACTERS	2600 TO 2677, 1408 TO 1471, 3600 TO 3677(o) 1920 TO 1983(D)
	EXP. CHAR. DISPLAY	14(o) 12(D)
	RELEASE EXP. CHAR. DISPLAY	15(o) 13(D)

DOC SYSTEM	DOC SEND SYSTEM "A"	412 (o)
	DOC SEND SYSTEM "B"	422 (o)

UNIT	MODE OF OPERATION	si ADDRESS
AUXILIARY DRUM*	READ, NO CHANGE IN ADDRESS SEQUENCE	700(o) 448(D)
	READ, SELECT NEW INITIAL ANGULAR POSITION	701(o) 449(D)
	READ, SELECT NEW GROUP	702(o) 450(D)
	READ, SELECT NEW INITIAL ADDRESS (GROUP AND ANGULAR POSITION)	703(o) 451(D)
	RECORD, NO CHANGE IN ADDRESS SEQUENCE	704(o) 452(D)
	RECORD, SELECT NEW INITIAL ANGULAR POSITION	705(o) 453(D)
	RECORD, SELECT NEW GROUP	706(o) 454(D)
	RECORD, SELECT NEW INITIAL ADDRESS (GROUP AND ANGULAR POSITION)	707(o) 455(D)
BUFFER DRUM*	READ, NO CHANGE IN ADDRESS SEQUENCE	710(o) 456(D)
	READ, SELECT NEW INITIAL ANGULAR POSITION	711(o) 457(D)
	READ, SELECT NEW GROUP	712(o) 458(D)
	READ, SELECT NEW INITIAL ADDRESS (GROUP AND ANGULAR POSITION)	713(o) 459(D)
	RECORD, NO CHANGE IN ADDRESS SEQUENCE	714(o) 460(D)
	RECORD, SELECT NEW INITIAL ANGULAR POSITION	715(o) 461(D)
	RECORD, SELECT NEW GROUP	716(o) 462(D)
	RECORD, SELECT NEW INITIAL ADDRESS (GROUP AND ANGULAR POSITION)	717(o) 463(D)
	SWITCH FIELDS	734(o) 476(D)
CAMERA	***INDEX CCDC CAMERA INDEX CAMERA	024(o) 20(D) 004
HEIGHT REQUEST	RECORD	006
	SWITCH TO PB	000
	CONDITION SWITCH TO PB	001
	UNCONDITIONAL FF REGISTER RESET	010(o) 008(D)
PROGRAM MARGINAL CHECKING	EXCURSION INDICATION	002
	LINE SELECTION	003
	RELEASE STEP MODE	013(o) 011(D)
CORE MEMORY	CLEAR	017(o) 015(D)
IN-OUT CHECK	ACTIVATE REGISTERS TO ONES	504(o) 324(D)
	READ INDICATOR LIGHT REGISTERS	501(o) 321(D)
	ACTIVATE ALL LIGHT GUNS	505(o) 325(D)
	CLEAR ALL INTERVENTION REGISTERS AND DISPLAY SWITCHES	500(o) 320(D)
	SET ALL INTERVENTION REGISTERS AND DISPLAY SWITCHES	502(o) 322(D)
	CHANGE ROOM 222 + 5 VOLTS TO +10 VOLTS	506(o) 326(D)
	CHANGE ROOM 222 + 5 VOLTS TO GROUND	507(o) 327(D)
	RELEASE ALL TEST RELAYS	503(o) 323(D)
CROSSTEL. OUTPUT CODER	SEND REFERENCE	420(o) 272(D)
	SEND ONE	410(o) 264(D)
	SEND ZERO	400(o) 256(D)
	RAYDIST	016(o) 014(D)

*ADD 1000(o), 512(D), TO THE DRUM si ADDRESS TO ENABLE THE SAR END CARRY TO ADD ONE TO THE GSR.

**PRINT OUT IS ONLY USEFUL WITH MT1, MT2, AND MT3.

o IN L x L OPERATION THE UNIT STOPS AFTER EACH READ. IN BLOCK OPERATION THE UNIT CONTINUES TO SUPPLY INFORMATION TO THE COMPUTER UNTIL THE UNIT IS STOPPED BY ANOTHER si COMMAND.

oo IN THE NORMAL MODE OF OPERATION ONE LINE IS READ IN ON EACH IO OPERATION. IN THE AUTOMATIC ASSEMBLY MODE OF OPERATION THREE LINES ARE READ AND AUTOMATICALLY ASSEMBLED INTO A COMPUTER WORD.

*** INDEX COMPUTER CAMERA (SPECIAL MOD- NO 500 MILLISECONDS COMPUTER DELAY ALLOWED)

UNCLASSIFIED

UNCLASSIFIED