Bendix

G-20

Peripheral Equipment Machine Language

PERIPHERAL EQUIPMENT

MACHINE LANGUAGE

CONTENTS

CHAPTER THREE – THE PC–10 PRINTER AND CARD COUPLER AND ASSOCIATED EQUIPMENT

CHAPTER FOUR – THE CC–10 CONTROL CONSOLE

CHAPTER ONE
G-20 COMMUNICATION SYSTEM

This manual describes the basic methods of programming the Bendix G-20 Computer for use with the Control Console CC-10, the Printer and Card Coupler PC-10, and the Magnetic Tape Module MT-10. The reader should be familiar with the material in the publication "Central Processor Machine Language" before using this manual for machine language input/output programming.

The Bendix G-20 Communication System is a transmit-answer system. Conversations between units take place via communication lines which physically connect main terminal units to each other in the system. Every communication on a communication line is a series of individual character transmissions, such that one character is sent from one unit and a reply is given to it from the other unit in the communication, for every character in a transmission.

The G-20 Communication System is a three-element system, consisting of a controller, transmitter and receiver for every transmission. The controller notifies the unit that is to be the transmitter and sets it for transmission. The controller notifies the unit that is to be the receiver and sets it for reception. The controller then initiates the transmission, which is carried out between transmitter and receiver. The controller may also be one of the transmitting or receiving units. For example, the central processor, as a controller, can initiate a communication between a magnetic tape unit and a control buffer; or the control buffer, as both receiver and controller, can initiate, and continue, a magnetic tape to control buffer communication.

1. PERIPHERAL EQUIPMENT OPERATING STATES

Certain operating states have been defined for the peripheral equipment in the G-20 Communication System. The names of the operating states are mnemonically significant to the operator and programmer.

To make the discussion of the operating states more clear, some of the communication line instructions will be introduced.

When a piece of peripheral equipment is waiting for instructions, it is in the STANDBY state. The unit in STANDBY examines and then rejects all call signals it hears until it receives its own unique call with correct parity.

When a piece of equipment receives its own call, it enters the CALLED state. In the CALLED state a unit can receive and act upon any line instruction which is meaningful to it.

A unit is in the INSTRUCTED state when it has been given the complete set of instructions required to establish a data transfer but has not yet been told by the control unit to begin transferring data. A unit which has been set up to receive will, upon receiving the start data transfer instruction called SDT, enter the MESSAGE state and send a request signal called REQ. A unit which has been set up to transmit will, upon hearing SDT, enter the MESSAGE state but will give no response to the SDT. A unit in the INSTRUCTED state, upon hearing its own call, will answer with the reply GRN and return to the CALLED state. No commands except SDT and the unit's own call are meaningful to a unit in the INSTRUCTED state.

When a unit is executing an operation other than a data transfer, or is interlocked, it is in the BUSY state. BUSY has two sub-states. Normally the BUSY unit is BUSY-QUIET where it can hear only its own call, which it will answer and then go to BUSY-ALERT. In BUSY-ALERT the unit will answer queries but will not execute commands except the command to go out of service called OUT. Upon hearing a call directed to another unit, the BUSY-ALERT unit will return to BUSY-QUIET without answering. Upon completing an operation, a BUSY-QUIET unit enters the STANDBY state and sends an interrupt request. A BUSY-ALERT unit upon completing an operation, enters the CALLED state and sends an interrupt request.

A unit engaged in a block communication is in the MESSAGE state. All other units are forbidden to use the data lines during this time. When the message has been completed, the transmitter sends the instruction END or ERR and returns to STANDBY. The END or ERR signal returns the receiver to STANDBY.

If the G-20 is the receiver it may terminate the block communication by sending END or ERR. When a unit is OUT of Service, a unit is effectively disconnected, and can only be put back on line by manual operation of the ON-LINE button. Such action causes the unit to go ON-LINE in the STANDBY state. Peripheral units may be called and directed by command to go OUT of service.

2. INTERRUPT SERVICE ROUTINE

During input and output communications in the G-20 system, the Central Processor must be informed of various conditions of the peripheral equipment and, therefore, computation must be interrupted. These input/output interrupt requests function like the internal overflow interrupt and do not need corresponding enabling bits placed in the Enable register.

An Interrupt Service Routine, called ISR, must be written to process these interrupts from the communication system. This routine is usually a part of the overall ISR written for the processing of internal interrupts.

The function of the ISR is simply to check the interrupt request positions in the interrupt request register and to transfer to the appropriate switch location of the exit switch locations. For instance, when a card is being read it must be sent from the hopper to the read brushes and then read. In the G-20 Communication System, line instructions are transmitted from the Central Processor to the card reader to move the card to the read brushes. While the card is being positioned, the Central Processor resumes computation. When the card is in position to be read, an interrupt request is automatically sent to the Central Processor. The requested interrupt halts computation and the ISR will determine which request position contains the interrupt and will transfer to the PC-10 transfer switch position which must contain the proper exit location. When the card is read, the ISR must also be able to return to the first computational program which was being processed before the interrupt.

3. INTERRUPT SERVICE ROUTINE FOR EXAMPLES

Many general samples as well as specific examples of input/output programs are presented to aid the understanding of the G-20 Communication System. A simplified Interrupt Service Routine is assumed to be stored in memory for use with all of the samples and examples. This ISR will determine which piece of terminal equipment sent the interrupt request and will leave all other interrupt processing to the individual routines. This procedure will give the reader a better understanding of what interrupt processing is necessary for each specific piece of equipment.

The ISR used in the samples and examples is a programmed switch. The pieces of peripheral equipment are assigned the following Interrupt Request register entries:

Bit Number	Equipment	Type of Interrupt
4	CC-10	Type-In Button
3	CC-10	All Others
2	PC-10	All
1	MT-10	All

When an interrupt occurs, the ISR is entered. The ISR determines on which line the interrupt request occurred and branches to the appropriate input/output routine by an address previously placed in the ISR exit. For instance, in the examples in this manual, the PC-10 exit on bit number 2, has to allow transfer to three different entry locations -- one for the card reader, one for the card punch and one for the line printer. *Therefore, it is assumed for the samples and examples that the appropriate exit location for each of the positions 1, 2, 3 and 4 is placed in the ISR before each interrupt request.*

The zero bit of the Enable register, called Control, will be turned off by the hardware when an interrupt occurs. *It is assumed that Control is turned on by the ISR before it transfers control to the appropriate input/output routine being considered.*

During the processing of interrupts, while using the PC-10, the ISR resets the "Interrupt Indicator" in the PC-10 before transferring back to the input/output routine.

4. RELATIONSHIP BETWEEN G-20 INTERNAL CHARACTERS AND G-20 WORDS

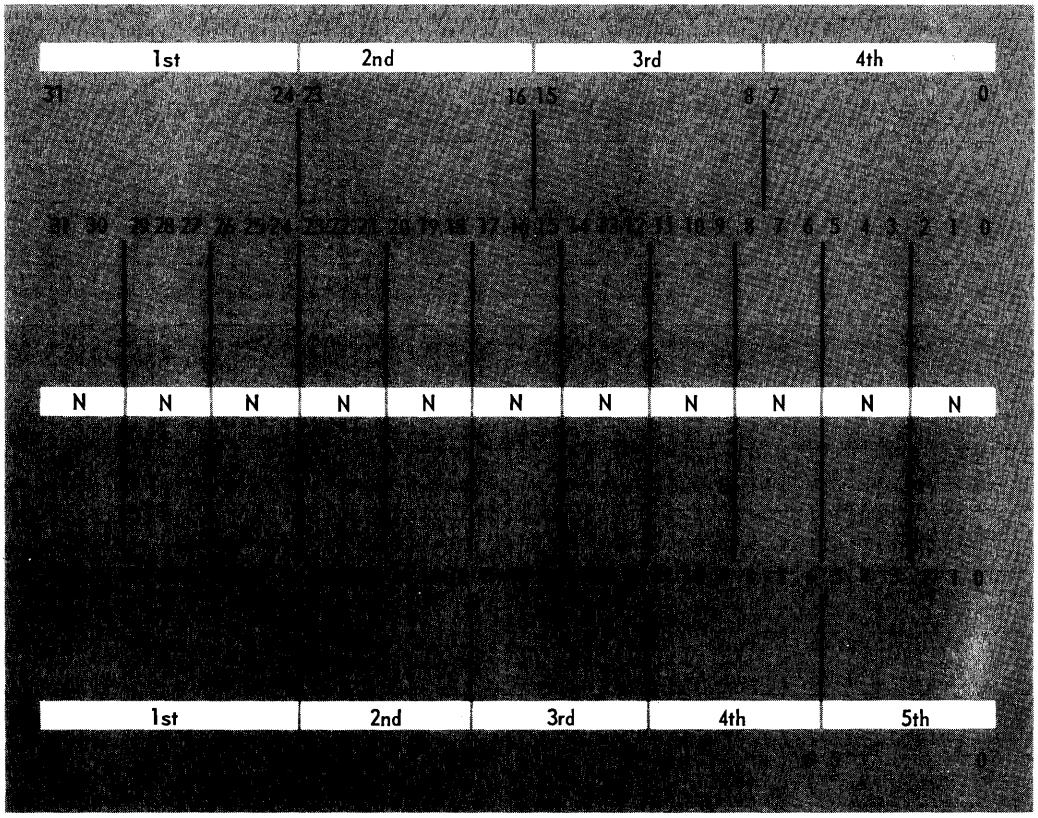
Information is received from and sent to a piece of peripheral equipment on the Communication Line in 6-bit and 8-bit characters. Below is shown the correspondence between four 8-bit characters packed in a G-20 word, five 6-bit characters packed in a word, the 32 bits of a word and the eleven octal digits which correspond to the 32 bits.

An example of the four 8-bit characters 060, 123, 255 and 053 is given below. The octal form of the word would be:

0	6	0	2	4	7	2	6	4	5	3

An example of the five 6-bit characters 060, 035, 052, 053 and 041 is given below. The octal form of the word would be:

0	6	0	3	5	5	2	5	3	4	1



CHAPTER TWO
THE MT-10 MAGNETIC TAPE MODULE

1. MAGNETIC TAPE OPERATIONS

MT-10 Magnetic Tapes are connected to the G-20 Communication System in groups. A tape group consists of a tape control unit (the TC-10) with up to four tape transports attached. The control unit couples either of two communication lines to the tape transports. One of the four transports may be receiving or transmitting information at any one time. All tape transports may be independently searching for specific blocks of information, in either the forward or reverse direction, at any time.

A tape control unit can switch its tape units from one communication line to a second line under control of a program in the Central Processor, or a Control Buffer, or a Data Communicator.

Some general specifications for the MT-10 are:

Read-write speed..... 120,000 8-bit characters/second
Tape Speed
- Read-write ..... 110 inches/second
- Slew ..... 110 inches/second and 220 inches/second
- Rewind..... 220 inches/second

File protection..... Tape cannot be written on without the presence of the "write enable" ring
Tape width ..... One inch

Line instructions to a magnetic tape unit can be transmitted one at a time or with a block transmission command; data is transmitted with block transmission commands. During receipt of line instructions, the tape control unit decodes the command, checks the parity, interrogates its registers, if necessary, sends the required response, and, if a set of instructions has just been completed, sends appropriate signals to the tape unit. By answering queries from the Central Processor or a Control Buffer, the control unit can give information about which transports are idle, which are slewing, rewinding, or interlocked.

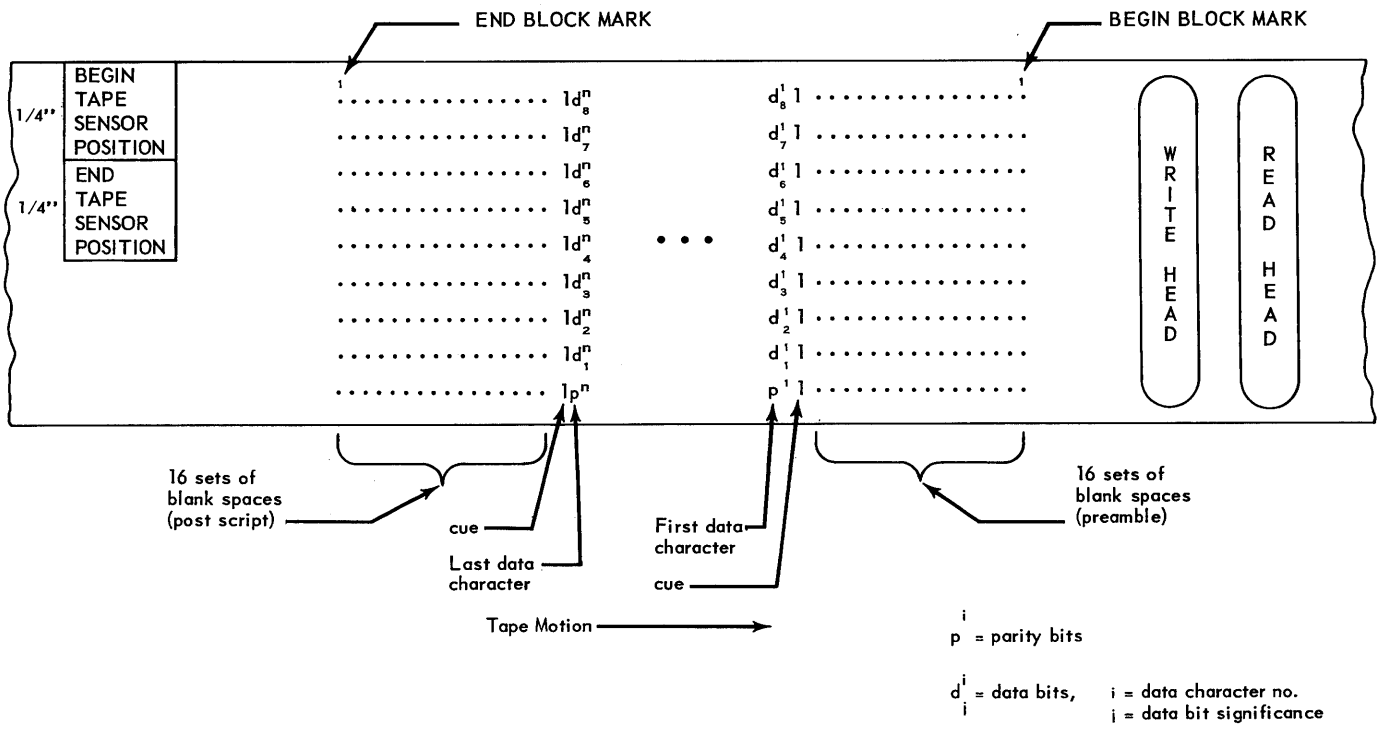
2. TAPE POSITIONING

Non-stop tape positioning to any block address on a reel (within 4095 of the current block's address) is provided in the MT-10 system. The speed of the tape is 110 inches per second for a slew operation of less than 512 blocks. For a slew operation of more than 512 blocks, the speed of the tape is 220 inches per second for the first n-512 blocks and inches per second for the last 512 blocks. (See sample, page 12). The slew instructions must be repeated for each 4095_{10} blocks to be slewed. After the transmission of the necessary line instructions, slew (or search) operations can take place while another tape unit in the same group is writing or reading tape. When transmitting instructions to instruct a tape unit to slew, the instruction character, FWD (forward) or BAC (backward), which indicates the direction of the slew, is followed by two instruction numerics. The first numeric indicates the two most significant octal digits and the second numeric indicates the two least significant octal digits of the four-digit integer which specifies the number of blocks to be slewed. The second numeric must follow the first numeric by not more than 500 microseconds. If the second does NOT come in 500 microseconds, the two least significant digits are set to zero and slewing will proceed. The numerics addressed to two different tape-handlers in the same tape group must be separated by at least 1.5 milliseconds.

3. INFORMATION FORMAT

A Central Processor machine word is written on magnetic tape as four 8-bit characters (one bit in each of eight tracks -- see figure next page). A bit representing even parity is written in a ninth track with each character. A tenth track on the tape carries the block marks which indicate the beginning and end of blocks of data.

MAGNETIC TAPE FORMAT

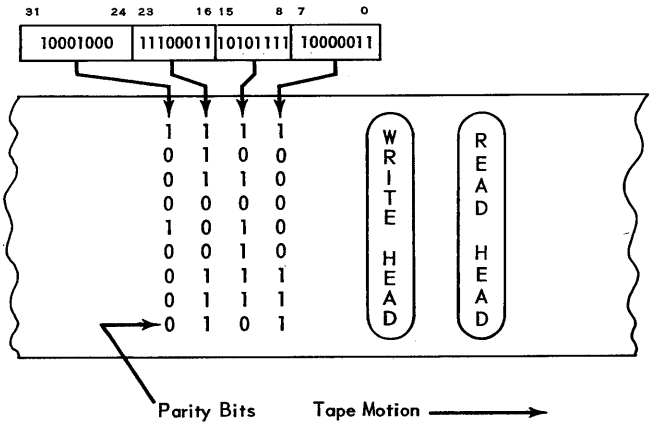


The End of Tape and Beginning of Tape indicators are pieces of one inch by one-quarter inch reflective tape attached to the end and the beginning of the tape. These indicators are sensed by the End of Tape and Beginning of Tape sensors, the condition of which can be ascertained by command.

The figure above shows the form in which information is written on tape (oxide-coated side front).

Data is written on tape in collections called blocks. The length of the blocks (i.e. the number of characters) is arbitrary. These blocks of data are separated by blank sections of tape called "inter-record gaps." Block markers (i.e. bits in the tenth tape track) denote the beginning and ending of each block of data. Block marks are laid down when a magnetic tape is initially written. A "preamble" of 16 blank characters (with blank parity) and a "cue" are automatically written at the beginning of each block on the tape. The "cue" is a "begin data character."

The figure below shows the way information is placed on tape from a Central Processor word. Information is taken from tape and placed in memory the same way.



4. MAGNETIC TAPE WRITING

Before any writing can be accomplished on a tape transport the "write enable" ring must be present on the reel hub.

When the block marks defining the data blocks are present on a magnetic tape, the rewriting of a block on tape is accomplished as follows. The controlling unit calls the specified tape transport, queries the transport for a ready condition, and sends the instructions to the unit to prepare it to receive data. At this time the tape will begin to move.

From 3.5 to 6 milliseconds elapse from the time the final instruction is received until the tape has moved into position to accept data to be written. Within the timing restriction of 3.5 milliseconds the programmer can use the start time for some minor computation or housekeeping. If the "Start Data Transfer" does not reach the tape transport in time, a blank block will be written within the already existing block marks (with the preamble and postscript also.) The data is transferred at about 8.3 microseconds per character. Therefore, a 256 G-20 word block could be written in about 8.3 milliseconds.

When the last character to be written is transmitted to the tape unit, the controlling device (Central Processor or Control Buffer) transmits an END character to terminate the transmission. The tape requires about 11.4 milliseconds to come to a complete stop and become ready for more instructions. When the tape has stopped and is again ready, an interrupt will be sent to the controlling unit. This stop time is also available to the programmer.

If a block length less than or equal to the one that is on the tape is specified, the preamble, then the data, and the postscript (immediately after the data) will be written on the tape. If a block length greater than the one that is on the tape is specified, two independent things will occur: (a) the tape handler will sense the block mark as it passes and stop the tape in its normal position; (b) the right heads will continue to write until the Central Processor sends the end of transmission instruction. The information written past the block mark is not recoverable. This attempt to write more than the available block length will set the Error indicator and the Query Error instruction will tell the programmer of the erroneous writing attempt. If such a block of tape is read, an error indication is transmitted instead of a data character when the end of block mark is encountered; also, an error query will tell the programmer of the previous erroneous writing attempt.

A rewrite attempt must not follow a slew of tape because the read-write heads do not position themselves for a write operation. (It is suggested that a block of tape be read between a slew and a rewrite.)

To write on a tape which has no block marks or data block envelopes, the "initial write" procedure, which is very similar to the rewrite procedure, is used. Instead of the usual instruction to prepare the tape transport to receive data, a special instruction called the "initial write" is used. Since the block marks must be placed on the tape and the blocks must be spaced on the tape, the start time of the "initial write" is longer. From 24.5 to 29.5 milliseconds elapse from the time the final instruction is received until the tape has moved into position to be written on. If the Start Data Transfer command does not reach the tape transport within 24.5 milliseconds, a blank block of zero length is written, complete with block marks, preamble and postscript. After the end of block instruction is received by the tape transport it takes 11.4 seconds for the tape to stop. If an initial write operation is terminated by end or beginning of tape sensors, block marks which cannot be deleted by a delete command are placed on tape there. Initial write should not be attempted if the tape is positioned at the beginning of tape sensor or at the end of tape sensor.

In using the initial write command in a tape preparation routine, where the number of words to be rewritten in general is n, then the initial writing must contain 1.05 times n words or more.

If at any time during a write operation there is a parity error, the parity error indicator is set. This condition can be detected by using the Query Error instruction.

5. MAGNETIC TAPE READING

To read a block of magnetic tape, the controlling unit (the Central Processor or Control Buffer) calls the specified tape transport, queries the transport for a ready condition, sends a forward instruction character, and sends the instruction to the unit to prepare for the transmission of data. At this time the tape will begin to move.

From 7.0 to 9.5 milliseconds elapse from the time the final instruction is received until the tape has moved into position to allow data to be read from it. (This interval is from 3.5 milliseconds to 6.0 milliseconds if the read attempt immediately follows a backward slew). Within the timing restriction of 7.0 milliseconds (3.5 milliseconds following a backward slew), the programmer can use the start time for some minor computation or housekeeping. If the Start Data Transfer command does not reach the tape transport in time, nothing will be read, but the tape will proceed to the normal end of read position at the interrecord gap.

When the data transmission is over and has been terminated by an END code, 7.9 milliseconds elapse before the tape has stopped and is ready for further use. When the tape has stopped and is again ready, an interrupt request will be sent to the controlling unit. During the read operation, when the specified number of words has been read but the end of the block has not been reached, the Central Processor (controlling device) will end the transmission by sending an END code to the tape transport. In all other cases, the tape transport sends the END code to the Central Processor when the read head reaches the "cue."

If at any time during the read operation a parity fault is detected, the erroneous character is replaced in the transmission by an ERR code. The ERR code is sent to the controlling device instead of the character; reading ceases and the tape slews to the normal end of block position. The error indicator is also set.

It should be noted that when the tape transport terminates the read operation with an END or with an ERR code sent to the Central Processor, the next command in sequence is executed in the Central Processor program. If an END or ERR code is received in a Control Buffer, the program branches to the error branch locations.

When a tape read operation is performed with a block length less than the block written on the tape, a word of zeros will be stored in the location which immediately follows the location of the last word read.

6. MAGNETIC TAPE DELETE

The Delete Block Mark DBM command, provided for use in tape preparation, enables a defective section of tape to be ignored. The "write enable ring" must be present to permit deletion of a block. If it is absent, the unit will respond with a RED (not ready to proceed) code to the delete command. Delete causes the tape to move backwards and erases all block marks, the preamble, the postscript and data for one block in the backward direction. While the tape is being deleted, the computer is free to compute. The time of deletion of one block of tape (256 machine-word length) is about 50 milliseconds. When the deletion is complete, an interrupt request will be transmitted from the tape transport to the Central Processor.

7. UNRECORDED TAPE HALT

To prevent tape runaway, the tape will be stopped from slewing or moving for any cause after it has traveled for one second without finding a block mark, except in rewind. When the tape has been stopped this way, the tape transport should be manually reset by depressing the zero machine switch. Failure to do this will result in an incorrect interpretation of line signals by the tape when it is called again. Zero reset is not necessary after an unrecorded halt during a slew operation.

If the tape is stopped while at high speed (220 inches per second) (other than by end-of-tape sensors), no further tape motion may be initiated for one second.

8. MANUAL CONTROL

Depressing the STOP button on the tape transport unconditionally stops any motion, puts the unit off-line and lights the STOP button. Provided that there is no interlock condition, pressing the ON-LINE button places the unit on-line for computer control and lights the ON-LINE button.

9. INTERRUPTS

Interrupts from the tape to the controlling device are sent as follows:

When the tape control again becomes available after a read, write or delete operation.

When an individual tape transport again becomes ready after a slew operation. If the tape control unit on which the slewing tape is attached is busy, the interrupt is sent when the control unit becomes non-busy.

When a tape transport, which has been commanded to stop by the STOP instruction, has done so.

An interrupt will never be sent when the tape is not READY.

10. OPERATING STATES

A tape unit may occupy one of the following states:

OUT of Service
STANDBY
CALLED
BUSY
MESSAGE

When OUT of Service, a tape unit is effectively disconnected from the communication line, and can only be put on-line by manual operation of the ON-LINE button on the unit. While OUT of Service, a magnetic tape unit will not answer if a call for the unit is transmitted on the communication line. The tape unit will enter the STANDBY state when the ON-LINE button is depressed. A tape unit is OUT of Service if interlocked and is interlocked if the front door is open, the load handle is in the load position, or a tension arm is fully extended.

When in the STANDBY state, a tape unit examines, and rejects, all call signals until it receives its own unique call (with correct parity). When the unit hears this call, it answers GRN and enters the CALLED state. The unit answers GRN and enters the BUSY-ALERT state if the called unit is slewing or rewinding.

If an IWR, TRA, RCV, or DBM operation is in progress on one of the tape units of the control unit group being called, the answer to the call is RED.

When in the CALLED state, a tape unit can answer all queries and execute all instructions meaningful to it. A call to another unit in the system will return the first unit to STANDBY.

When in the BUSY state, a tape unit is slewing or rewinding or is interlocked. The BUSY state has two sub-states – BUSY-QUIET and BUSY-ALERT. In the BUSY-QUIET sub-state, a tape unit can hear only its own call, which the unit will answer and then enter the BUSY-ALERT state. In the BUSY-ALERT state, a tape unit will answer all queries. After being alerted by a call, a tape unit in the BUSY state replies RED to a query ready, QRD. When another unit is called, the first unit returns to BUSY-QUIET.

While engaged in a block data transmission, a tape unit is said to be in the MESSAGE state. Communication with any other unit of that tape group is forbidden until transmission of the block is completed.

11. LINE INSTRUCTIONS

The line instructions which are used with the MT-10 are given below. The octal code, the alpha code and the name of each are given.

Octal	Alpha	Name
nnn	CAL	Call

The tape transport with the call number nnn is called with the CAL instruction.

The answer to CAL is GRN if the control unit to which the transport is attached is free. The reply is RED if the control unit is BUSY. If the tape transport is out of service or interlocked, there is no answer.

If one tape unit is in the CALLED state and another unit is called, the first unit goes to the STANDBY state without reply. If one tape unit is in the BUSY-ALERT state and another unit is called, the first unit goes to the BUSY-QUIET state without reply.

If the magnetic tape does not answer the Call, the program can be recovered by calling another unit on the Communication Line and then calling the tape unit again. The unit will answer the second time.

002	GRN	Green

A GRN character is the response from a terminal unit the Central Processor (or Control Buffer) receives to instruct it to continue in the expected sequence.

003	RED	Red

A RED character is the response from a terminal unit the Central Processor (or Control Buffer) receives to instruct it to branch to an alternate action.

010 SDT Start Data Transfer

After a unit is fully instructed to participate in a data transfer, the Start Data Transfer instruction, SDT, is transmitted to it. The unit enters the MESSAGE state. Then, the REQ signal is immediately transmitted from the receiving unit to the transmitting unit.

004 END End of Block

The END character signifies the completion of a data transmission. The END character requires no reply. The tape transport sends the END code if the tape comes to the end of the block during a read operation. The controlling device sends the END code at the end of a write or read operation where the termination is by command rather than by the end of block.

The tape unit is not available after the END is transmitted until motion of the tape is stopped and an interrupt request pulse has been sent by the tape unit.

005 ERR Error

An ERR character instead of the expected response is sent from a magnetic tape unit to the Central Processor (or Control Buffer), if a parity error has been detected during the read operation. ERR requires no reply. The tape unit is not available after the ERR character is sent until tape motion has stopped. The tape will halt at the normal interrecord position. The reading of the block is terminated by the sending of ERR.

006 SW1 Switch to Line 1 (Primary)

The controlling device (Central Processor or Control Buffer) transmits the SW1 instruction character to a magnetic tape unit to instruct the tape unit to switch to Communication Line 1, the Primary Line. If the unit is on Line 2, the unit answers GRN and will not acknowledge any transmission for 5 milliseconds during switching. This instruction switches all of the tape units on that control unit to Line Number 1. If the tape unit is already on Line 1, no switching occurs, the unit goes to the STANDBY state and no response is transmitted to the controlling unit.

007 SW2 Switch to Line 2 (Secondary)

The controlling unit transmits the SW2 instruction character to a magnetic tape unit to instruct the tape unit to switch to Communication Line 2, the Secondary Line. If the unit is on Line 1, it answers GRN and will not acknowledge any transmission for 5 milliseconds during switching. This instruction switches all of the tape units on that control unit to Line Number 2. If the tape unit is already on Line 2, no action occurs, the unit goes to the STANDBY state and no response is transmitted to the controlling unit.

011 OUT Out of Service

The controlling unit transmits the OUT instruction character to a magnetic tape unit to instruct the tape unit to go out of service (off-line). The unit answers GRN and disconnects itself from the communication line. The operator must depress the ON-LINE button on the tape unit to return the unit to service. OUT unconditionally terminates any tape motion.

014 RCV Receive

The controlling unit transmits the RCV instruction character to a magnetic tape unit to instruct the tape unit to receive data characters. The tape begins to move. The reply to RCV is normally GRN. If the "Write Enable" ring is not in place, the reply is RED. After a RED reply, the tape unit must be called again to return it to the CALLED state.

016 TRA Transmit

The controlling unit transmits the TRA instruction character to a magnetic tape unit to instruct the tape unit to transmit data characters. The tape begins to move. The reply to TRA is GRN. TRA must be preceded by FWD.

044 FWD Forward

The FWD instruction signifies that the forthcoming TRA or slew operation is to be in the forward direction. The unit answers GRN. (FWD preceding IWR, RCV or DBM is ignored).

045 BAC Backward

The BAC instruction signifies that the forthcoming slew operation is to be in the backward direction. The unit answers on GRN. (BAC preceding IWR, RCV, DBM is ignored; BAC preceding TRA is not allowed).

046 IWR Initial Write

The controlling unit transmits the IWR instruction character to a magnetic tape unit to instruct the tape unit to receive data characters, write them on unrecorded tape and supply block marks. The reply to IWR is normally GRN. If the "Write Enable" ring is not in place, the reply is RED and no action occurs. After a RED reply, the tape unit must be called again to return it to the CALLED state.

047 REW Rewind

The controlling unit transmits the REW instruction character to a magnetic tape unit to cause the tape to be rewound to the beginning of tape sensor. The reply is GRN.

050 STP Stop

The controlling unit transmits the STP instruction character to a magnetic tape unit to cause the tape motion to halt at once. The next tape action following the STP instruction must be a slew or rewind instruction. On unrecorded tape, the next action can be initial write. The first end-of-block mark which is encountered will be considered as the first block. The reply is GRN. STP will not stop a rewind operation.

056 DBM Delete Block Mark

The controlling unit transmits the DBM instruction character to a magnetic tape unit, on which the "Write Enable" ring is present, to cause the block marks and data to be erased from one block in the backward direction. The normal reply is GRN. If the "Write Enable" ring is not in place, the reply is RED and no action occurs.

060 QRD Query Ready

The controlling unit transmits the QRD instruction character to a magnetic tape unit to determine if the tape unit is available for use. If the unit is stopped, it answers GRN. If stopped at the beginning or end of the tape, or otherwise not ready, the reply is RED.

061 QER Query Error

The controlling unit transmits the QER instruction character to a magnetic tape unit to determine if there has been an error on that unit since the last QER was sent. If so, the unit answers RED; if not, the unit answers GRN. The error indicator is reset.

063 QUT Query Unrecorded Tape

The controlling unit transmits the QUT instruction character to a magnetic tape unit to determine if the tape has been stopped by the unrecorded tape detector. If so, the unit answers RED; if not, the unit answers GRN. The unrecorded tape detector is reset.

064 QBT Query Beginning of Tape

The controlling unit transmits the QBT instruction character to a magnetic tape unit to determine if the unit has stopped at the beginning of the tape. If so, the unit answers RED; if not, the unit answers GRN.

If the magnetic tape does not answer the QBT, the program can be recovered by calling another unit on the Communication Line, calling the tape unit again, and then sending the QBT. The tape unit will transmit an answer the second time.

065 QET Query End of Tape

The controlling unit transmits the QET instruction character to a magnetic tape unit to determine if the unit has stopped at the end of the tape. If so, the unit answers RED; if not, the unit answers GRN.

If the magnetic tape does not answer the QET, the program can be recovered by calling another unit on the Communication Line, calling the tape unit again, and then sending the QET. The tape unit will transmit an answer the second time.

lnn nn Command Numeric

When preceded by an FWD or BAC instruction character, two lnn numeric characters specify the number of blocks to be slewed. The first numeric specifies the two most significant octal digits and the second numeric specifies the two least significant octal digits of the four-digit number specifying the number of blocks to be slewed. The maximum number is 4095_{10}. The unit answers GRN.

12. A SLEW MAGNETIC TAPE SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program sample considered will slew a magnetic tape $1571_0 = 2358$ blocks in the forward direction.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The call designation for the magnetic tape unit is to be 304.

The starting address for the program is 200.

The starting address for the storage of the instruction characters is 1100.

An error routine is assumed to start in location 2100.

Another program in the Central Processor is assumed always to start in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN characters and interrupt signals. None of these characters or signals must be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading "MT-10".

200 | 0 | 033 | 00 | 01100 |

This command is the first word of the block command.

201 | 0 | 14 | (CAL) 304 | 00001 |

This command is the second word of the block command.

1100 | (QRD) (FWD) (02) (35) |
| 0 6 0 1 1 0 4 1 1 3 5 |

Central Processor	MT-10	
CAL	GRN	The magnetic tape unit is called and answers GRN.
QRD	GRN	The tape unit is queried to see if it is ready to accept more instructions. The reply is GRN.
FWD	GRN	The magnetic tape unit is instructed to move the tape in the forward direction when the tape is instructed to move. The reply is GRN.
102	GRN	The command numeric 102 gives the most significant two octal integers of the slewing destination (i.e. 02). The tape begins to move. The reply is GRN.
135	GRN	The command numeric 135 gives the least significant two octal integers of the slewing destination (i.e. 35). The reply is GRN.

202 | 0 | (TRA) 017 | 00 | 02100 |

This command is the error exit for a non-GRN response.

203 | 0 | (TRA) 017 | 00 | 02000 |

The Central Processor returns to another program. When the magnetic tape unit completes the slewing operation, the unit transmits an interrupt to the Central Processor.

If the tape control which controls transport 304 is not busy, an interrupt will be transmitted when the slewing operation is completed. If the tape control is busy, the interrupt is sent when the tape control becomes available.

13. SLEW MAGNETIC TAPE EXAMPLE PROGRAM

The following example program will slew the tape $235_8 = 157_{10}$ blocks. The program is assumed to start in location 200. A minimum ISR is assumed to be in memory for use with this example.

Locations 200 and 201 contain a block transmit command which is used to transmit the necessary line instructions to the tape unit. These instructions are located in location 204.

Location 202 contains an error exit to a standard error routine (see explanation below). If a non-GRN answer is received from any of the instructions, the error exit is taken by the program.

Location 203 contains an exit to another part of the program.

When the slew is completed, an interrupt request will be sent to the Central Processor from the tape unit.

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		033	00	01100		Block Transmit
201	0		143	04	00001		Instruction characters
202	0		017	00	02100		Error exit
203	0		017	00	02000		Out to another program
204	0		601	10	41135		QRD, FWD, 02, 35

14. READ MAGNETIC TAPE SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program considered will read one 256_{10} (i.e. 400_8) word block of tape (1024_{10} 8-bit characters).

The following conditions are assumed to be true for this sample.

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The call designation for the magnetic tape unit is 304.

The starting address for the program is 200.

The starting address for the storage of the instruction characters is 1100.

The starting address for the storage of the data to be received is 2400.

An error routine is assumed to start in location 2100.

The error routine will handle all non-GRN (or non-RED when applicable) responses which occur during the transmission of line instructions to peripheral equipment. The routine will service a RED response from any line instruction. Any instruction numeric or data character which is received into the Line Response register in place of GRN will also be serviced by the error routine. During a data transmission (block or single character) the error routine will service all non-data or non-REQ responses.

Another program in the Central Processor is always assumed to start in location 2000.

The sequence of Communication Line signals is given below in conjunction with the program commands. The communications system automatically transmits each of the GRN and END characters, and each of the REQ and interrupt signals. None of these characters or signals need be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading "MT-10."

200	0 033 00 01100	This command word is the first word of the block input/output command.
201	0 14 (CAL) 304 00001	This command word, the second word of the block input/output command, transmits instruction characters to the tape unit.

1100 (CAL) (QRD) (FED) (TRA)
3 0 4 4 2 0 2 2 0 1 6

Central Processor	MT-10	
CAL	GRN	The magnetic tape unit is called and answers GRN.
CAL	GRN	The magnetic tape unit is called and answers GRN.
QRD	GRN	The magnetic tape unit is queried to see if it is ready to accept more instructions. The reply is GRN.
FWD	GRN	The magnetic tape is instructed to go forward when motion begins.
TRA	GRN	The magnetic tape unit is instructed to be the transmitter in the data transmission to follow. The reply is GRN. The tape begins to move when this instruction is received.

202	0 (TRA) 017 00 02100	This command is an error exit for a non-GRN reply.

Location 203 can contain a command which transfers the Central Processor away from the read program while the magnetic tape is coming into the read position. This has to be timed so the program will return to process the data by 7.0 milliseconds.

203	0 033 00 02400	This command is the first word of the block input/output command.
204	0 12 (SDT) 010 00400	This command word, the second word of the block input/output command, initiates the transmission of 1024 characters from the Magnetic Tape unit to the Central Processor.

Note: When the block length indicated in the block input/output command is equal to the block length on the tape to be read, the transmission is terminated by the Central Processor.

The line signals during the transmission are:

SDT		The Central Processor informs the tape unit that the data transmission has started. There is no reply expected.
REQ	ddd	A request signal is sent on the REQ signal line to request the first data character. The tape unit answers with a data character.
REQ	ddd	A request for the next data character is sent. The tape unit answers with the next data character.

The data transmission continues through the next to last data character of that block. Then:

REQ	ddd	A request for the last data character is sent. The tape unit answers with the last data character of the block.
END		The Central Processor sends END to indicate the end of the block.

Note: The computer may send END earlier if it wishes to read a block of information which is less than the block size on the tape but the motion of the tape unit will continue until the end of block mark is encountered. The communication line and the computer are free during this latter time. The tape unit will send the END if the end of block mark is reached before the block length of the command is reached.

		(TRA)			
205	0	017	00	02100	This command will be the exit if due to some error the tape unit transmits an END or ERR to terminate the transmission.

As shown above, the block command in locations 203 and 204 causes the Central Processor to automatically receive 1024 8-bit characters, construct from them 256_{10} machine words and place the 256 (400_8) words in locations 2400_8 through 3077_8.

2400	First Character	Second Character	Third Character	Fourth Character
2401	Fifth Character	Sixth Character	Seventh Character	Eighth Character
3077	1021st Character	1022nd Character	1023rd Character	1024th Character

Then the Central Processor (under program control) checks the terminating response and returns to another program.

15. READ MAGNETIC TAPE EXAMPLE PROGRAM

The following example program will read a 256_{10} word block into locations 2400 through 3077. The program is assumed to start in location 200. A minimum ISR is assumed to be in memory for use with this example.

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		005	00	00000		CL A 0
201	0		113	00	04000		0 → (4000)
202	0		012	01	00003		3 → (01)
203	0		157	00	00304		Send CAL
204	0		017	00	02100		Error exit
205	0		157	00	00060		Send QRD
206	0		017	00	00217		Exit to Not Ready Routine
207	0		157	00	00044		Send FWD
210	0		017	00	02100		Error exit
211	0		157	00	00016		Send TRA
212	0		017	00	02100		Error exit
213	0		033	00	02400		Block receive data
214	0		120	10	00400		256 word block
215	0		017	00	00226		Exit to END – ERR check
216	0		017	00	03100		Out to another program
217	0		157	00	00064		Begin Not Ready routine
220	0		017	00	03200		To a slew routine
221	0		157	00	00065		Send QET
222	0		017	00	03300		To a slew back routine
223	0		026	01	00001		(01) -1 = 0?
224	0		017	00	00205		Transfer to send QRD
225	0		017	00	02100		Error exit
226	0		065	00	00001		Begin END – ERR routine
227	0		113	00	04000		Location of exit into (4000)
230	0		072	02	00377		(Line Response) Λ 377 into (ACC)
231	0		131	00	00004		(ACC) = 4?
232	0		017	00	03500		Transfer
233	0		131	00	00005		(ACC) = 5?
234	0		017	00	03400		To a tape error detection routine
235	0		017	00	02100		Error exit

Explanation of the Read Routine

200	0		005	00	00000	CL A 0
201	0		113	00	04000	0 → (4000)
202	0		012	01	00003	3 → (01)

Locations 200 through 202 are used to do some preliminary bookkeeping.

203	0		157	00	00304	Send CAL
204	0		017	00	02100	Error exit
205	0		157	00	00060	Send QRD
206	0		017	00	00217	Exit to Not Ready Routine
207	0		157	00	00044	Send FWD
210	0		017	00	02100	Error exit
211	0		157	00	00016	Send TRA
212	0		017	00	02100	Error exit

The commands in locations 203 through 212 are used to set the MT-10 to read when the data transfer begins. The command in location 203 sends the Call to the MT-10. The command in location 205 sends a Query Ready instruction to the MT-10. The command in location 207 sets the MT-10 to move in the forward direction. The command in location 211 sends the line instruction which sets the MT-10 to transmit to the Central Processor when the transmission begins. Locations 204, 210 and 212 contain unusual error exits. Location 206 contains an exit to a Not Ready routine which begins in location 217.

213	0		033	00	02400	Block receive data
214	0		120	10	00400	256 word block
215	0		017	00	00226	Exit to END – ERR check
216	0		017	00	03100	Out to another program

The block command in locations 213 and 214 causes the Central Processor to send the Start Data Transfer instruction to the MT-10. Then the Central Processor receives the data as it is read from the tape. If there is no error, the transmission in this example will be terminated by the sending of an END from the Central Processor to the MT-10. In this case, the transfer in location 216 is executed. If the MT-10 were to terminate the transmission, the command in location 215 would transfer control to an END-ERR routine beginning in location 226 to determine the cause of such a termination.

217	0		157	00	00064		Begin Not Ready routine
220	0		017	00	03200		To a slew routine
221	0		157	00	00065		Send QET
222	0		017	00	03300		To a slew back routine
223	0		026	01	00001		(01) -1 = 0?
224	0		017	00	00205		Transfer to send QRD
225	0		017	00	02100		Error exit

On a transfer from location 206, these commands will determine why the MT-10 is not ready. The command in location 217 sends the Query Beginning of Tape line instruction to the MT-10. If the tape is at the beginning of tape sensor, the command in location 220 transfers to a slew forward routine (this routine is not given in this example). The command in location 221 sends the query end of tape line instruction to the MT-10. If the tape is at the end of tape sensor, the command in location 222 transfers to a slew backward routine (this routine is not given in this example). The command in location 223 decrements a counter which determines the number of times this test is performed. The command in location 224 transfers to the send QRD command in location 205. An error exit is found in location 225.

226	0		065	00	00001		Begin END – ERR routine
227	0		113	00	04000		Location of exit into (4000)
230	0		072	02	00377		(Line Response) Λ 377 into (ACC)
231	0		131	00	00004		(ACC) = 4?
232	0		017	00	03500		Transfer
233	0		131	00	00005		(ACC) = 5?
234	0		017	00	03400		To a tape error detection routine
235	0		017	00	02100		Error exit

On a transfer from location 215, these commands will determine what the response was that terminated the data transmission and then branch to the appropriate error checking routines. Locations 226 and 227 place the address of the last data character to be received in location 4000. The response which terminated the transmission is brought from the Line Response register to the accumulator with the command in location 230. The command in location 231 tests whether the character is END. If so, the command in location 232 transfers control to some error finding routine (this routine is not given in this example). If not, the command in location 233 tests whether the character is ERR. If so, the command in location 234 transfers control to some error finding routine (this routine is not given in this example). If not, the unusual error exit is taken from location 235.

When the motion of the magnetic tape has stopped after the read operation, an interrupt request will be sent to the Central Processor. The ISR will be entered and will take the exit previously prepared by the programmer.

16. REWRITE MAGNETIC TAPE SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program considered will rewrite one 256 (i.e. 400_8) word block of tape (1024 8-bit characters).

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The call designation for the Magnetic Tape unit is 304.

The starting address for the program is 200.

The starting address for the storage of instruction characters is 1100.

The starting address for the storage of the data to be transmitted is 1400.

An error routine is assumed to start in location 2100.

Another program in the Central Processor is always assumed to start in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and END characters, and each of the REQ and interrupt signals. None of these characters or signals need be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading MT-10.

200	0	033	00	01100	This command word is the first word of the block input/output command.
201	0	14	(CAL) 304	00001	This command word, the second word of the block input/output command, transmits instruction characters to the tape unit.

1100 (CAL) (CAL) (QRD) (RCV)
3 0 4 6 1 0 3 0 0 1 4

Central Processor	MT-10	
CAL	GRN	The magnetic tape unit is called and answers GRN.
CAL	GRN	
CAL	GRN	
QRD	GRN	The magnetic tape unit is queried to see if it is ready to accept more instructions. The reply is GRN.
RCV	GRN	The magnetic tape unit is instructed to be the receiver in the data transmission to follow. The reply is GRN. The tape begins to move when this instruction is received.

202	0	(TRA) 017	00	02100	This command is an error exit for a non-GRN reply.

The last command in location 203 can transfer the Central Processor away from the read program while the magnetic tape is coming into the write position. This has to be timed to return the program within 3.5 milliseconds.

203	0	033	00	01400
204	0	10	(SDT) 010	00400

This command is the first word of the block input/output command.

This command word, the second word of the block input/output command, initiates the transmission of 1024 characters from the Central Processor to the Magnetic Tape unit.

Note: If the block length specified exceeds the block length which exists on the magnetic tape, the information sent after the block length is lost.

The line signals during the transmission are:

SDT	REQ	The Central Processor informs the tape unit that the data transmission has started. A request signal is sent by the tape unit on the REQ signal line to request the first data character.
ddd	REQ	The Central Processor answers with the first data character. A request for the second data character is sent.
ddd	REQ	The Central Processor answers with a data character. A request for the next data character is sent.

The data transmission continues through the next to last data character of the block. Then:

ddd	REQ	The Central Processor answers with the last data character to be sent. A request for another character is sent from the tape unit.
END		An END instruction character is sent from the Central Processor to the Magnetic Tape unit to denote the end of the data transmission. No reply is sent from the tape unit.

Then the Central Processor (under program control) returns to another program. When the motion of the tape has stopped, an interrupt is transmitted from the tape transport.

The Initial Write of magnetic tape is identical with the rewrite except RCV is replaced by IWR in location 1100, character 4.

17. REWRITE MAGNETIC TAPE EXAMPLE PROGRAM

The following example program will rewrite a four-word block in a 256_{10} word envelope. Each character written on the tape will have a one-bit in its most significant bit position and will have zeros in all other positions (i.e. octal 200). The characters to be written on tape are stored in locations 1400 through 1403. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

LOCATION	F (31)	M (30-29)	OP (28-27 to 21)	I (20 to 15)	A Field (14 to 0)	u	Comments
200	0		012	01	00003		3 → (01)
201	0		012	02	00004		4 → (02)
202	0		073	02	04000		0 → (4000 + (02)
203	0		026	02	00001		(02) − 1 = 0
204	0		017	00	00202		Transfer
205	0		157	00	00304		Send CAL
206	0		017	00	02100		Error exit
207	0		157	00	00060		Send QRD
210	0		017	00	00217		Exit to Not Ready
211	0		157	00	00014		Send RCV
212	0		017	00	02100		Error exit
213	0		033	00	01400		Block transmit data
214	0		100	10	00004		4 word block
215	0		017	00	00226		Exit to End-ERR routine
216	0		017	00	03100		Out to another program
217	0		157	00	00064		Send QBT
220	0		017	00	03200		Slew forward routine
221	0		157	00	00065		Send QET
222	0		017	00	03300		Slew back routine
223	0		026	01	00001		(01) − 1 = 0?
224	0		017	00	00207		Transfer
225	0		017	00	02100		Error exit
226	0		065	00	00002		Begin END-ERR
227	0		113	00	04000		Exit location into (4000)
230	0		072	02	00377		(LRE) Λ 377 into (ACC)

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
231	0		131	00	00004		(ACC) = 4?
232	0		017	00	00236		Transfer
233	0		131	00	00005		(ACC) = 5?
234	0		017	00	00240		Transfer
235	0		017	00	02100		Error exit
236	0		113	00	04001		(ACC) → 4001
237	0		017	00	03100		To error routine
240	0		072	02	70000		(LRE) Λ 70000 into (ACC)
241	0		131	00	30000		Three characters left?
242	0		017	00	00251		Yes, transfer
243	0		131	00	20000		Two left?
244	0		017	00	00253		Yes, transfer
245	0		131	00	60000		One left?
246	0		017	00	00255		Yes, transfer
247	0		113	00	04005		(ACC) → (4005)
250	0		017	00	03400		To tape error exit
251	0		113	00	04002		(ACC) → (4002)
252	0		017	00	03400		Tape error exit
253	0		113	00	04003		(ACC) → (4003)
254	0		017	00	03400		Tape error exit
255	0		113	00	04004		(ACC) → (4004)
256	0		017	00	03400		Tape error exit
1400	2		004	01	00200		Information to be
1401	2		004	01	00200		rewritten
1402	2		004	01	00200		
1403	2		004	01	00200		

Explanation of the Rewrite Example Program

LOCATION	F (31–30)	M (29–28)	OP (27–21)	I (20–15)	A Field (14–0)	u	Comments
200	0		012	01	00003		3 → (01)
201	0		012	02	00004		4 → (02)
202	0		073	02	04000		0 → (4000 + (02)
203	0		026	02	00001		(02) − 1 = 0
204	0		017	00	00202		Transfer

Locations 200 through 204 are used to do some preliminary bookkeeping.

205	0		157	00	00304		Send CAL
206	0		017	00	02100		Error exit
207	0		157	00	00060		Send QRD
210	0		017	00	00217		Exit to Not Ready
211	0		157	00	00014		Send RCV
212	0		017	00	02100		Error exit

The commands in locations 205 through 212 are used to set the MT-10 to rewrite when the data transfer begins. The command in location 205 sends the Call to the MT-10. The command in location 207 sends a Query Ready instruction to the MT-10. The command in location 211 sends the line instruction which sets the MT-10 to receive data information from the Central Processor when the transmission begins. Locations 206 and 212 are unusual error exits. Location 210 contains an exit to a Not Ready routine which begins in location 217.

213	0		033	00	01400		Block transmit data
214	0		100	10	00004		4 word block
215	0		017	00	00226		Exit to End-ERR routine
216	0		017	00	03100		Out to another program

The block command in locations 213 and 214 causes the Central Processor to send the Start Data Transfer instruction to the MT-10. Then the Central Processor transmits the data from location 1400 through 1403 to the MT-10. If there is no error in the transmission, this example will be terminated by the sending of an END from the Central Processor to the MT-10. In this case, the transfer in location 216 is executed. If the MT-10 were to terminate the transmission, the command in location 215 would transfer control to an END-ERR routine beginning in location 226 to determine the cause of such a termination.

217	0		157	00	00064		Send QBT
220	0		017	00	03200		Slew forward routine
221	0		157	00	00065		Send QET
222	0		017	00	03300		Slew back routine
223	0		026	01	00001		(01) − 1 = 0?
224	0		017	00	00207		Transfer
225	0		017	00	02100		Error exit

On a transfer from location 210, these commands will determine why the MT-10 is not ready. This series of commands is identical to the Not Ready routine from the MT-10 Read example.

226	0		065	00	00002		Begin END-ERR
227	0		113	00	04000		Exit location into (4000)
230	0		072	02	00377		(LRE) Λ 377 into (ACC)
231	0		131	00	00004		(ACC) = 4?
232	0		017	00	00236		Transfer
233	0		131	00	00005		(ACC) = 5?
234	0		017	00	00240		Transfer
235	0		017	00	02100		Error exit
236	0		113	00	04001		(ACC) → 4001
237	0		017	00	03100		To error routine

On a transfer from location 215, these commands will begin the determination of what the response was that terminated the data transmission. The various exits from these commands will lead to micro-routines which will also determine at which character the transmission terminated. Locations 226 and 227 place the address of the last data character sent in location 4000. The response which terminated the transmission is brought from the Line Response register to the accumulator with the command in location 230. The command in location 231 tests whether the character is END. If so, the command in location 232 transfers control to location 236. If not, the command in location 233 tests whether the character is ERR. If so, the command in location 234 transfers control to location 240. If not, the unusual error exit is taken from location 235.

On transfer from location 232, the command in location 236 places the contents of the accumulator into location 4001 and then the command in location 237 transfers control to some error routine (this error routine is not given in the example).

240	0		072	02	70000		(LRE) Λ 70000 into (ACC)
241	0		131	00	30000		Three characters left?
242	0		017	00	00251		Yes, transfer
243	0		131	00	20000		Two left?
244	0		017	00	00253		Yes, transfer
245	0		131	00	60000		One left?
246	0		017	00	00255		Yes, transfer
247	0		113	00	04005		(ACC) → (4005)
250	0		017	00	03400		To tape error exit

The command in location 240 brings the character count from the Line Response register into the accumulator. The command in location 241 tests whether the character count is three left in the word that was being processed. If so, the command in location 242 transfers control to location 251. If not, the command in location 243 tests whether the character count is two left in the word that was being processed. If so, the command in location 244 transfers control to location 253. If not, the command in location 245 tests whether the character count is one left in the word that was being processed. If so, the command in location 246 transfers control to location 255. If not, the command in location 247 stores the contents of the accumulator in location 4005 and then the command in location 250 transfers control to some tape error routine.

251	0		113	00	04002		(ACC) → (4002)
252	0		017	00	03400		Tape error exit
253	0		113	00	04003		(ACC) → (4003)
254	0		017	00	03400		Tape error exit
255	0		113	00	04004		(ACC) → (4004)
256	0		017	00	03400		Tape error exit

On a transfer from the appropriate location, these commands store the contents of the accumulator and exit to some tape error routine as indicated.

When the motion of the magnetic tape has stopped after the rewrite operation, an interrupt request will be sent to the Central Processor. The ISR will be entered and will take the exit previously prepared by the programmer.

CHAPTER THREE
THE PC-10 PRINTER AND CARD COUPLER

1. PRINTER AND CARD COUPLER OPERATIONS

The Printer and Card Coupler, PC-10, is used in the G-20 Computing System to couple the Central Processor to the pieces of line printing, card reading and punching, and tabulating equipment used on-line. The standard equipment configuration used with the PC-10 is a Bendix Line Printer, a card punch and verify station, and a card read and verify station. Standard pieces of card equipment can be coupled to the Central Processor with the PC-10. Line printing, card reading and card punching are performed under program control from the Central Processor. Information is transmitted between the Central Processor and the PC-10 via the communication line connecting them.

More than one card machine can be operated concurrently through a PC-10 on a time-shared basis. For example, a punch with verification and a reader with verification can operate simultaneously if there is sufficient time for the Central Processor to transmit new information to both machines, one at a time, while the cards in the respective machines are being shifted from one row to the next during processing. Data is received by, and transmitted from, the Printer and Card Coupler in rows. For example, to read a twelve-row card, twelve blocks of information, one block for each row on a card, is transmitted from the card reader to the Central Processor.

The PC-10 takes 8-bit characters from the Communication Line during a transmission to a card punch or line printer and accumulates them serially into data rows of appropriate length. If 6-bit data characters are transmitted to the PC-10, they appear as and are processed as 8-bit data characters with the most significant two digits of each 8-bit character being zero.The PC-10 receives row data from a card reader, collects the information in 8-bit data characters and transmits these on the communication line. The PC-10 can sense when a complete row of card information is read but has no other counting facility. If the Central Processor transmits more data characters than the PC-10 needs for one card or print row, the PC-10 does not process any of the additional characters but will reply with a request for each.

The PC-10 operates in the following States:

OUT of Service

STANDBY

CALLED

INSTRUCTED, or

MESSAGE

A special state, INITIAL LOAD, which is a modified MESSAGE state, is also permitted. When a unit is in the INITIAL LOAD state, initial loading of the memory in the Central Processor is permitted via cards. After a PC-10 is turned on, a switch labeled INITIAL LOAD is operated to select the initial load mode. When the PC-10 is turned on, it enters the STANDBY state.

2. CARD READER

Cards are read in row binary via the PC-10. After a card reader has been instructed to read, an interrupt request is sent to the Central Processor each time a card row is in position to be read. The Central Processor processes the interrupt request and determines that the interruption was from a card reader attached to the PC-10. Then the Central Processor instructs the PC-10 to transmit the data from that row. The Central Processor can keep a count of the rows of a card during processing or it can transmit a "Query Last Row", QLR, while each card row is being processed. If the last row is being processed when the Central Processor transmits the QLR, the reader will respond with RED; if not, the response is GRN.

While reading cards with the 088 reader, the cards read may be stacked in one of three stackers. The stacker designation can be selected under program control (See AS1, AS2, page 34).

See page 35 for a complete card read sample.

The card reader attached to the PC-10 is used for initially loading the Central Processor as follows:

After power has been supplied to the system, (1) place the Central Processor in the INITIAL LOAD state, activate the System Interlock Switch, (2) place the initial load cards in the reader and press the reader START button once, and (3) then turn the INITIAL LOAD switch of the PC-10 on. This causes one card to be read into the memory of the Central Processor starting at location 64.

During initial load only, the words are packed as follows:

> The first 32 column positions of row 12 are placed in location 64. Column positions 33 through 64 of row 12 are placed in location 65. Column positions 65 through 80 of row 12 are placed in the most significant sixteen bit positions of location 66; column positions 1 through 32 of row 11 are placed in the least significant sixteen bit positions of location 66. This procedure continues until one card is read into memory into locations 64 through 93.

The Central Processor then executes the command in location 65. Since one card row may not contain more than 2½ machine words, the initial loading program must be constructed to initiate further loading with 30 words or less.

The PC-10 returns to the STANDBY state when the first card is completely entered.

See page 79 for a sample sequence of communication line signals involved in the initial load procedure.

3. CARD PUNCH

Cards are punched in row binary via the PC-10. After the card punch has been instructed to punch, an interrupt signal is sent to the Central Processor each time a card row is in position to be punched. The Central Processor processes the interrupt and determines that the interruption was from a card punch attached to the PC-10. Then the Central Processor block transmits the information for one card row, sends an END character, and returns to its program. The row number is counted as explained above with the card reader. See page 47 for a complete card punch sample.

While punching cards with the 544 punch, cards may be selectively offset as they are stacked in the after-punch stacker. According to which of the two sets of offset hubs on the 544 plug board has been wired, one of the line instructions, 051 or 052, is used to offset the current card. After the last row of the card approximately 20 milliseconds are available to the programmer to call the punch and send the offset instruction to stack the card out of alignment.

4. LINE PRINTER

The Bendix high-speed Line Printers each have a 66-character, fixed position, print wheel containing 63 printable characters. The LP-10 has a 72-column print wheel; the LP-11 has a 120-column print wheel. The print characters are arranged on the print wheel with the 0 position being the first to be printed, and with the alphabetic characters in the odd row positions and the numeric characters in the even row positions (see Table 2). Line printing can be performed in three modes: alphanumeric, numeric and alphabetic. These modes are selected by a line instruction character,

PAN Print Alphanumeric,

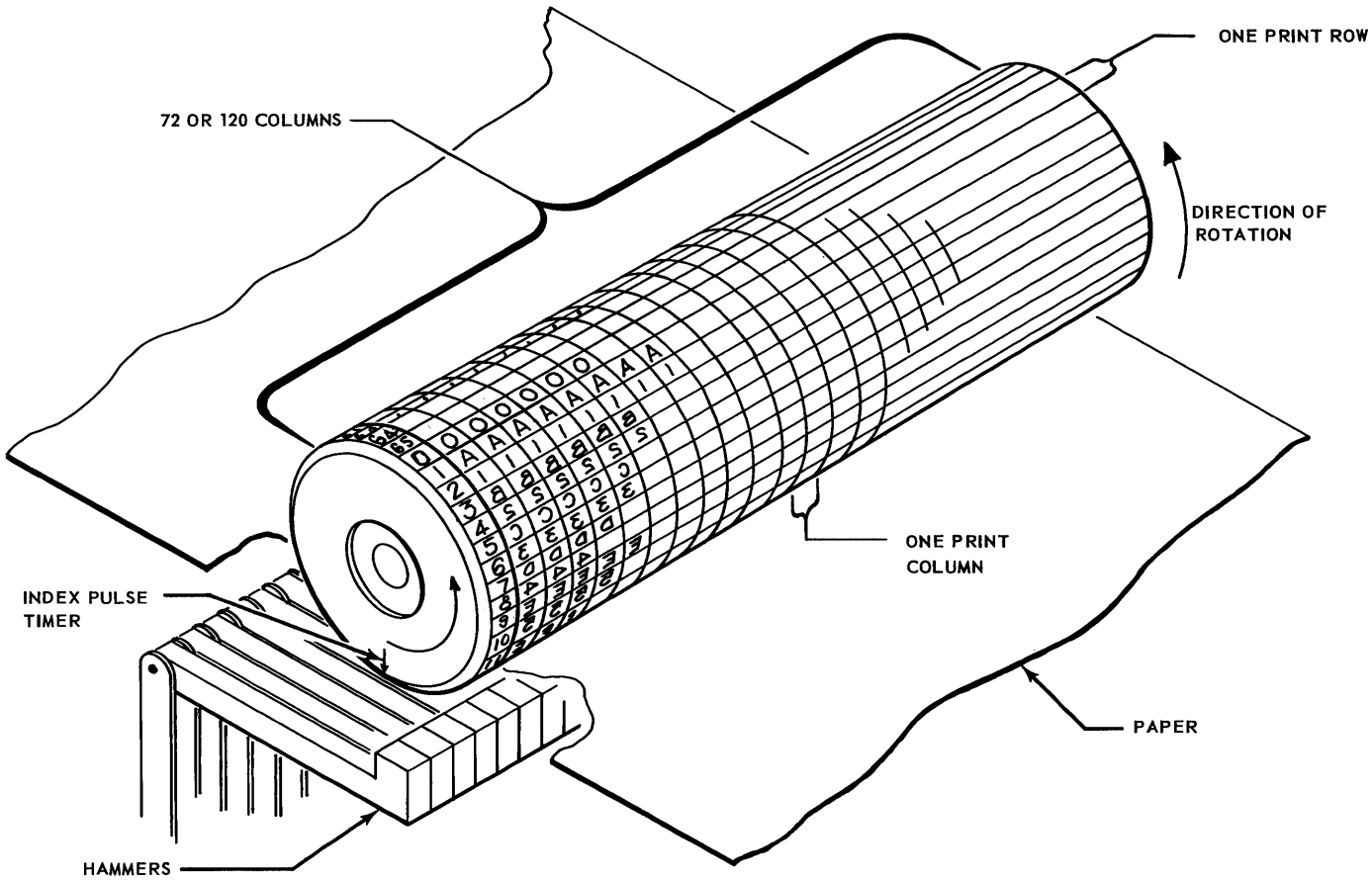
PNU Print Numeric, or

PAB Print Alphabetic,

transmitted from the Central Processor to the PC-10. The printing mode will remain as selected until changed by a subsequent printing mode line instruction. In the alphabetic printing mode, only the characters in the odd row positions can be printed. In the numeric printing mode, only the characters in the even row positions can be printed. All characters can be printed in the alphanumeric printing mode. (Note: Even though the space is a legitimate character in the Bendix code, it is not considered a character to be printed and must not be printed.)

As shown in the figure, the printer consists of a rotating print wheel with fixed print characters, a set of hammers which do the actual printing and the necessary associated hardware. To print a line of print it is necessary to cause the hammers to selectively press the print paper against the rows of characters on the print wheel as the rows pass under the hammers. For example, to print ACB in the first three positions on the paper, the first hammer is activated when A row of the print wheel is in place,

PRINT WHEEL FORMAT



the third hammer is activated when the B row of the print wheel is in place and the second hammer is activated when the C row of the print wheel is in place. Thus the print wheel must make one complete revolution to print one line of print.

To instruct the line printer to activate the desired set of hammers as the different print character rows come into position, a code of zeros and ones is sent to the line printer for each row. This code consists of 72 or 120 bit positions depending on the size of the print wheel being used. To cause characters of the row currently in place to be printed, 1's are placed in the corresponding column positions and zeros are placed in the remaining column positions. Since this code must be transmitted for each character row on the wheel (depending on the mode), the codes to be sent for all of the characters can be thought of as a matrix (of ones and zeros) which is 72 or 120 bits wide and 63 bits long for the alphanumeric mode, 72 or 120 bits wide and 32 bits long for the numeric mode, and 72 or 120 bits wide and 31 bits long for the alphabetic mode. This matrix is called the "print image" matrix for one line of print. Considering the print image matrix as a whole:

> When the mth row of the print image matrix is processed, a 1 in the nth column position of the mth row of the print image matrix will cause the mth character on the print wheel to be printed in the nth column position of the line of type.

A printing cycle is entered by calling the line printer, selecting the printing mode and transmitting to the line printer the SIN (Send Index Pulse) instruction character. Following the SIN character, the PC-10 will transmit an interrupt to the Central Processor when the print wheel is in position to begin printing a line of print and the paper is not being upspaced. (see example, page 62). The Central Processor then queries the line printer about the index pulse (QIP), transmits a Query Ready (QRD), and instructs the line printer to prepare to print. The Central Processor, using a block transmit command then transmits an SDT (Start Data Transmission) instruction character,

followed by one row of the print image, to the line printer via the PC-10. The remaining rows of the print image, each representing a character on the print wheel, are transmitted in the same way (see sample below). The approximate time in milliseconds available between print characters is as follows:

Printing Mode	LP-10	LP-11	
	(900 RPM)	(1000 RPM)	(1500 RPM)
Alphanumeric	1	.92	.615
Alphabetic	2	1.84	1.23
Numeric	2	1.84	1.23

When print images are being prepared between lines of print, the maximum printing speed available is limited by the speed of the Central Processor. In the numeric printing mode only about 50 per cent of the print wheel is normally used; therefore, the remaining wheel time is available for constructing the print image. In the numeric printing mode, a complete print image can ordinarily be constructed for the LP-10 during the time remaining for the print wheel to complete a revolution.

When the last print character to be used in a print cycle has been printed, the print cycle can be terminated with an STT (Start Paper Upspace) or the CLQ (Clear Queries) line instruction character. When the last print character of a line of print has been printed, the program should test for an error with the QER instruction character and then enter a tight Query Ready (QRD) loop. When a reply of GRN is received, the STT instruction character can then be sent.

Upspacing of the paper in the line printer can be controlled by a paper tape format loop; or, paper can be advanced under direct program control without reference to a fixed format. The desired fixed format is selected by the instruction numeric 10n (n is the desired format).

Example:

Use the LP-10 in the alphanumeric printing mode to print " JFG 695.0". Rows 10, 11, 12 and 13 of the print image would be the following:

Column Number	1	2	3	4	5	6	7	8	9	10	11	...	72
Row 10	0	0	0	0	0	0	0	0	0	1	0	...	0
Row 11	0	0	1	0	0	0	0	0	0	0	0	...	0
Row 12	0	0	0	0	0	0	0	1	0	0	0	...	0
Row 13	0	0	0	1	0	0	0	0	0	0	0	...	0

A row, say row 10, of a print image would be placed in memory to be in position to be transmitted to the PC-10 by a block transmission command, as follows:

Location	31			0
3000	00000000	01000000	00000000	00000000
3001	00000000	00000000	00000000	00000000
3002	00000000	00000000	00000000	00000000

If the full 72 column positions in a given row are not used, an END character, transmitted after the transmission of the last, non-zero, 8-bit character, will terminate the transmission (without error). Therefore, in the above example, only the first four 8-bit characters containing information for row 10 need be sent, followed by END. Note: Transmission of the END is automatic but is controlled by the block length specified in the block transmit data command.

When part of the characters on the print wheel will not be printed during a print cycle, it is necessary only to enter and leave the MESSAGE state at each character time. This can be done by transmitting one

word of zeros with a block transmit data command (i.e. by sending SDT and END).

When printing with an IBM 402, 403 or 407 tabulator, the punch adaptor is used with the PC-10 as a coupler. The information to be printed should be formed into a tabulator image in Hollerith code. The tabulator image is read into the tabulator in row binary at the appropriate point in the mechanical cycle of the tabulator. An interrupt request is transmitted from the PC-10 at that point.

5. PC-10 ERROR INDICATORS

Three "unusual condition" indicators – Error, Incomplete Transmission and Interlock – are available to the programmer in the PC-10. Each of these can be queried by instruction and will answer RED if the indicator is turned on and GRN if it is not. A parity check is performed on all characters received by the PC-10 from the Communication Line. An instruction character with wrong parity is ignored. A data character with wrong parity is processed as usual but the Error indicator in the PC-10 is set.

The Error indicator is turned on:

When a data character with wrong parity is processed,

When a row of print image is not completely received in the time available between characters,

When a row of a card is not processed in the time available.

In addition to the general Error indicator, there are more specialized indicators which allow the program to find out exactly which type of error occurred.

The Incomplete Transmission Indicator is turned on when a row of a print image is not completely received in the time available between characters.

The Incomplete Transmission Indicator is turned on when a row of a card is not processed in the time available to the reader and verify reader.

The Incomplete Transmission Indicator is turned on when a row of a card is not processed in the time available to the punch and verify punch.

Two Interlock Indicators are available in the PC-10 and can be queried by a Query Interlock command. One indicator is associated with the reader feed and the other with the punch feed. Each of the indicators reflects the conditions in the assigned feed (i.e., power off, no cards, stacker full, jammed card, and so on). An interlock condition must be manually corrected. When the condition is corrected, the interlock indicator is reset automatically.

6. OPERATING SWITCHES

The three switches associated with control on the PC-10 are the Power Switch, the Communication Line (ON-LINE/OFF-LINE) Switch, and the INITIAL LOAD Switch. The Power Switch has three positions: ON, SLAVE, OFF. In the SLAVE position, the power to the unit is under control of the Master ON-OFF switch of the system. The communication line switch has two positions: ON-LINE and OFF-LINE. The INITIAL LOAD switch has two positions: ZERO SET and INITIAL LOAD. When the INITIAL LOAD switch is in the ZERO SET position, the equipment comes on in the STANDBY state. When the ON-LINE switch and then the INITIAL LOAD switch are turned on, the initial load operation begins.

7. PROGRAMMING AND LINE INSTRUCTIONS

Though the terminal end of all line communication is handled by the PC-10, the programmer may consider that transmissions take place directly to, and directly from, the terminal accessory attached to the PC-10.

LINE INSTRUCTIONS

The line instructions which are used with the PC-10 and its terminal equipment are given below. The octal code, the alpha code and the name of each are given. Some of these instructions are not valid for all of the units attached to the PC-10. In these cases, the units which may be addressed by the instruction are listed.

nnn	CAL	Call

The unit with the call number nnn is called with the CAL instruction. If the unit is in the STANDBY, CALLED or INSTRUCTED state, it enters the CALLED state. A reply, the character GRN, is transmitted from the called unit to the caller.

If one unit is in the CALLED state and another unit is called, the first unit goes to the STANDBY state. No reply is transmitted. The second called unit replies as explained in the first paragraph.

002 GRN Green

A GRN character is the response from a terminal unit the Central Processor receives to instruct it to continue in the expected sequence.

003 RED Red

A RED character is the response from a terminal unit the Central Processor receives to instruct it to branch to an alternate action.

010 SDT Start Data Transfer

After a unit is fully instructed to participate in a data transfer, the Start Data Transfer instruction, SDT, is transmitted to it. The unit enters the MESSAGE state. A reply, the REQ signal, is transmitted from the receiving unit to the transmitter.

014 RCV Receive

The unit receiving the RCV enters the INSTRUCTED state and is instructed to receive data characters when the transmission begins. The reply to the RCV instruction is the character GRN. A printer or punch station may be addressed with this instruction.

016 TRA Transmit

The unit receiving the TRA enters the INSTRUCTED state and is instructed to transmit data characters when the transmission begins. The reply to the TRA instruction is the character GRN. A card reader station may be addressed with this instruction.

004 END End of Block

The unit which is the "transmitter" in a communication sends the END character to the "receiver" to signify the completion of a data transmission. When a unit coupled to the PC-10 sends or receives an END, it then returns to the STANDBY state without further reply.

005 ERR Error

A unit in a communication sends the ERR character to signify that the data transmission ended erroneously. When a unit coupled to the PC-10 sends or receives an ERR, the PC-10 then returns to the STANDBY state without further reply. The ERR will be transmitted if any of the following occur:

A parity error

Incomplete transmission error

Card machine interlock

Time delay error (from Central Processor)

Memory overflow (from Central Processor)

060 QRD Query Ready

The Central Processor transmits the QRD character to a terminal unit to determine if the unit is ready to receive data information. If the unit is ready, it answers GRN; if not, it answers RED.

061 QER Query Error

The Central Processor transmit the QER character to a terminal unit to determine if there has been a parity error or an incomplete transmission error in that unit since the last CLQ instruction was transmitted. If so, the unit answers RED; if not, it answers GRN.

062 QIL Query Interlock

The Central Processor transmits the QIL character to a terminal card unit to determine if an interlock condition exists. If so, the unit answers RED; if not, it answers GRN. The Interlock indicator is not reset by QIL.

063 QIT Query Incomplete Transmission

The Central Processor transmits the QIT character to a terminal unit to determine if the previous row of information was processed in the time allowed. If not, the unit answers RED; if so, it answers GRN. The Incomplete Transmission indicator is not reset by QIT.

065 QIP Query Index Pulse

The Central Processor transmits the QIP character to a line printer unit to determine if the interval between the index pulse and the next print character position pulse is present at this time. If so, the unit answers GRN; if not, it answers RED.

065 QLR Query Last Row

The Central Processor transmits the QLR to a card machine to determine if the unit is processing the last row of the card. If so, the unit answers RED; if not, it answers GRN.

067 QIN Query Interrupt

The Central Processor transmits the QIN character to the PC-10 to determine if the PC-10 sent an interrupt since the last CLQ instruction was transmitted. If so, the PC-10 answers RED; if not, it answers GRN. The Interrupt indicator is not reset by QIN.

001 CLQ Clear Indicators

The Central Processor transmitting the CLQ character to the PC-10 clears all of the following:

The Error indicator

The Interrupt indicator

The Incomplete Transmission Indicator
(for a specified unit)

056 STT Start Feed

The Central Processor transmits the STT character to initiate a mechanical feed. If the unit is a card machine, this instruction feeds one card. If the unit is a line printer, the paper is upspaced one line. The specified unit sends a GRN character in reply.

The CFD (Continue Feed) instruction should follow STT within 6.5 milliseconds for the LP-11 or 8.3 milliseconds for the LP-10 to continue the upspacing operation.

If the format instruction, 10n, follows STT within 6.5 milliseconds for the LP-11 or 8.3 milliseconds for the LP-10, the paper will be upspaced according to the format channel n selected.

Example: If format channel 3 has been prepared for an upspacing of 3 lines, the 103 numeric instruction will cause the line printer to upspace 3 lines and stop.

054 CFD Continue Feed

The Central Processor transmits the CFD character to the unit engaged in the communication to continue a card feed or upspace operation initiated by an STT. The unit sends a GRN character in reply.

If the unit is a card machine, CFD should be sent during the processing of the last card row or within approximately 10 milliseconds after the last card row, depending on the particular card equipment.

The CFD will cause an upspace of one line of paper. If the CFD is sent each time before the paper stops from the previous action, continuous upspacing of the paper will occur. An interrupt request is sent to the Central Processor at the completion of each line of paper upspace after the first upspace (which was initiated by the STT).

10n FCS Format Channel Select

The Central Processor transmits the FCS character to the printer unit engaged in the communication. The unit answers GRN. To select one of the eight upspace formats – 0 through 7 – on the LP-10 or LP-11, the FCS instruction must follow the STT instruction or the CFD instruction within 6.5 milliseconds for the LP-11 or 8.3 milliseconds for the LP-10.

055 SIN Send Index Interrupt

The Central Processor transmits the SIN character to the printer unit engaged in the communication. The unit answers GRN. The SIN causes an interrupt to be sent at the next index pulse time after upspacing of the paper stops.

051 AS1 Use Alternate Stacker 1

The Central Processor transmits the AS1 character to a card unit to instruct the card machine to place the card being operated on into card stacker number 1. The unit answers GRN. Alternate Stacker 1 is reset to stacker 0 at the end of the card cycle.

052 AS2 Use Alternate Stacker 2

The Central Processor transmits the AS2 character to a card unit to instruct the card machine to place the card being operated on into card stacker number 2. The unit answers GRN. Alternate Stacker 2 is reset to stacker 0 at the end of the card cycle.

050 PAN Print Alphanumeric

The Central Processor transmits the PAN character to a line printer to set up the alphanumeric printing mode. The unit answers GRN. The print alphanumeric mode is effective until changed by a 051 or 052 command through the PC-10.

051 PNU Print Numeric

The Central Processor transmits the PNU character to a line printer to set up the numeric printing mode. This mode prints the even positions on the print roll only. The unit answers GRN. The Print Numeric Mode is effective until changed by a 050 or 052 command through the PC-10.

052 PAB Print Alphabetic

The Central Processor transmits the PAB character to a line printer to set up the alphabetic printing mode. This mode prints the odd positions on the print roll only. The unit answers GRN. The Print Alphabetic mode is effective until changed by a 050 or 051 through the PC-10.

PRINT ROLL LAYOUT AND PRINTER CODE

Position	Character	Position	Character
65	Space	0	0
1	A	2	1
3	B	4	2
5	C	6	3
7	D	8	4
9	E	10	5
11	F	12	6
13	G	14	7
15	H	16	8
17	I	18	9
19	J	20	'
21	K	22	10
23	L	24	+
25	M	26	-
27	N	28	*
29	O	30	/
31	P	32	=
33	Q	34	∨
35	R	36	≠
37	S	38	∧
39	T	40	<
41	U	42	$
43	V	44	>
45	W	46	;
47	X	48	(
49	Y	50	[
51	Z	52	]
53	|	54	)
55	←	56	↓
57	→	58	↑
59	¬	60	:
61	,	62	!
63		64	

TABLE 2

8. CARD READ SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program example considered will read 3 cards from a card reader attached to a PC-10.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The octal code for the card reader is 300.

The starting address for the program is 200.

The starting address for the storage of the instruction characters is 1100.

The starting address for the storage of the received data is 1400.

An error routine is assumed to start in location 2100.

Another portion of the program in the Central Processor is assumed always to be in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and END characters and each of the REQ and interrupt signals. None of these characters or signals must be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading "PC-10."

In the following sample, decimal addresses for the command list have been used for simplicity.

		Central Processor	PC-10 (Card Reader)	
200	0 (TLC) 157 00 (CAL) 00300	CAL	GRN	The TLC command transmits "call" to the card reader with octal code 300. The card unit answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
202	0 (TLC) 157 00 (QIL) 00062	QIL	GRN	If an interlock condition exists at the initiation of a card read, this QIL will detect it.
203	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
204	0 (TLC) 157 00 (STT) 00056	STT	GRN	The TLC command transmits the Start Card Feed (STT) instruction to the card reader. The card feed starts bringing the first card into position to be read and answers GRN.
205	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
206	0 (TRA) 017 00 02000			

The last TRA command transfers the Central Processor away from the input output program to another portion of the program while the card reader is positioning the first card. When the card is positioned, an interrupt will be sent to the Central Processor from the PC-10. This interrupt will be processed and the Central Processor will return (under program control) to the input/output program.

207	0 (TLC) 157 00 (CAL) 00300	CAL	GRN	The TLC command transmits a call to the card reader. The answer is GRN.
208	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
209	0 (TLC) 157 00 (QIN) 00067	QIN	RED	A QIN is sent to query whether an interrupt has been sent. The RED answer indicates that there has been an interrupt.
210	0 (TRA) 017 00 00212			This command skips one command to resume in sequence.

```
211   0  (TRA) 017   00   02100
212   0   033      00   01100
213   0   14   (CAL) 300  00001
```

This command is the error exit for a non-RED response.

The 033 command is the first word of the block communication command.

This command word, the second word of the block communication command, initiates transmission of the CAL, QRD, QLR and QIL instructions.

```
1100  (CAL) (QRD) (QLR) (QIL)
      3 0 0 1 4 0 3 2 4 6 2
```

The line signal sequence is:

CAL	GRN	The first station on the PC-10 is called. Let the first be the reader station. The PC-10 answers GRN.
CAL	GRN	
QRD	GRN	A QRD is sent to query whether or not the first station called is the ready station (or if any stations are still ready). A GRN response indicates that the first station called is the ready station. A RED response indicates that another station on the PC-10 must be called to determine the ready station.
QLR	GRN	A QLR is sent to query whether the current row is the last row of the card. A GRN answer indicates that this is not the last row. A RED answer indicates that this is the last row. The routine at the error exit (RED exit) must send the continue feed instruction to the PC-10, to send another card through. This routine (two commands) is placed in location 216 for this example and the error routine at the exit of the block command will transfer to a last row routine when a RED response is encountered.
QIL	GRN	A QIL is sent to query whether an interlock condition exists at this time. A GRN response indicates that there is no interlock; a RED response indicates an interlock.

214	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response on any of the above transmissions. If the exit is on QLR, the error routine transfers to location 216.

(Each of the above instructions can be transmitted using a TLC command, which therefore allows each to have a different error exit for easier processing).

215	0 (TRA) 017 00 00218			This command is used to skip the CFD on a non-last-row pass through the routine.
216	0 (TLC) 157 00 (CFD) 00054	CFD	GRN	The CFD instruction is sent at this time to feed the next card to be read.
217	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
218	0 (TLC) 157 00 (RCV) 00014	TRA	GRN	The TRA instructs the card reader to be the transmitter in the data communication to follow.
219	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
220	0 033 00 01400			The 033 command is the first word of the block communication command.
221	0 12 (SDT) 010 00003			This command word, the second word of the block communication command, initiates the transmission of the data row from the card reader.
222	0 (TRA) 017 00 02100			This command is the error exit for a non-data response during the data transmission. This includes END and ERR.
223	0 (TLC) 157 00 (CAL) 00301			This command is the exit if the Central Processor ends the transmission.

The line signals during the transmission are:

SDT		The card unit is instructed to start the data transmission. There is no reply to the SDT instruction character while in the card read mode.
REQ	ddd	A request signal is sent on the REQ signal line to request the first data character. The card unit answers with a data character.

REQ	ddd	A request for the next data character is sent. The card unit answers with the next data character.

The data transmission continues through the next to last data character of that row. Then:

REQ	ddd	A request for the last data character is sent. The card unit answers with the last data character.
REQ	END	The eleventh request of the row transmission is sent by the Central Processor. An END instruction is then sent from the PC-10 to the Central Processor to denote the end of the data transmission. The Central Processor processes the END character (under program control). Since the normal transmission will terminate with only two characters in the third machine word, the Central Processor will automatically shift these two characters into bit positions 31 through 16 and supply zeros in positions 15 through 0 and store the word in memory.

The Central Processor receives the ten 8-bit data characters, constructs the three machine words from them and places them in locations 1400, 1401 and 1402:

1400	First Character	Second Character	Third Character	Fourth Character
1401	Fifth Character	Sixth Character	Seventh Character	Eighth Character
1402	Ninth Character	Tenth Character	0	0

The information in locations 1400, 1401, and 1402 can be processed between row transmissions or an index can be used to increment location 1400 by three on each return.

After checking for the END or ERR response from the error exit (location 222), the error routine transfers to location 224.

224	0	(TLC) 157	00	(CAL) 00300	CAL	GRN The card unit is called, answers GRN and enters the CALLED state.
225	0	(TRA) 017	00	02100		This command is the error exit.
226	0	(TLC) 157	00	(QER) 00061	QER	GRN The QER is sent to query whether any error other than interlock has occurred.
227	0	(TRA) 017	00	02100		This command is the error exit.
228	0	(TLC) 157	00	(CLQ) 00001	CLQ	GRN The CLQ must be sent to reset all error indicators as well as the interrupt indicator. If the interrupt indicator is not reset, no further interrupts can be obtained from the PC-10.
229	0	(TRA) 017	00	02100		This is an error exit.

At this point the programmer can set up the counter to count the number of cards to be processed. In this example the counter will count three cards.

After the necessary bookkeeping, the programmer has some time left before the interrupt occurs for the next row. When the interrupt occurs, the interrupt servicing program should transfer back to location 207 to initiate the reading of the next row.

9. CARD READ EXAMPLE PROGRAM

The following example program will use a card reader attached to the PC-10 to read one card. After the card is read into memory, the contents of what were the last eight columns of row number zero will be tested equal to 003. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		012	02	00000		Set index 02 to zero
201	0		157	00	00300		Send CAL
202	0		017	00	02100		Error exit, unusual errors
203	0		157	00	00062		Send Query Interlock
204	0		017	00	03100		To interlock error routine
205	0		157	00	00056		Send start card feed
206	0		017	00	02100		Error exit
207	0		012	01	00044		Set index 01 to 44
210	0		073	01	01377		Store zeros
211	0		026	01	00001		(01) − 1 = 0?
212	0		017	00	00210		Return to 210, ≠ 0
213	0		005	00	00000		CLA 0, = 0
214	0		113	00	04000		Store in 4000
215	0		017	00	00215		Wait for interrupt
216	0		157	00	00300		Send CAL
217	0		017	00	02100		Error exit
220	0		157	00	00067		Send Query Interrupt
221	0		017	00	00223		Transfer
222	0		017	00	02100		Error exit
223	0		033	00	00301		Block transmit
224	0		143	00	00001		Instructions
225	0		017	00	00253		Error exit, block command
226	0		157	00	00016		Send transmit
227	0		017	00	02100		Error exit
230	0		033	02	01400		Block receive
231	0		120	10	00003		Data Character
232	0		017	00	00273		Exit for END, ERR test
233	0		017	00	02100		Error exit
234	0		157	00	00300		Send CAL
235	0		017	00	02100		Error exit

236	0		157	00	00061		Send QER
237	0		017	00	03200		To an error finder routine
240	0		157	00	00001		Send CLQ
241	0		017	00	02100		Error exit
242	0		002	02	00003		Add 3 to index 02
243	0		401	00	04000		Check for last row flag
244	0		017	00	00246		Transfer, > 0
245	0		017	00	00215		Wait for interrupt, ≤ 0
246	0		415	00	00302		CAL 77600000
247	0		515	00	01410		Extract (1410) Λ 77600000
250	0		161	00	00003		If (003) = (ACC)
251	0		017	00	02100		Error exit
252	0		017	00	03300		To another program
253	0		072	02	70000		Begin block command error
254	0		131	00	20000		Exit on QRD?
255	0		017	00	00261		Out to Not Ready routine
256	0		131	00	10000		Exit on QLR?
257	0		017	00	00270		Out to last row routine
260	0		017	00	03100		To interlock error routine
261	0		012	03	00002		Beginning of Not Ready routine
262	0		157	00	00060		Send QRD
263	0		017	00	00265		Transfer
264	0		017	00	00223		Return to block command
265	0		026	03	00001		(03) – 1 = 0?
266	0		017	00	00262		Return if not third time
267	0		017	00	02100		Error exit if still not ready
270	0		005	00	00001		Beginning of last row
271	0		113	00	04000		Store 1 in 4000
272	0		017	00	00226		Return to send Transmit
273	0		072	02	00377		Beginning of END – ERR
274	0		131	00	00004		= END?
275	0		017	00	00234		Transfer to CALL
276	0		131	00	00005		= ERR?
277	0		017	00	03100		Error type-out
300	0		017	00	02100		Unusual error exit
301	3		001	40	32462		CAL, QRD, QLR, QIL
302	0		007	76	00000		Mask

Explanation of the Read Cards Example Program:

200	0		012	02	00000		Set index 02 to zero
201	0		157	00	00300		Send CAL
202	0		017	00	02100		Error exit, unusual errors
203	0		157	00	00062		Send Query Interlock
204	0		017	00	03100		To interlock error routine
205	0		157	00	00056		Send start card feed
206	0		017	00	02100		Error exit

Locations 201 through 206 are used to start the card feed and position a card to be read. The command in location 201 sends the call to the card reader and the command in location 203 queries whether an interlocked condition exists in the card machine. The command in location 205 sends the start-card-feed line instruction and the card positioning begins. Locations 202 and 206 contain transfers to the unusual error routine. Location 204 contains a transfer to the interlock error routine.

207	0		012	01	00044		Set index 01 to 44
210	0		073	01	01377		Store zeros
211	0		026	01	00001		(01) – 1 = 0 ?
212	0		017	00	00210		Return to 210, $\neq 0$
213	0		005	00	00000		CLA 0, = 0
214	0		113	00	04000		Store in 4000
215	0		017	00	00215		Wait for interrupt

Locations 207 through 212 contain a short routine which is used to zero the card input area. The commands in locations 213 and 214 place a tag of zero in location 4000. Location 215 contains a transfer to itself. This is a wait for interrupt command.

The ISR will be entered when the interrupt occurs. The ISR will determine that the card reader sent the interrupt and will transfer to location 216 (see Chapter 1, Section 4).

216	0		157	00	00300		Send CAL
217	0		017	00	02100		Error exit
220	0		157	00	00067		Send Query Interrupt
221	0		017	00	00223		Transfer
222	0		017	00	02100		Error exit
223	0		033	00	00301		Block transmit
224	0		143	00	00001		Instructions
225	0		017	00	00253		Error exit, block command
226	0		157	00	00016		Send transmit
227	0		017	00	02100		Error exit

The command in location 216 sends the Call to the card reader. The command in location 220 queries whether an interrupt request was sent from the card reader. The commands in locations 223 and 224 send the line instructions CAL, QRD, QLR and QIL to the card reader. The command in location 226 sends the TRA line instruction to the card reader.

The commands contained in locations 217, 222 and 227 are transfers to the unusual error routine. Location 225 contains the exit for the block command.

230	0		033	02	01400		Block receive
231	0		120	10	00003		Data Character
232	0		017	00	00273		Exit for END, ERR test
233	0		017	00	02100		Error exit

The block transmit data command in location 230 and 231 causes the Central Processor to start the data transmission and then receive the data read from one row of the card. This data is placed in locations 1400 through 1402 for the first row. For the following iterations, data is put in locations 1403 through 1405 for the second row, and so on. The command in location 232 is a transfer to the END, ERR test routine which begins at location 273. Since the next command to be executed following the block command above is expected to be in location 232, an error exit is provided in location 233.

234	0		157	00	00300		Send CAL
235	0		017	00	02100		Error exit
236	0		157	00	00061		Send QER
237	0		017	00	03200		To an error finder routine
240	0		157	00	00001		Send CLQ
241	0		017	00	02100		Error exit

The commands in locations 234 through 241 are used to query for any errors during the last transmission and then to clear the query indicators. The command in location 234 sends the Call to the reader. The command in location 236 sends the query for error line instruction. And the command in location 240 sends the clear query indicators line instruction. The commands in locations 235 and 241 are the unusual error exits. The command in location 237 is the exit to the error-finding routine (this routine is not given in the example).

242	0		002	02	00003		Add 3 to index 02
243	0		401	00	04000		Check for last row flag
244	0		017	00	00246		Transfer, > 0
245	0		017	00	00215		Wait for interrupt, $\leq$ 0

Location 242 contains the index incrementer for the location of the incoming data. The command in location 243 checks for a one in location 4000. If a one is present, this means that the last row of the card has been read and the transfer in location 244 is executed. If another row is coming, the wait for interrupt command in location 245 is executed.

When the interrupt occurs for another card row, the ISR is entered. The ISR determines that the card reader sent the interrupt request and transfers back to location 216 (See Chapter 1, Section 4).

246	0		415	00	00302		CAL 77600000
247	0		515	00	01410		Extract (1410) Λ 77600000
250	0		161	00	00003		If (003) = (ACC)
251	0		017	00	02100		Error exit
252	0		017	00	03300		To another program

The commands in locations 246 through 252 are used to check for the 003 which should have been read from the card into location 1410. The command in location 252 transfers out of the card read routine.

253	0		072	02	70000		Begin block command error
254	0		131	00	20000		Exit on QRD?
255	0		017	00	00261		Out to Not Ready routine
256	0		131	00	10000		Exit on QLR?
257	0		017	00	00270		Out to last row routine
260	0		017	00	03100		To interlock error routine
261	0		012	03	00002		Beginning of Not Ready routine
262	0		157	00	00060		Send QRD
263	0		017	00	00265		Transfer
264	0		017	00	00223		Return to block command
265	0		026	03	00001		(03) − 1 = 0?
266	0		017	00	00262		Return if not third time
267	0		017	00	02100		Error exit if still not ready
270	0		005	00	00001		Beginning of last row
271	0		113	00	04000		Store 1 in 4000
272	0		017	00	00226		Return to send Transmit

The commands in locations 253 through 272 form an error exit routine for the block transmit instruction command found in locations 223 and 224. The command in location 253 brings the contents of bits 12, 13 and 14 of the Line Response register to the accumulator to be inspected. The command in location 254 tests whether the exit from the block command occurred on the QRD instruction. If so, the contents of location 255 transfers to the Not Ready subroutine. If not, the command in location 256 tests whether the exit from the block command occurred on the QLR instruction. If so, the contents of location 257 transfer to the Last Row subroutine. If not, the contents of location 260 transfer to some interlock error type-out routine (which is not given in this example).

The Not Ready routine begins at location 261. The commands in locations 261 through 267 send the QRD to the reader four times or until the reader becomes ready. If, in four times, the reader is still not ready, command in location 267 transfers out of the read routine.

The Last Row routine begins at location 267. The command in locations 267 through 272 store a one-tag in location 4000 to indicate when the last row of the card has been processed.

273	0		072	02	00377		Beginning of END − ERR
274	0		131	00	00004		= END?
275	0		017	00	00234		Transfer to CALL
276	0		131	00	00005		= ERR?
277	0		017	00	03100		Error type-out
300	0		017	00	02100		Unusual error exit

The commands in locations 273 through 300 form a routine to check whether the block transmit data command is terminated with an END or an ERR character from the reader. An error type-out is indicated in location 277. This typeout routine is not given in this example.

301	3		001	40	32462	CAL, QRD, QLR, QIL

Location 301 contains the line instructions CAL, QRD, QLR, and QIL.

10. CARD PUNCH SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program example considered will punch 3 cards using a card punch attached to a PC-10.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The call number for the card punch is to be 301.

The starting address for the program is 200.

The starting address for the storage of the instruction characters is 1100.

The starting address for the storage of the data to be transmitted is 1400.

An error routine is assumed to start in location 2100.

Another portion of the program in the Central Processor is assumed always to be in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and END characters and each of the REQ and interrupt signals. None of these characters or signals must be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading "PC-10."

In the following sample, decimal addresses for the command list have been used for simplicity.

200	0 (TLC) 157 00 (CAL) 00301	CAL	GRN — The TLC command transmits the "call" to the card punch with octal code 301. The card unit answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100		This command is the error exit for a non-GRN response.

202	0 (TLC) 157 00 (QIL) 00062	QIL	GRN	If an interlock condition exists at the initiation of a card punch operation, this QIL will detect it.
203	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
204	0 (TLC) 157 00 (STT) 00056	STT	GRN	The TLC command transmits the Start Card Feed (STT) instruction to the card punch.
				The card feed starts bringing the first card into the punch position and answers GRN.
205	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
206	0 (TRA) 017 00 02000			

The last TRA command transfers the Central Processor away from the input output program to another portion of the program while the card punch is positioning the first card. When the card is positioned, an interrupt will be sent to the Central Processor from the PC-10. The interrupt will be processed and the Central Processor will return (under program control) to the input/output program.

207	0 (TLC) 157 00 (CAL) 00301	CAL	GRN	The TLC command transmits a call to the card punch. The answer is GRN.
208	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
209	0 (TLC) 157 00 (QIN) 00067	QIN	RED	A QIN is sent to query whether an interrupt has been sent. The RED answer indicates that there has been one.
210	0 (TRA) 017 00 00212			This command skips one command.
211	0 (TRA) 017 00 02100			This command is the error exit for a non-RED response.
212	0 033 00 01100			The 033 command word is the first word of the block communication command.
213	0 14 (CAL) 301 00001			This command word, the second word of the block communication command, initiates the transmissions of the CAL, QRD, QLR and QIL instruction characters.

```
1100  (CAL) (QRD) (QLR) (QIL)
      3 0 1 1 4 0 3 2 4 6 2
```

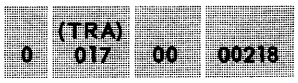
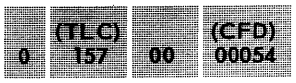
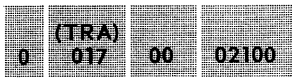
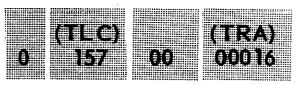
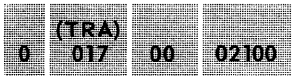
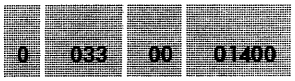
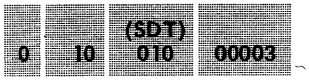
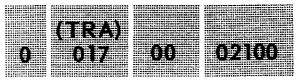
The line signal sequence is:

CAL	GRN	The first station on the PC-10 is called. Let the first called be the punch station. The PC-10 answers GRN.
CAL	GRN	
QRD	GRN	A QRD is sent to query whether or not the first station called is the ready station (or if any stations are still ready). A GRN response indicates that the first station called is the ready station. A RED response indicates that another station on the PC-10 must be called to determine the ready station.
QLR	GRN	A QLR is sent to query whether the current row is the last row of the card. A GRN answer indicates that this is not the last row. A RED answer indicates that this is the last row. The routine at the error exit must send the continue feed instruction to the PC-10 to send another card through. This routine is placed in location 216 for this example and the error routine at the exit of the block command will transfer to the Last Row routine when a RED response is encountered.
QIL	GRN	A QIL is sent to query whether an interlock condition exists at this time. A GRN response indicates that there is no interlock; a RED response indicates an interlock.

```
214   0  (TRA) 017   00   02100
```

This command is the error exit for a non-GRN response during the data transmission.

(Each of the above instructions can be transmitted using a TLC command and therefore allow each to have a different error exit for easier processing.)

215				This command is used to skip the CFD on a non-last-row pass through the routine.
216		CFD	GRN	The CFD instruction is sent at this time to feed the next card to be punched.
217				This command is the error exit for a non-GRN response.
218		RCV	GRN	The RCV instructs the card punch to be the receiver in the data communication to follow.
219				This command is the error exit for a non-GRN response.
220				The 033 command is the first word of the block communication command.
221				This command word, the second word of the block communication command, initiates the transmission of the data row to the card punch.
222				This command is the error exit for a non-data response.
		SDT	REQ	The Central Processor informs the card punch that the data transmission has started. A request signal is sent by the card punch on the REQ line to request the first data character.
		ddd	REQ	The Central Processor answers with a data character. A request signal is sent by the PC-10 on the REQ signal line to request the next data character.
		ddd	REQ	The Central Processor answers with the next data character. A request for the next data character is sent.

The data transmission continues through the next to last data character of that row. Then:

	ddd	REQ	The Central Processor answers with the last data character. A request for another data character is sent.

	END		An END instruction character is sent from the Central Processor to the PC-10 (and the card punch) to denote the end of the data transmission. No reply is sent from the PC-10.

As shown above, the block command in locations 215 and 216 causes the Central Processor to automatically transmit to the PC-10 the ten 8-bit data characters found in locations 1400, 1401 and 1402. The first ten characters are accepted and processed by the card punch; the next two characters initiate the REQ response but otherwise are ignored.

1400	First Character	Second Character	Third Character	Fourth Character
1401	Fifth Character	Sixth Character	Seventh Character	Eighth Character
1402	Ninth Character	Tenth Character	0	0

223	0 (TRA) 157 00 (CAL) 00301	CAL	GRN	The card unit is called, answers GRN and enters the CALLED state.
224	0 (TRA) 017 00 02100			This command is the error exit.
225	0 (TLC) 157 00 (QER) 00061	QER	GRN	The QER is sent to query whether any error other than interlock has occurred.
226	0 (TRA) 017 00 02100			This command is the error exit.
227	0 (TLC) 157 00 (CLQ) 00001	CLQ	GRN	The CLQ must be sent to reset all error indicators as well as the interrupt indicator.
228	0 (TRA) 017 00 02100			This is an error exit.

At this point the programmer can set up the counter to count the number of cards to be processed. In this example the counter will count three words. After the necessary bookkeeping, the programmer has some time left before the interrupt occurs for the next row. When the interrupt occurs, the program should transfer back to location 207 to initiate the punching of the next row.

11. CARD PUNCH EXAMPLE PROGRAM

The following example program will use a card punch attached to the PC-10 to punch one card. A punch will be put in every row of column 80 and nothing else will be punched in the card. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		012	02	00000		0 into (02)
201	0		157	00	00301		Send CAL
202	0		017	00	02100		Error exit
203	0		157	00	00062		Send Query Interlock
204	0		017	00	03100		To interlock error
205	0		157	00	00056		Start Card Feed
206	0		017	00	02100		Error exit
207	0		005	00	00000		0 into (ACC)
210	0		113	00	04000		0 into (4000)
211	0		017	00	00211		Wait for interrupt
212	0		157	00	00301		Send CAL
213	0		017	00	02100		Error exit
214	0		157	00	00067		Send Query Interrupt
215	0		017	00	00217		Transfer
216	0		017	00	02100		Error exit
217	0		033	00	00261		} Block transmit
220	0		143	01	00001		} instructions
221	0		017	00	00241		Error exit, block command
222	0		157	00	00014		Send Receive
223	0		017	00	02100		Error exit
224	0		033	02	00262		} Block transmit data
225	0		120	10	00003		} characters
226	0		017	00	02100		Error exit
227	0		157	00	00301		Send CAL
230	0		017	00	02100		Error exit

LOCATION	F 31 30	M 29 28	OP 27 21	I 20 15	A Field 14 0	u	Comments
231	0		157	00	00061		Send QER
232	0		017	00	03200		To error detecting routine
233	0		157	00	00001		Send CLQ
234	0		017	00	02100		Error exit
235	0		002	02	00003		Add 3 to index 02
236	0		401	00	04000		Check for last row flag
237	0		017	00	03300		Out to new program
240	0		017	00	00211		To wait for interrupt
241	0		072	02	70000		Begin block command exit
242	0		131	00	20000		Exit on QRD?
243	0		017	00	00252		Yes, to Not Ready routine
244	0		131	00	10000		Exit on QLR
245	0		017	00	00247		Yes, to last row
246	0		017	00	03100		Error type-out routine
247	0		005	00	00001		Beginning last row
250	0		113	00	04000		Store 1 in 4000
251	0		017	00	00222		Transfer
252	0		012	02	00002		Beginning Not Ready
253	0		157	00	00060		Send QRD
254	0		017	00	00256		Transfer
255	0		017	00	00217		Return to block command
256	0		026	02	00001		(02) – 1 = 0?
257	0		017	00	00253		Return if not third time
260	0		017	00	02100		Out if still not ready
261	3		011	40	32462		CAL, QRD, QLR, QIL

LOCATION	F (31)	M (30-29)	OP (28-21)	I (20-15)	A Field (14-0)	u	Comments
262	0		000	00	00000		
263	0		000	00	00000		
264	0		000	02	00000		
265	0		000	00	00000		
266	0		000	00	00000		
267	0		000	02	00000		
.			.		.		
.			.		.		
.			.		.		
323	0		000	00	00000		
324	0		000	00	00000		
325	0		000	02	00000		

Explanation of the card punch example program

200	0		012	02	00000		0 into (02)
201	0		157	00	00301		Send CAL
202	0		017	00	02100		Error exit
203	0		157	00	00062		Send Query Interlock
204	0		017	00	03100		To interlock error
205	0		157	00	00056		Start Card Feed
206	0		017	00	02100		Error exit
207	0		005	00	00000		0 into (ACC)
210	0		113	00	04000		0 into (4000)
211	0		017	00	00211		Wait for interrupt

The commands in locations 201 through 206 are used to start the card feed and position a card to be read. The command in location 201 sends the Call to the card punch and the command in location 203 queries whether an interlocked condition exists in the card machine. The command in location 205 sends the start card feed line instruction and the card positioning begins. Locations 202 and 206 contain transfers to the unusual error routine. Location 204 contains a transfer to the interlock error routine.

The commands in locations 207 and 210 place a tag of zero in location 4000. Location 211 contains a transfer to itself. This is a wait for interrupt command (See Chapter 1, Section 4).

The ISR will be entered when the interrupt occurs. The ISR will determine that the card punch sent the interrupt and will transfer to location 212.

212	0	157	00	00301		Send CAL
213	0	017	00	02100		Error exit
214	0	157	00	00067		Send Query Interrupt
215	0	017	00	00217		Transfer
216	0	017	00	02100		Error exit
217	0	033	00	00261		Block transmit
220	0	143	01	00001		instructions
221	0	017	00	00241		Error exit, block command
222	0	157	00	00014		Send Receive
223	0	017	00	02100		Error exit

The command in location 212 sends the Call to the card punch. The command in location 214 queries whether an interrupt request was sent from the card punch. The commands in locations 217 and 220 send the line instructions CAL, QRD, QLR, and QIL to the card punch. The command in location 222 sends the RCV line instruction to the card punch.

The commands contained in locations 213, 216 and 223 are transfers to the unusual error routine. Location 221 contains the exit for the block command.

224	0	033	02	00262		Block transmit data
225	0	120	10	00003		characters
226	0	017	00	02100		Error exit

The block transmit data command in location 224 and 225 causes the Central Processor to start the data transmission and then send the data for one row of the card to the card punch. This data is taken from locations 262 through 264 for the first row. The next time through the routine, data is taken from locations 265 through 267 for the second row, and so on. Location 226 contains the unusual error exit.

227	0		157	00	00301		Send CAL
230	0		017	00	02100		Error exit
231	0		157	00	00061		Send QER
232	0		017	00	03200		To error detecting routine
233	0		157	00	00001		Send CLQ
234	0		017	00	02100		Error exit
235	0		002	02	00003		Add 3 to index 02
236	0		401	00	04000		Check for last row flag
237	0		017	00	03300		Out to new program
240	0		017	00	00211		To wait for interrupt

The commands in locations 227 through 234 are used to query for any errors during the last transmission and then to clear the query indicators. The command in location 227 sends the Call to the punch. The command in location 231 sends the query for error line instruction. And the command in location 233 sends the clear query indicators line instruction. The commands in locations 230 and 234 are the unusual error exits. The command in location 232 is the exit to the error-finding routine (this routine is not given in the example).

Location 235 contains the index incrementer for the location of the outgoing data. The command in location 236 checks for a one in location 4000. If a one is present, the transfer to some other program in location 237 is executed. Location 240 contains a transfer to the wait for interrupt command at location 211 (See Chapter 1, Section 4).

When the interrupt occurs for another card row, the ISR is entered. The ISR determines that the card punch sent the interrupt request and transfers to location 212.

241	0		072	02	70000		Begin block command exit
242	0		131	00	20000		Exit on QRD?
243	0		017	00	00252		Yes, to Not Ready routine
244	0		131	00	10000		Exit on QLR
245	0		017	00	00247		Yes, to last row
246	0		017	00	03100		Error type-out routine
247	0		005	00	00001		Beginning last row
250	0		113	00	04000		Store 1 in 4000
251	0		017	00	00222		Transfer
252	0		012	02	00002		Beginning Not Ready
253	0		157	00	00060		Send QRD
254	0		017	00	00256		Transfer
255	0		017	00	00217		Return to block command
256	0		026	02	00001		(02) − 1 = 0?
257	0		017	00	00253		Return if not third time
260	0		017	00	02100		Out if still not ready

The commands in locations 241 through 260 form an error exit routine for the block transmit instructions command found in locations 217 and 220.

The command in location 241 brings the contents of bits 12, 13 and 14 of the Line Response register to the accumulator to be inspected. The command in location 242 tests whether the exit from the block command occurred on the QRD instruction. If so, the contents of location 243 transfers to the Not Ready subroutine. If not, the command in location 244 tests whether the exit from the block command occurred on the QLR instruction. If so, the contents of location 245 transfers to the Last Row subroutine. If not, the contents of location 246 transfer to some interlock error typeout routine (which is not given in this example).

The Last Row routine begins at location 247. The commands in location 247 through 251 store a one-tag in location 4000 to indicate when the last row of the card has been processed.

The Not Ready routine begins at location 252. The commands in locations 252 through 260 send the QRD to the punch three times or until the punch becomes ready. If, in three times, the punch is still not ready, the command in location 260 transfers out of the punch routine.

261	3		011	40	32462		CAL, QRD, QLR, QIL

Location 261 contains the line instructions CAL, QRD, QLR and QIL.

262	0		000	00	00000		
263	0		000	00	00000		
264	0		000	02	00000		
265	0		000	00	00000		
266	0		000	00	00000		
267	0		000	02	00000		
.			.		.		
.			.		.		
.			.		.		
323	0		000	00	00000		
324	0		000	00	00000		
325	0		000	02	00000		

The data to be sent to the punch for the one card is stored in locations 262 through 325.

12. LINE PRINTER SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program example considered will print one line of alphanumeric print using the 72-column LP-10 attached to a PC-10.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are included as they occur in memory. The call number for the line printer is to be 302.

The starting address for the program is 200.

The starting address for the storage of the instruction characters is 1100.

The starting address for the storage of the data to be transmitted is 1400.

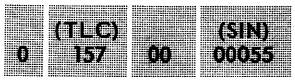
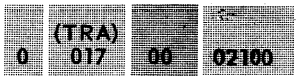
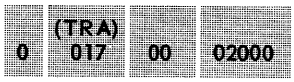
An error routine is assumed to start in location 2100.

Another program in the Central Processor is assumed always to be in location 2000.

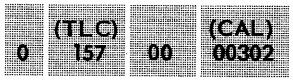
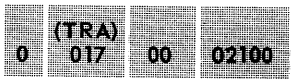
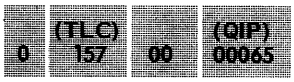
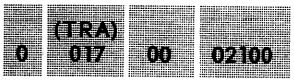
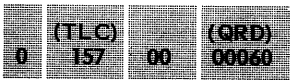
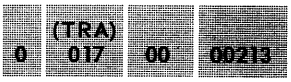
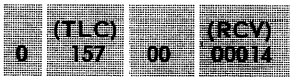
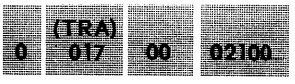
The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and END characters, and each of the REQ and interrupt signals. None of these characters or signals must be programmed. The instruction or signal under the heading "Central Processor" initiates the action, and the response, which comes later in time, is under the heading "PC-10."

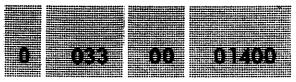
In the following sample, decimal addresses in the command sequence have been used for simplicity.

		Central Processor	PC-10	
200	0 (TLC) 157 00 (CAL) 00302	CAL	GRN	The TLC command transmits the "call" to the line printer with call number 302. The line printer answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
202	0 (TLC) 157 00 (CLQ) 00001	CLQ	GRN	The CLQ insures the resetting of all interrupt and error indicators.
203	0 (TRA) 017 00 02100			This command is an error exit.
204	0 (TLC) 157 00 (PAN) 00050	PAN	GRN	The TLC command transmits the Print Alphanumeric (PAN) instruction to the line printer. The line printer prepares to print alphanumeric. The printer answers GRN.
205	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.

206		SIN	GRN	The TLC command transmits the Send Interrupt (SIN) instruction to the line printer. An interrupt will be sent from the line printer at "index pulse time". The printer answers GRN.
207				This command is the error exit for a non-GRN response.
208				This command transfers the Central Processor from the input/output program while the print wheel is coming into position to print.

At "index pulse time", an interrupt will be sent to the Central Processor from the PC-10. The interrupt service routine will determine that the interrupt came from the line printer and will transfer control to the following command sequence of the input/output program.

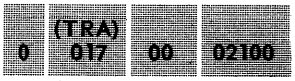
209		CAL	GRN	The TLC command transmits a Call to the line printer with Call number 302. The line printer answers GRN and enters the CALLED state.
210				This command is the error exit for a non-GRN response.
211		QIP	GRN	The TLC command transmits the Query Index Pulse instruction to the line printer. The answer is GRN.
212				This command is the error exit for a non-GRN response.
213		QRD	GRN	The TLC command transmits the Query Ready instruction to the line printer. The answer is GRN.
214				This command is the error exit for a non-GRN RESPONSE. A counter should be put in here to avoid completely "hanging up" on the QRD if the printer does not become ready.
215		RCV	GRN	The TLC command transmits the Receive instruction to the line printer. The answer is GRN.
216				This command is the error exit for a non-GRN response.

217 

The 033 command word is the first word of the block communication command.

218 

This command word, the second word of the block communication command, initiates to the line printer the transmission of one 9-character row of the print image.

219 

This command is the error exit for a non-GRN response.

The line signals during the transmission are:

SDT	REQ	The Central Processor informs the line printer that the data transmission has started. A request signal is sent by the line printer on the REQ signal line to request the first data character.
ddd	REQ	The Central Processor answers with the first data character from the print image. A request for the next data character is sent.
ddd	REQ	The Central Processor answers with a data character from the print image. A request for the next data character is sent.

The data transmission continues through the next to last data character in that row of the print image. Then:

ddd	REQ	The Central Processor answers with the last data character in that row of the print image. A request for another character is sent.
END		An END instruction character is sent from the Central Processor to the PC-10 (and the line printer) to denote the end of the data transmission. No reply is sent from the PC-10. The PC-10 returns to STANDBY.

As shown above, the block command causes the Central Processor to automatically transmit to the PC-10 the twelve 8-bit characters found in locations 1400, 1401 and 1402. The first nine characters are accepted and processed by the line printer. The next three characters initiate the REQ response but otherwise are ignored.

1400	First Character	Second Character	Third Character	Fourth Character
1401	Fifth Character	Sixth Character	Seventh Character	Eighth Character
1402	Ninth Character	0	0	0

The PC-10 goes to STANDBY and the line printer finishes printing that part of the line of type. The next row of the print image can be placed in locations 1400, 1401 and 1402; or, an index can be used to increment location 1400 by three on each return.

220	0 (TLC) 157 00 (CAL) 00302	CAL		GRN	The TLC command transmits the call to the line printer with call number 102. The PC-10 enters the CALLED state.

The Central Processor (under program control) returns to location 213 62 times to complete the transmission of the print image. Then:

221	0 033 00 01100				The 033 command word is the first word of the block communication command.
222	0 14 (CAL) 302 00001				This command word, the second word of the block communication command, initiates the transmission of the CAL, QER, STT, and SIN instructions.
		1100	(QER) (CLQ) (STT) (SIN) 0 6 1 0 0 2 3 0 4 5 5		
		CAL		GRN	The line printer is called and answers GRN.
		QER		GRN	The line printer is queried to see if there have been any errors.
		CLQ		GRN	The CLQ resets all query indicators.
		STT		GRN	The line printer is instructed to start the paper feed.
		SIN		GRN	The SIN sets the send interrupt on for the next line of print.
223	0 (TRA) 017 00 02100				This command is the error exit for a non-GRN response.

The Central Processor (under program control) returns to another program. If another line is to be printed, the print image for the line of print will be constructed at this time.

13. LINE PRINT EXAMPLE PROGRAM

The following sample will assemble the necessary print image and then use a line printer attached to the PC-10 to print the following:

G-20

PRIN with a four space left margin

The print image assembly routine begins in location 200. The line print routine begins in location 272. The minimum ISR is assumed to be in memory for use with this example.

A flow chart for the part of the print routine which constructs the print image is given on page 69 . Locations 200 through 271 are used to construct the print image; the rest of the program is the actual printing routine. The figure on page 70 shows in general how the print image is constructed with the print image making routine (locations 200 through 271).

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		012	03	00001		Begin print image maker
201	0		012	02	01673		1673 → (02)
202	0		073	02	01377		0 → 1377 + (02)
203	0		026	02	00001		(02) − 1 = 0?
204	0		017	00	00202		No, To 202
205	0		012	01	00000		Yes, 0 → (01)
206	0		012	04	00000		0 → (04)
207	0		002	04	00001		(04) + 1 → (04)
210	0		002	01	00001		(01) + 1 → (01)
211	0		415	03	04000		CAL (4000) + (03)
212	0		531	00	00237		(ACC)= 30060140300
213	0		017	00	00244		To last line test
214	0		477	04	00232		Shifter
215	0		515	00	00231		Extract 8-bit character
216	0		151	00	00000		(ACC) = 0?
217	0		017	00	00251		Yes, transfer
220	0		040	00	05000		5000+ (ACC) into (0A)
221	0		405	00	00000		CLA (0A)
222	0		113	00	00236		Put (ACC) into Ti
223	0		401	01	00241		Operand > 0?
224	0		017	00	00257		Yes, transfer
225	0		405	01	05200		No, CLA (5200)
226	0		755	00	00236		(ACC) $\vee$ (Ti) into (ACC)
227	0		573	00	00236		(ACC) into (Ti)
230	0		017	00	00206		Transfer

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
231	0		000	00	00377		Extractor
232	0		110	00	00001		Multiplier-shift 3 8-bit
233	0		106	00	00004		Multiplier-shift 2 8-bit
234	0		103	00	00002		Multiplier-shift 1 8-bit
235	0		000	00	00001		Multiplier - shift
236	0		000	00	00000		Temporary storage Ti
237	3		006	01	40300		End of line code
240	2		004	01	00200		End of print code
241	0		200	00	00032		Tester -32
242	0		200	00	00064		Tester -64
243	0		200	00	00072		Tester -72
244	0		002	03	00001		Add 1 to (03)
245	0		415	03	04000		CAL (03) + (4000)
246	0		531	00	00240		(ACC) = End print code
247	0		017	00	03000		Leave print routine
250	0		017	00	00272		To print routine
251	0		551	04	00254		0 $\vee$ (04) -4 = 0?
252	0		017	00	00255		Transfer to 255
253	0		017	00	00207		Transfer to 207
254	0		200	00	00004		Tester -4
255	0		002	03	00001		Add 1 to (03)
256	0		017	00	00206		Transfer
257	0		401	01	00242		Operand > 0?
260	0		017	00	00265		Yes, transfer
261	0		405	00	00236		CLA (236)

LOCATION	F 31 30	M 29 28	OP 27 21	I 20 15	A Field 14 0	u	Comments
262	0		045	00	00001		Add 1 to (ACC)
263	0		113	00	00236		STS 236
264	0		017	00	00225		Transfer to 225
265	0		401	00	00243		Operand > 0?
266	0		017	00	02100		To unusual error
267	0		405	00	00236		CLA (Ti)
270	0		045	00	00002		Add 2 to (ACC)
271	0		017	00	00263		Transfer to 263
272	0		012	05	00000		Begin Printer routine
273	0		157	00	00302		Send CAL
274	0		017	00	02100		Error exit
275	0		157	00	00001		Send CLQ
276	0		017	00	02100		Error exit
277	0		157	00	00050		Send PAN
300	0		017	00	02100		Error exit
301	0		157	00	00055		Send SIN
302	0		017	00	02100		Error exit
303	0		017	00	00303		Wait for interrupt
304	0		157	00	00302		Send CAL
305	0		017	00	02100		Error exit
306	0		157	00	00065		Send QIP
307	0		017	00	00346		To Call again routine
310	0		157	00	00060		Send QRD
311	0		017	00	00353		To Query again routine
312	0		157	00	00014		Send RCV

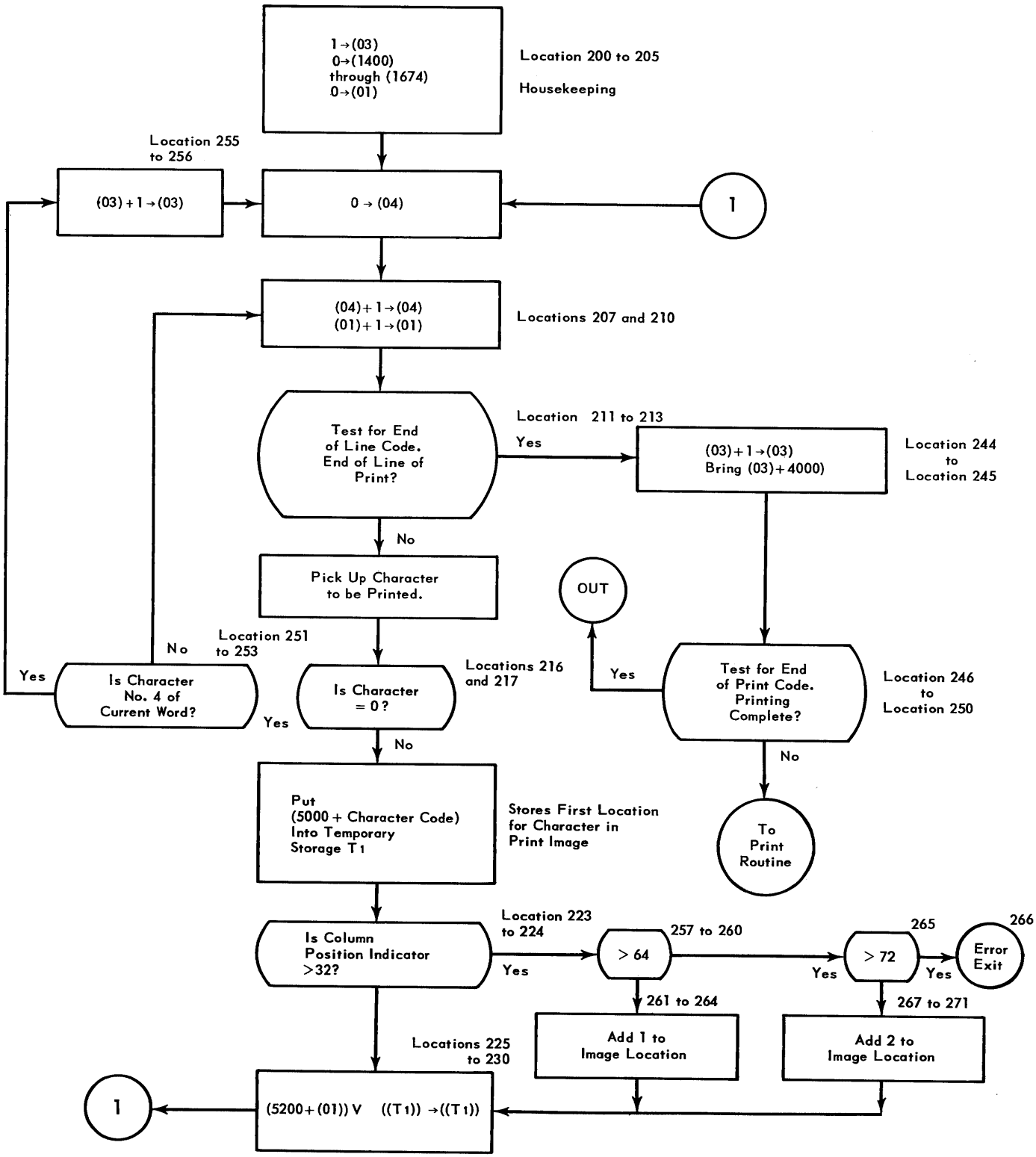
LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
313	0		017	00	02100		Error exit
314	0		033	05	01400		Block transmit data
315	0		100	10	00003		Characters
316	0		017	00	02100		Error exit
317	0		002	05	00003		Add 3 to (05)
320	0		405	00	00005		CLA (05)
321	0		121	00	00275		If less than 275
322	0		017	00	00343		Transfer
323	0		157	00	00302		Send CAL
324	0		017	00	02100		Error exit
325	0		157	00	00061		Send QER
326	0		017	00	02100		Error exit
327	0		157	00	00001		Send CLQ
330	0		017	00	02100		Error exit
331	0		157	00	00055		Send STT to upspace
332	0		017	00	02100		Error exit
333	0		157	00	00054		Send CFD, upspace second line
334	0		017	00	02100		Error exit
335	0		017	00	00335		Wait interrupt
336	0		157	00	00302		Send CAL
337	0		017	00	02100		Error exit
340	0		157	00	00001		Send CLQ
341	0		017	00	02100		Error exit
342	0		017	00	00201		To print image builder
343	0		157	00	00302		Send CAL

LOCATION	F 31 30	M 29 28	OP 27 21	I 20 15	A Field 14 0	u	Comments
344	0		017	00	02100		Error exit
345	0		017	00	00310		Transfer to 310
346	0		157	00	00001		Send CLQ
347	0		017	00	02100		Error exit
350	0		157	00	00065		Send QIP
351	0		017	00	00350		Return to QIP
352	0		017	00	00310		Transfer to 310
353	0		012	06	00004		4 → (06)
354	0		157	00	00060		Send QRD
355	0		017	00	00357		Transfer to 357
356	0		017	00	00312		Transfer to 312
357	0		026	06	00001		(06) - 1 = 0?
360	0		017	00	00354		Transfer
363			017	00	02100		Error exit
5001	0		000	00	01403		Relates A to print image location
5002	0		000	00	01411		Relates B to print image location
5003	0		000	00	01417		Relates C to print image location
5004	0		000	00	01425		Relates D to print image location
.			.		.		.
.			.		.		.
.			.		.		.
5076	0		000	00	01664		Relates : to print image location
5077	0		000	00	01672		Relates ' to print image location
5200	2		000	00	00000		Location column 1, word 1
5201	1		000	00	00000		Locator Column 2, word 1

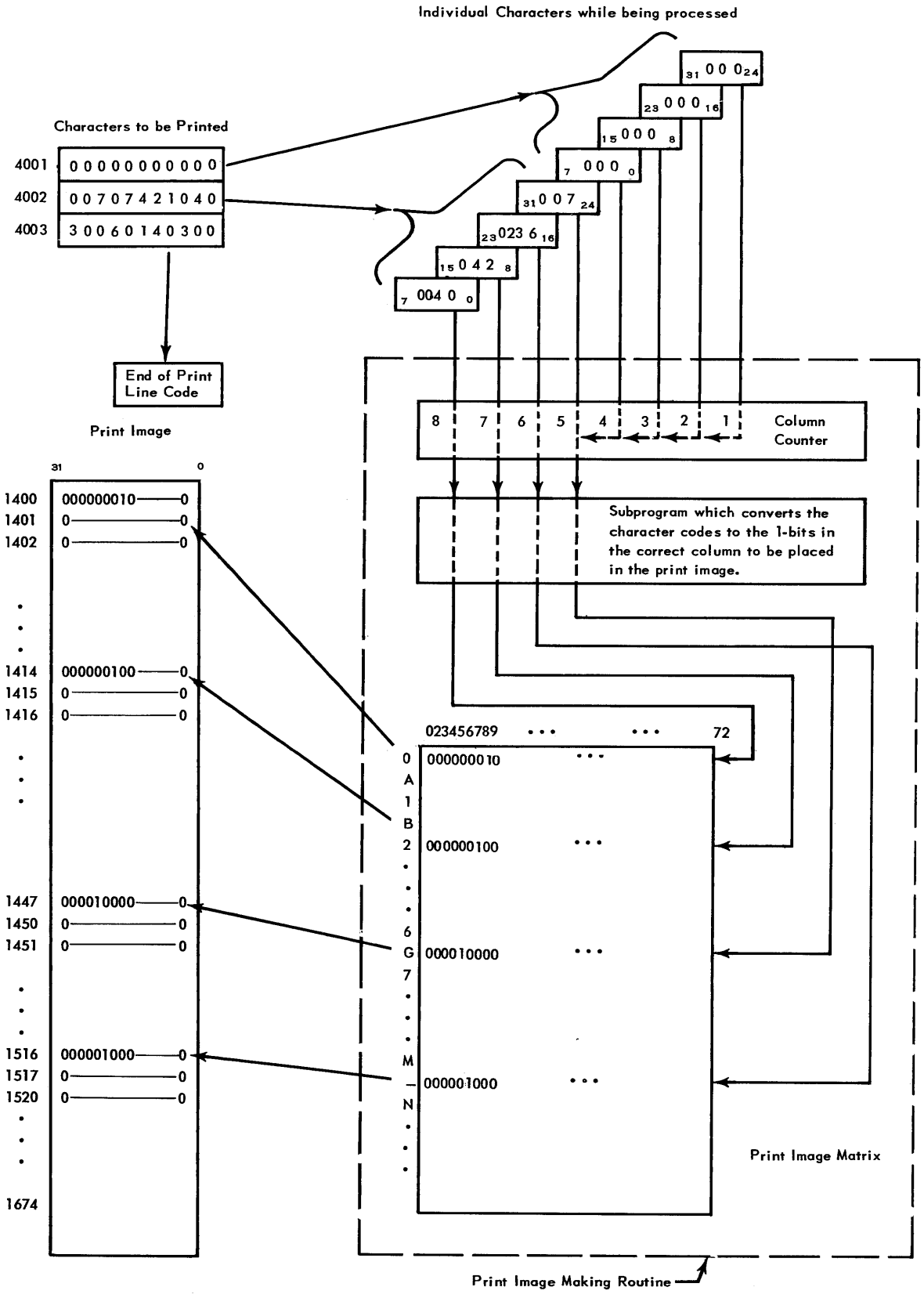
LOCATION	F 31 30	M 29 28	OP 27 21	I 20 15	A Field 14 0	u	Comments
5202	0		400	00	00000		Locator column 3, word 1
.			.		.		.
.			.		.		.
.			.		.		.
5306	0		000	04	00000		Locator column 16, word 3
5307	0		000	02	00000		Locator column 17, word 3
4001	0		000	00	00000		Four spaces for print row 1
4002	0		070	74	21040		8-bit characters for "G-20"
4003	3		006	01	40300		End of row code
4004	0		000	00	00000		Four spaces for print row 2
4005	0		200	44	04416		8-bit characters for "PRIN"
4006	3		006	01	40300		End of row code
4007	2		004	01	00200		End print out code
			Print Image Assembled for "G-20"				
1400	0		010	00	00000		
1401	0		000	00	00000		To be sent for "0"
1402	0		000	00	00000		
:			:		:		All entries zero
1414	0		020	00	00000		
1415	0		000	00	00000		To be sent for "2"
1416	0		000	00	00000		
:			:		:		All entries zero
1447	0		100	00	00000		
1450	0		000	00	00000		To be sent for "G"
1451	0		000	00	00000		

LOCATION	F (31)	M (30–29)	OP (28–27 … 21)	I (20 … 15)	A Field (14 … 0)	u	Comments
⋮			⋮		⋮		All entries zero
1516	0		040	00	00000		
1517	0		000	00	00000		To be sent for "—"
1520	0		000	00	00000		
⋮			⋮		⋮		All entries zero
1674	0		000	00	00000		
			Print Image Assembled for "PRIN"				
1400	0		000	00	00000		All entries zero
⋮			⋮		⋮		
1462	0		020	00	00000		
1464	0		000	00	00000		To be sent for "I"
1465	0		000	00	00000		
⋮			⋮		⋮		All entries zero
1521	0		010	00	00000		
1522	0		000	00	00000		To be sent for "N"
1523	0		000	00	00000		
⋮			⋮		⋮		All entries zero
1535	0		100	00	00000		
1536	0		000	00	00000		To be sent for "P"
1537	0		000	00	00000		
⋮			⋮		⋮		All entries zero
1551	0		400	00	00000		
1552	0		000	00	00000		To be sent for "R"
1553	0		000	00	00000		
⋮			⋮		⋮		All entries zero
1674	0		000	00	00000		

PRINT IMAGE ASSEMBLY ROUTINE



FUNCTIONAL DIAGRAM OF PRINT IMAGE ROUTINE



The preceding figure gives a schematic representation of the first print image constructed by the example program. Only the print image for the line print "G-20" is shown. In the upper left part of the figure in 4001 are shown the set of four all zero 8-bit characters which represent the four spaces to be left on the page of print; in 4002 are shown the four characters which represent G-20; in 4003 is the end of print line code set up by the program parameters. To the right is shown the eight characters as they are processed. The print image making routine shown inside the dotted lines sets up the print image matrix and stores it in the print image area, locations 1400_8 through 1674_8.

Explanation of the Print Image Assembly Routine

200	0	012	03	00001		Begin print image maker
201	0	012	02	01673		1673 → (02)
202	0	073	02	01377		0 → 1377 + (02)
203	0	026	02	00001		(02) − 1 = 0?
204	0	017	00	00202		No, To 202
205	0	012	01	00000		Yes, 0 → (01)
206	0	012	04	00000		0 → (04)

The commands in locations 200 through 206 are used to set the indexes to be used in the routine and to zero the area in which the print image will be placed.

207	0	002	04	00001		(04) + 1 → (04)
210	0	002	01	00001		(01) + 1 → (01)

The commands in locations 207 and 210 are used to increment indexes 04 and 01 during the running of the routine.

211	0	415	03	04000		CAL (4000) + (03)
212	0	531	00	00237		(ACC)= 30060140300
213	0	017	00	00244		To last line test

Characters to be printed are assumed to be presented to this print image routine in the following form:

8-bit characters packed four to a word

The characters packed in contiguous words

A code word to indicate the end of a print line placed after the last word with print characters in it.

A code word to indicate the end of the entire printing placed after the last end of print line code word.

The list of characters to be printed in the above format begin in location 4001.

The commands in locations 211 and 212 pick up a word in the printout block and test for the end of print line code. If the word contains the end of print line code, the command in location 213 transfers to an end printout code test.

214	0		477	04	00232		Shifter
215	0		515	00	00231		Extract 8-bit character
216	0		151	00	00000		(ACC) = 0?
217	0		017	00	00251		Yes, transfer

The commands in locations 214 and 215 place in the accumulator the next character to be processed into the print image. The command in location 216 tests the contents of the accumulator for a blank. If the character is a blank, an exit to location 251 is found in location 217.

220	0		040	00	05000		5000+ (ACC) into (OA)
221	0		405	00	00000		CLA (OA)
222	0		113	00	00236		Put (ACC) into Ti

A look-up table is assumed to be stored in locations 5001 through 5077. The location of the table entry corresponds to the G-20 8-bit internal code (i.e. 5000 + code) of the character to be printed. The contents of the entry location is the address of the first word of the print image for the specified print character. For example, if the character in the accumulator is the letter G with the internal code of character 007, then the contents of location 5007 is 1447. And 1447 is the position in the print image where the three-word set of 72 bits, which will be sent to the line printer for the letter F, is placed by this routine.

The commands in locations 220 through 222 place the contents of the location formed by 5000 + the character code into a temporary storage location.

223	0		401	01	00241		Operand > 0?
224	0		017	00	00257		Yes, transfer

The command in location 223 checks the contents of index 01 to test whether the character being placed in the print image should be placed in the first of the three words (i.e. should it be printed in a column less than or equal to 32). If not, the transfer in location 224 transfers to a test for the second and third words.

225	0	405	01	05200		No, CLA (5200)
226	0	755	00	00236		(ACC) $\vee$ (Ti) into (ACC)
227	0	573	00	00236		(ACC) into (Ti)
230	0	017	00	00206		Transfer

A second look-up table is assumed to be stored in location 5200 through 5307. The location of the table entry is an address made by adding the contents of index 01 to 5200. Index 01 keeps a count of which print column is currently being built in the print image. The entry location contains a one-bit in the position, counting from the most significant end, which matches the current contents of index 01. All other entries in the word are zeros.

Locations 225 through 227 are used to place a one-bit in the print image in the location specified by the character currently being processed and in the column position specified by the count in index 01.

The transfer in location 230 returns control to the beginning of the program to process another character.

231	0	000	00	00377		Extractor
232	0	110	00	00001		Multiplier-shift 3 8-bit
233	0	106	00	00004		Multiplier-shift 2 8-bit
234	0	103	00	00002		Multiplier-shift 1 8-bit
235	0	000	00	00001		Multiplier - shift
236	0	000	00	00000		Temporary storage Ti
237	3	006	01	40300		End of line code
240	2	004	01	00200		End of print code
241	0	200	00	00032		Tester -32
242	0	200	00	00064		Tester -64
243	0	200	00	00072		Tester -72

Locations 231 through 243 contain the various constants needed to aid in processing the print characters.

244	0	002	03	00001		Add 1 to (03)
245	0	415	03	04000		CAL (03) + (4000)
246	0	531	00	00240		(ACC) = End print code
247	0	017	00	03000		Leave print routine
250	0	017	00	00272		To print routine

The commands in locations 244 through 246 check whether the end of print-out code has been found. If so, the command in location 247 transfers out of the printing routine to another program. If not, the print routine is entered using the command in location 250.

251	0	551	04	00254		0 ∨ (04) -4 = 0?
252	0	017	00	00255		Transfer to 255
253	0	017	00	00207		Transfer to 207
254	0	200	00	00004		Tester -4

The commands in locations 251 through 253 check whether the character being processed is the fourth character in the current word. If so, control goes to location 255. If not, control returns to location 207.

255	0	002	03	00001		Add 1 to (03)
256	0	017	00	00206		Transfer

The commands in location 255 and 256 increment the print character pick-up and then transfer to location 206.

257	0	401	01	00242		Operand > 0?
260	0	017	00	00265		Yes, transfer

The command in location 257 checks the contents of index 01 to test whether the character being placed in the print image should be placed in the second of the three words (i.e., should it be printed in a column greater than 32 and less than 65). If not, the transfer in location 260 transfers to a test for the third word. If so, control goes to location 261.

261	0	405	00	00236		CLA (236)
262	0	045	00	00001		Add 1 to (ACC)
263	0	113	00	00236		STS 236
264	0	017	00	00225		Transfer to 225

The commands in locations 261 through 263 are used to increment the contents of the temporary storage location 236 by one. This increases the address of the print image entry by one so that a print character will be entered into the print image in a column between 32 and 65. The command in location 264 transfers control to location 225.

265	0		401	00	00243		Operand > 0?
266	0		017	00	02100		To unusual error
267	0		405	00	00236		CLA (Ti)
270	0		045	00	00002		Add 2 to (ACC)
271	0		017	00	00263		Transfer to 263

The command in location 265 checks the contents of index 01 to test whether the character being placed in the print image should be placed in the third of the three words (i.e., should it be printed in a column greater than 64 and less than 73). If not, the transfer in location 266 places control in an error routine. If so, the commands in locations 267 through 270 add 2 to the address of the print image entry for the character being processed so that the character will be entered in the print image in a column between 64 and 73. The command in location 271 transfers control to location 263.

Explanation of the Line Print Routine

272	0		012	05	00000		Begin Printer routine
273	0		157	00	00302		Send CAL
274	0		017	00	02100		Error exit
275	0		157	00	00001		Send CLQ
276	0		017	00	02100		Error exit
277	0		157	00	00050		Send PAN
300	0		017	00	02100		Error exit
301	0		157	00	00055		Send SIN
302	0		017	00	02100		Error exit
303	0		017	00	00303		Wait for interrupt

The commands in locations 273 through 303 are used to call the line printer, clear the indicators in the line printer, set the printer to print alphanumeric characters and transmit the SIN line instruction. Locations 274, 276, 300 and 302 are unusual error exits. The command in location 303 is a transfer to itself which is used to wait for the Index Pulse interrupt request. (See Chapter 1, Section 4).

The ISR will be entered when the interrupt occurs. The ISR will determine that the line printer sent the interrupt and will transfer to location 304.

304	0		157	00	00302		Send CAL
305	0		017	00	02100		Error exit
306	0		157	00	00065		Send QIP
307	0		017	00	00346		To Call again routine
310	0		157	00	00060		Send QRD
311	0		017	00	00353		To Query again routine
312	0		157	00	00014		Send RCV
313	0		017	00	02100		Error exit

The command in location 304 sends the Call to the line printer. The command in location 306 queries whether the index pulse was sent from the line printer. The command in location 310 queries whether the printer is in the ready condition. The command in location 312 sends the RCV line instruction. Location 307 contains an exit to a Call again routine. Location 311 contains an exit to a Query Ready again routine. Locations 305 and 313 contain unusual error exits.

314	0		033	05	01400		Block transmit data
315	0		100	10	00003		Characters
316	0		017	00	02100		Error exit

The block transmit data command in locations 314 and 315 causes the Central Processor to start the data transmission and then send to the line printer the data for one character from the print image. The data for this row is taken from locations 1400, 1401 and 1402. The next time this command is executed, the data for the second character will be taken from locations 1403, 1404 and 1405, and so on. Location 316 contains the unusual error exit.

317	0		002	05	00003		Add 3 to (05)
320	0		405	00	00005		CLA (05)
321	0		121	00	00275		If less than 275
322	0		017	00	00343		Transfer

The index used to keep a count of the characters sent while printing one line is incremented by three and tested less than 275 with the commands in locations 317 and 321. If the count is greater than 275, control is transferred to location 343.

323	0		157	00	00302	Send CAL
324	0		017	00	02100	Error exit
325	0		157	00	00061	Send QER
326	0		017	00	02100	Error exit
327	0		157	00	00001	Send CLQ
330	0		017	00	02100	Error exit
331	0		157	00	00055	Send STT to upspace
332	0		017	00	02100	Error exit
333	0		157	00	00054	Send CFD, upspace second line
334	0		017	00	02100	Error exit
335	0		017	00	00335	Wait interrupt

The command in location 323 sends the Call to the line printer. The command in location 325 sends the Query Error instruction. The command in location 327 sends the Clear Query instruction. The command in location 331 starts the paper upspace for one line. The command in location 333 continues the upspace for one more line. Locations 324, 326, 330, 332 and 334 contain unusual error exits. The command in location 335 is a wait for interrupt command (See Chapter 1, Section 4).

It is assumed that the exit from the ISR has been plugged so that when the interrupt occurs after the printer has upspaced the second line, the ISR will be entered and will transfer to location 336.

336	0		157	00	00302	Send CAL
337	0		017	00	02100	Error exit
340	0		157	00	00001	Send CLQ
341	0		017	00	02100	Error exit
342	0		017	00	00201	To print image builder

The command in location 336 sends the Call to the line printer. The command in location 340 sends the clear query instruction. The command in location 342 transfers back into the print image builder routine.

343	0		157	00	00302	Send CAL
344	0		017	00	02100	Error exit
345	0		017	00	00310	Transfer to 310

The command in location 343 is used to place the line printer in the CALLED state before the command in location 345 transfers control to location 310.

346	0		157	00	00001		Send CLQ
347	0		017	00	02100		Error exit
350	0		157	00	00065		Send QIP
351	0		017	00	00350		Return to QIP
352	0		017	00	00310		Transfer to 310

On a transfer from location 307, these commands clear the indicators and query the Index Pulse until the print wheel again comes into position to begin printing. Then control goes to location 310.

353	0		012	06	00004		4 → (06)
354	0		157	00	00060		Send QRD
355	0		017	00	00357		Transfer to 357
356	0		017	00	00312		Transfer to 312
357	0		026	06	00001		(06) - 1 = 0?
360	0		017	00	00354		Transfer
363			017	00	02100		Error exit

On a transfer from location 311, these commands send the Query Ready instruction to the line printer five times and, if there is then no ready condition, control is transferred to some error program. If a ready condition is found, control is returned to location 312.

14. INITIAL LOAD LINE SIGNAL SEQUENCE

See page 84 for the procedure of operation.

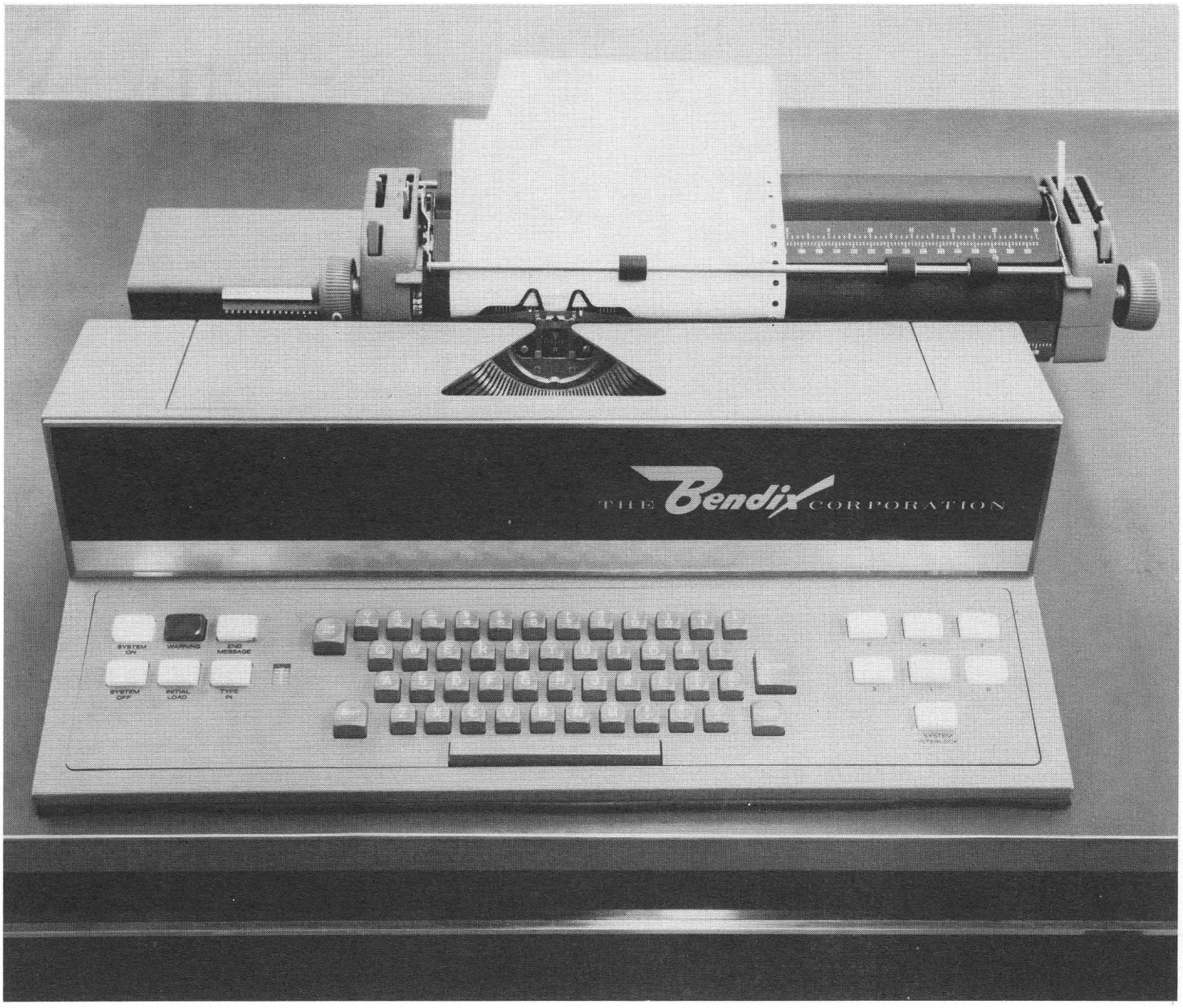
The line signals are explained below:

Central Processor	PC-10	
	ddd	The first data character is sent by the card reader to the Central Processor.
REQ	ddd	The request signal is sent by the Central Processor to the PC-10 to request the next data character. The card reader answers with a data character.

The data transmission continues through character number 119. Then:

REQ	ddd	A request signal is sent by the Central Processor to the PC-10 to request the next data character. The card reader answers with data character number 120.
REQ	END	A request for the next data character is sent. The PC-10 sends an END character; then it enters the STANDBY state.

The Central Processor then transfers to location 65, considers the contents to be a command, and executes the command.



CC-10 CONTROL CONSOLE

CHAPTER FOUR
THE CC-10 CONTROL CONSOLE

1. The Control Console, CC-10, is intended primarily as a controlling and monitoring device, although it can also be used for the exchange of short messages between the operator and the Central Processor. The CC-10 consists of an input/output electric typewriter, several control switches, six program-controlled lights, and an extension tone speaker. The control console provides for the following functions:

Operator control of the G-20 system

Input of messages (including a printed record of all input)

Output of messages from the Central Processor

Display of visual information on the program lights

Production of audible signals via the extension tone speaker for warning and monitoring purposes.

The CC-10 typewriter operates in the single character mode only by interrupting the program in the Central Processor briefly for the transmission or reception of each character. Figure 1 shows the typewriter keyboard. Eighty-eight characters including upper and lower case letters, figures, and special symbols (see Table 1) are available.

The switches and indicators of the CC-10 are, in general, self-explanatory and do not incorporate any unusual or unfamiliar features. The six program lights, which can be selectively turned on by sending a command numeric to the console, are used as program status indicators. Each of the last six bits of the command numeric corresponds to one of the lights. The lights remain turned on until another command numeric is sent to specify a new configuration.

A few special rules must be observed in operating the typewriter of the CC-10 since the typewriter can be operated by both the operator and the computer. The keyboard should always remain locked except during operator type-in. While the keyboard is locked, the computer has control of the typewriter and none of the keys can be depressed by the operator. The operator requests control by operating the TYPE-IN switch which transmits an interrupt request to the Central Processor. The keyboard may now be unlocked (under program control) and the operator can begin typing in his message. To conclude the message, the operator depresses the END MESSAGE switch. This automatically locks the keyboard. Thus, the keyboard can be unlocked only by the computer, and locked only by the operator. The control program for input should be so written that the computer will not attempt to use the typewriter while the keyboard is unlocked.

Only the Central Processor has control of the format used during type-out. The Central Processor can transmit the data numerics corresponding to Set Tab, Clear Tab and Backspace. Except for the keyboard lock and the above-mentioned characters, all typewriter operations can be controlled by both the operator and the computer. The margin on the CC-10 typewriter cannot be manually adjusted. All left margin adjustments are made by tabulating the appropriate amount from the left limit stop. When the right limit stop is reached during type-in, the operator must depress the carriage return key to return the carriage to the left margin limit. Depressing the carriage return key transmits an interrupt request to the Central Processor and places the carriage return code in the CC-10 holding register. Before each character to be typed is transmitted to the CC-10 during type-out, the Central Processor (under program control) must transmit a query to determine if a right margin limit interlock has occurred. There is no automatic interlock indication sent to the Central Processor during type-out.

Depressing any typewriter key (whether by operator or computer):

Records the character typed on paper or performs the designated mechanical function.

Places in the CC-10 holding register the numeric code corresponding to the key depressed, and

Transmits to the Central Processor an interrupt request signifying completion of a typing stroke.

The contents of the CC-10 holding register should be interrogated by the Central Processor within about 20 milliseconds after reception of the interrupt request,

since 20 milliseconds is the minimum time required for a typist to type out two characters on a typewriter. Delaying longer might result in loss of the former character in the holding register due to the typing of the second character. The information will remain in the holding register indefinitely if not disturbed. The Central Processor may interrogate the CC-10 holding register during type-out and use this information to verify the type-out character transmission. Set Tab, Clear Tab, and Unlock Keyboard cannot be verified.

The Control Console can operate in the following states:

STANDBY

CALLED

BUSY-QUIET

BUSY-ALERT

OUT of Service

Complete programming samples for Type-out without Verification, Type-out with Verification and Type-in are shown on pages 85, 90 and 94, respectively.

TABLE 1
CONSOLE DATA CHARACTER CODES
(Except as noted otherwise, these characters can be both transmitted and received.)

Octal* Code	Character	Octal Code	Character	Octal Code	Character
400	(Space)	444	4	510	h
401	A	445	5	511	i
402	B	446	6	512	j
403	C	447	7	513	k
404	D	450	8	514	l
405	E	451	9	515	m
406	F	452	↑0	516	n
407	G	453	.	517	o
410	H	454	+	520	p
411	I	455	-	521	q
412	J	456	*	522	r
413	K	457	/	523	s
414	L	460	=	524	t
415	M	461	∨	525	u
416	N	462	not used	526	v
417	O	463	∧	527	w
420	P	464	<	530	x
421	Q	465	$	531	y
422	R	466	>	532	z
423	S	467	;	533 through 557	not used
424	T	470	(		
425	U	471	[		
426	V	472	]	560	Tab
427	W	473	)	561	Carriage return
430	X	474	↓	562	Set tab stop (receive only)
431	Y	475	↑	563	Clear tab stop (receive only)
432	Z	476	:	564	not used
433	|	477	'	565	not used
434	←	500	not used	566	Backspace (receive only)
435	→	501	a	567	Unlock keyboard (receive only)
436	—	502	b	570	End of message (transmit only)
437	'	503	c	571 through 777	not used
440	0	504	d		
441	1	505	e		
442	2	506	f		
443	3	507	g		

*See next page

If any character labeled "not used" is inadvertently transmitted to the Control Console, the octal character "500" will be placed in the holding register. The CC-10 will transmit GRN and an interrupt as usual in response to such an erroneous character.

* The most significant bit of the three octal digit code is present only on the communication line. The codes themselves are octal representations of eight binary digits. The code for A is noted as 401_8 but is written as 001_8 in machine language coding ($001_8 = 00000001_2$).

2. LINE INSTRUCTIONS

The line instructions which are used with the CC-10 are given below. The octal code, the alpha code and the name of each are given.

nnn CAL Call

The unit with the call number nnn is called with the CAL instruction. If the CC-10 is called while in the STANDBY or CALLED state, it enters the CALLED state. If the CC-10 is called while in the BUSY-QUIET or BUSY-ALERT state, it enters the BUSY-ALERT state. Any call to the CC-10 is answered with the character GRN. Upon entering the CALLED state, the CC-10 can obey all data character line instructions pertaining to the typewriter or program indicator lights. Upon entering the BUSY-ALERT state, the CC-10 can obey succeeding line instructions but will not accept data character instructions pertaining to the typewriter.

If another unit on the communication line is called while the CC-10 is in the CALLED or BUSY-ALERT state, the CC-10 will enter the STANDBY or BUSY-QUIET state, respectively, without replying.

002 GRN Green

A GRN character is the response from the CC-10 which the Central Processor receives to instruct it to continue in the expected sequence (i.e. skip one command word).

003 RED Red

A RED character is the response from the CC-10 which the Central Processor receives to instruct it to branch to an alternate action (i.e. take the next command in sequence).

016 TRA Transmit

The CC-10, upon receiving the TRA command, replies by transmitting to the Central Processor the data character currently contained in the Control Console Holding register. The data character is transmitted into the least significant nine positions of the Line Response register.

060 QRD Query Ready

The Central Processor transmits the QRD character to the CC-10 to determine whether the console is ready. If the typewriter operation cycle is completed, and during output from the Central Processor, the console is not interlocked, the CC-10 answers GRN. If the cycle is not completed, if the right margin is interlocked, or if a key is stuck, the CC-10 answers RED.

062 QIL Query Interlock

The Central Processor transmits the QIL character to the CC-10 to determine whether the console is interlocked at the right margin while the G-20 is using the console. If so, the CC-10 answers RED; if not, the CC-10 answers GRN.

066 QN2 Query Interrupt Two

The CC-10 has an interrupt indicator which is turned on by depressing the TYPE-IN key on the keyboard. The Central Processor transmits the QN2 character to the CC-10 to determine if the CC-10 sent the type-in interrupt. If the indicator is on, the CC-10 answers RED; if not, it answers GRN. The QN2 character turns the indicator off.

067 QN1 Query Interrupt One

The CC-10 has an interrupt indicator which is turned on whenever the Control Console transmits an interrupt request during a transmission. The Central Processor transmits the QN1 character to a CC-10 to determine if the CC-10 sent an interrupt due to the typing in or typing out of a data character. If the indicator is on, the CC-10 answers RED; if not, it answers GRN. The QN1 character turns the indicator off.

lnn nn Set Indicator Lights

The Central Processor transmits the lnn to the CC-10 to selectively turn the program lights on or off. The octal nn, interpreted as a 6-bit binary number, should contain a "1" to turn the corresponding light on and a "0" to turn the corresponding light off. For example, let $nn = 12_8 = 001010_2$. The numeric 112 is transmitted and the resulting light configuration is:

off	off	on
off	on	off

The CC-10 answers GRN.

011 OUT Out

The Central Processor transmits the OUT character to cause the CC-10 to disconnect itself from the communication line and go OUT of Service. The CC-10 replies GRN.

3. OPERATING SWITCHES

TYPE-IN

The Type-in signal is transmitted when the TYPE-IN switch is depressed. The TYPE-IN switch requests control of the console typewriter.

END MESSAGE

The End Message signal is transmitted when the END MESSAGE button is depressed. Depressing the END MESSAGE button locks the keyboard, sends an interrupt request and places the numeric character 570 in the CC-10 holding register.

INITIAL LOAD Indicator

The INITIAL LOAD Indicator on the CC-10 Control Console is lit while the Central Processor is initially loading.

SYSTEM ON

When this button is depressed the entire system is turned on.

SYSTEM OFF

When this button is depressed simultaneously with SYSTEM INTERLOCK, the system is turned off. This procedure prevents accidental power turn-off.

SYSTEM INTERLOCK

The SYSTEM INTERLOCK button must be depressed simultaneously with SYSTEM OFF to turn off the system. Operation of the SYSTEM INTERLOCK button, independently, clears the CC-10 (sets all functions back to zero).

4. TYPE-OUT WITHOUT VERIFICATION SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the command and instruction characters are presented as they occur in memory. The octal code for the Control Console is to be 250.

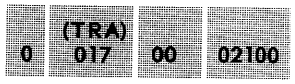
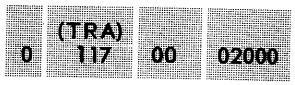
The starting address for the program is 200.

A hypothetical error routine is assumed to start in location 2100.

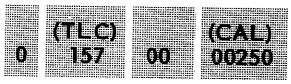
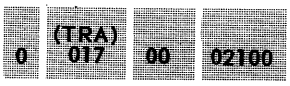
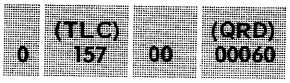
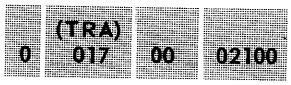
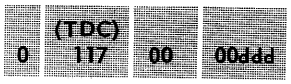
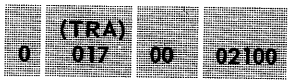
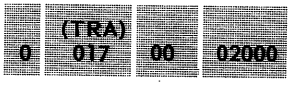
Another portion of the program in the Central Processor is assumed always to be in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and RED characters and each of the interrupt request signals. None of these signals need be programmed. The instruction or signal under the heading "Central Processor" comes earlier in time than the entry on the same line under the heading "CC-10."

		Central Processor	CC-10	
200	0 (TLC) 157 00 (CAL) 00250	CAL	GRN	The TLC command transmits the "call" to the Control Console with the call number 250. The Control Console answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
202	0 (TLC) 157 00 (QRD) 00060	QRD	GRN	The TLC command transmits the Query Ready (QRD) instruction to the Control Console. The Control Console is queried to see if the carriage is interlocked at the right margin. The answer is GRN.
203	0 (TRA) 017 00 02100			If the reply to the query is RED, control is returned to location 2100. The routine beginning at 2100 transmits the carriage return and returns control to location 204.
204	0 (TDC) 117 00 00ddd	ddd	GRN	The TDC command transmits the first data character to be typed out to the Control Console. The Control Console answers GRN. The ddd represents an eight-bit octal data character.

205			This command is the error exit for a non-GRN response.
206			This command transfers the Central Processor from the type-out routine to another portion of the program while the Control Console is completing the typing cycle.

(A) When the typing cycle is completed, an interrupt request is transmitted to the Central Processor on Line Number One. The interrupt will be completely processed and the Central Processor will return (under program control) to the type-out program.

207		CAL	GRN	The TLC command transmits the Call to the Control Console. The Control Console answers GRN.
208				This is the error exit for a non-GRN response.
209		QRD	GRN	The TLC command transmits the Query Ready (QRD) instruction to the Control Console. The Control Console is queried to see if the carriage is interlocked at the right margin. The answer is GRN.
210				This command is the error exit for a non-GRN response.
211		ddd	GRN	The TDC command transmits the next data character to be typed out to the Control Console. The Control Console answers GRN.
212				This command is the error exit for a non-GRN response.
213				

5. TYPE-OUT WITHOUT VERIFICATION EXAMPLE PROGRAM

The following example program will type the characters JFG in the leftmost three places on the CC-10 typewriter. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

200	0	012	01	00000		0 to index 01
201	0	157	00	00250		Send CAL
202	0	017	00	02100		Error exit, irregular error
203	0	157	00	00067		Send QN1, turns indicator off
204	0	017	00	00201		Exit for interrupt query
205	0	117	00	00161		Send carriage return character
206	0	017	00	02100		Error exit, irregular error
207	0	017	00	00207		Wait for interrupt
210	0	157	00	00250		Send CAL
211	0	017	00	02100		Error exit, irregular error
202	0	157	00	00067		Send QN1, turns indicator off
213	0	017	00	00215		Transfer
214	0	017	00	03000		Transfer out of program, error
215	0	157	00	00060		Send QRD
216	0	017	00	00205		Transfer to carriage return
217	0	517	01	00230		Send character to be typed
220	0	017	00	02100		Error exit
221	0	017	00	00221		Wait for interrupts
222	0	405	00	00001		CLA (00001)
223	0	161	00	00002		Finished with third letter?
224	0	017	00	00226		No, transfer
225	0	017	00	02000		Yes, out to another program
226	0	002	01	00001		Add 1 to index 01
227	0	017	00	00210		Return for another typeout
230	0	000	00	00012		Character J
231	0	000	00	00006		Character F
232	0	000	00	00007		Character G

Explanation of the Type-out Program

200	0		012	01	00000		0 to index 01
201	0		157	00	00250		Send CAL
202	0		017	00	02100		Error exit, irregular error
203	0		157	00	00067		Send QN1, turns indicator off
204	0		017	00	00201		Exit for interrupt query
205	0		117	00	00161		Send carriage return character
206	0		017	00	02100		Error exit, irregular error
207	0		017	00	00207		Wait for interrupt

The commands in locations 201 through 207 are used to position the carriage before beginning the type out. The command in location 201 transmits the call to the CC-10. The command in location 203 transmits a Query Interrupt line command which checks whether or not an interrupt has been sent from the CC-10, and also turns the number one interrupt indicator off to allow future interrupt requests to be generated. If the command in location 204 is executed, the program will return to the beginning of the routine. The carriage return character is sent by the command in location 205. Locations 202 and 206 contain transfers to the usual all-purpose error routine. Since no other computation is to be done in this example, location 207 contains a transfer to itself to wait for the interrupt request which follows the return of the carriage (See Chapter 1, Section 4).

The ISR will be entered when the interrupt occurs. The ISR will determine that the CC-10 sent the interrupt request and will transfer to location 210 according to a pre-arranged exit in the routine.

210	0		157	00	00250		Send CAL
211	0		017	00	02100		Error exit, irregular error
202	0		157	00	00067		Send QN1, turns indicator off
213	0		017	00	00215		Transfer
214	0		017	00	03000		Transfer out of program, error
215	0		157	00	00060		Send QRD
216	0		017	00	00205		Transfer to carriage return

The commands in locations 210 through 216 are used to set the CC-10 to receive the character to be typed. The command in location 210 transmits the call to the CC-10. The command in location 212 sends the QN1 line instruction to query whether an interrupt has been sent from the CC-10. The QN1 instruction also turns the line one interrupt indicator off. Since an interrupt had been sent, the command in the red exit location 213, which is a transfer to location 215, is executed. The command in location 215 sends the QRD line instruction to query whether the carriage is interlocked at the right margin. If so, the command in location 216 is executed. If some unexpected error has occurred and the interrupt indicator is off, the command in location 214 transfers out of the type-out program to some other routine.

217	0		517	01	00230		Send character to be typed
220	0		017	00	02100		Error exit
221	0		017	00	00221		Wait for interrupts

Location 217 contains the command used to send the character to be typed to the typewriter. The index 01 indicated in location 217 is used to pick up the three different letters to be typed. The codes for these letters are in locations 230, 231 and 232. The command in location 221 is used to wait for the interrupt request which will signify the completion of the typing stroke (See Chapter I, Section 4).

The ISR will be entered when the interrupt occurs. The ISR will determine that the CC-10 sent the interrupt and will transfer to location 222.

222	0		405	00	00001		CLA (00001)
223	0		161	00	00002		Finished with third letter?
224	0		017	00	00226		No, transfer
225	0		017	00	02000		Yes, out to another program
226	0		002	01	00001		Add 1 to index 01
227	0		017	00	00210		Return for another typeout

Locations 222 and 223 contain a test for whether the last type-out has occurred. The command in location 224 transfers control to location 226. The transfer command in location 225 is the exit from the type-out routine. Location 227 contains the return to type another character.

The Central Processor goes to another portion of its program (under program control) by the command in location 225. When the letter corresponding to the data character has been typed, the Control Console transmits an interrupt request on interrupt line number one.

The Central Processor should transmit CAL and QN1 to turn the interrupt indicator off, then return to another portion of its program. The Control Console will remain in the CALLED state until some other unit is called. Then the CC-10 will return to STANDBY.

6. TYPE-OUT WITH VERIFICATION SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are presented as they occur in memory. The octal code for the control console is to be 250.

The starting address for the program is 200.

An error routine is assumed to start in location 2100.

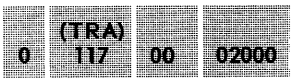
Another portion of the program in the Central Processor is assumed always to be in location 2000.

The sequence of communication line signals is given below in conjunction with the program commands. The communication system automatically transmits each of the GRN and RED characters and each of the interrupt request signals. None of these signals need be programmed. The instruction or signal under the heading, "Central Processor" comes earlier in time than the entry on the same line under the heading "CC-10."

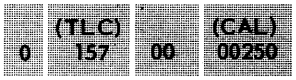
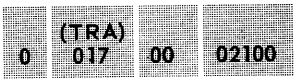
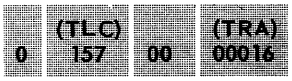
		Central Processor	CC-10	
200	0 (TLC) 157 00 (CAL) 00250	CAL	GRN	The TLC command transmits the "Call" to the Control Console with call number 250. The Control Console answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
202	0 (TLC) 157 00 (QRD) 00060	QRD	GRN	The TLC command transmits the Query Ready (QRD) instruction to the Control Console. The Control Console is queried to see if the carriage is interlocked at the right margin. The answer is GRN.
203	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.

(If the carriage position is known, the above QRD need not be entered.)

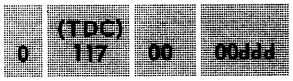
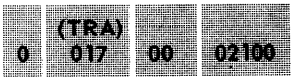
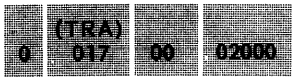
204	0 (TDC) 117 00 00ddd	ddd	GRN	The TLC command transmits the first data character to be typed out to the Control Console. The Control Console answers GRN.
205	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.

206				This command transfers the Central Processor away from the type-out routine to another portion of the program while the Control Console is completing the typing cycle.

(A) When the typing cycle is completed, an interrupt request is transmitted to the Central Processor on line number one. The interrupt will be processed and the Central Processor will return (under program control) to the type-out program.

207		CAL	GRN	The TLC command transmits the "Call" to the Control Console with call number 250. The Control Console answers GRN and enters the CALLED state.
208				This command is the error exit for a non-GRN response.
209		TRA	ddd	The TLC command transmits the transmit data character (TRA) instruction. The data character currently in the holding register is transmitted into the least significant nine bit positions of the Line Response register. The verification can then be performed (under program control).

After the verification the Central Processor returns to the type-out program.

210		ddd	GRN	The TDC command transmits the next data character to be typed out to the Control Console. The Control Console answers GRN.
211				This command is the error exit for a non-GRN response.
212				The Central Processor then executes another portion of its program by this command.

When the letter corresponding to the data character has been typed, the Control Console transmits an interrupt request on Interrupt Line Number One.

7. TYPE-OUT WITH VERIFICATION EXAMPLE PROGRAM

The following example program will type the characters JFG in the leftmost three places on the CC-10 typewriter. The type-out will be verified as each letter is typed. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
200	0		012	01	00000		0 into index 01
201	0		157	00	00250		Send CAL
202	0		017	00	02100		Error exit, irregular error
203	0		157	00	00067		Send QN1, turns off indicator
204	0		017	00	00201		Exit for interrupt query
205	0		117	00	00161		Send carriage return character
206	0		017	00	02100		Error exit
207	0		017	00	00207		Wait for interrupt from typed character
210	0		157	00	00250		Send CAL
211	0		017	00	02100		Error exit
212	0		157	00	00067		Send QN1, turns indicator off
213	0		017	00	00215		Transfer
214	0		017	00	03000		Transfer out of program, error
215	0		157	00	00060		Send QRD
216	0		017	00	00205		Transfer to carriage return
217	0		557	01	00241		Send character to be typed
220	0		017	00	02100		Error exit
221	0		017	00	00221		Wait for interrupt
222	0		157	00	00250		Send CAL
223	0		017	00	02100		Error exit
224	0		157	00	00016		Send TRA
225	0		017	00	00227		Transfer
226	0		017	00	02100		Error exit
227	0		072	02	00377		Character from CC-10 into (ACC)
230	0		531	01	00241		Verify typed character

LOCATION	F (31 30)	M (29 28)	OP (27 21)	I (20 15)	A Field (14 0)	u	Comments
231	0		017	00	00233		Transfer if verified
232	0		017	00	02100		CC-10 type error exit
233	0		405	00	00001		CL A (00001)
234	0		161	00	00002		Finished with third letter?
235	0		017	00	00237		Transfer
236	0		017	00	02000		Yes, out to another program
237	0		002	01	00001		Add 1 to index 01
240	0		017	00	00210		Return to another type-out
241	0		000	00	00012		Character J
242	0		000	00	00006		Character F
243	0		000	00	00007		Character G

The type-out with verification example is very similar to the type-out without verification from location 200 to location 221. The ISR should transfer to location 222 after the interrupt at location 221.

222	0		157	00	00250		Send CAL
223	0		017	00	02100		Error exit
224	0		157	00	00016		Send TRA
225	0		017	00	00227		Transfer
226	0		017	00	02100		Error exit
227	0		072	02	00377		Character from CC-10 into (ACC)
230	0		531	01	00241		Verify typed character
231	0		017	00	00233		Transfer if verified
232	0		017	00	02100		CC-10 type error exit

The commands in locations 222 through 232 are used to verify each character after it is typed. The command in location 222 calls the CC-10. The command in location 224 sends the TRA line instruction to direct the CC-10 to transmit to the Line Response register the last character typed. The command in location 227 picks up the character and places it in the accumulator where the command in location 228 verifies that it is the correct character. If the charactor has been correctly typed, the command in location 231 is executed and control is transferred to location 233. Otherwise, a type-out error routine is entered (this error routine is not given in this example).

The command sequence in locations 233 through 237 is identical with locations 222 through 227 of the type-out without verification program example.

The type cycle of the third character will terminate with the interrupt request on interrupt line one. The Central Processor should transmit CAL and QN1 to turn the interrupt indicator off, then return to another portion of its program. The Control Console will remain in the CALLED state until some other unit is called. Then the CC-10 will return to STAND-BY.

8. TYPE-IN SAMPLE

The following sample indicates the manner in which a program could be written to solve the given problem. An actual program is given in its entirety in the following section.

The program example considered will allow type-in of characters on the CC-10.

The following conditions are assumed to be true for this sample:

All command flags are 0. All addressing modes are 0. The contents of the Operand Assembly register and all Index fields are zero.

The octal codes for the commands and instruction characters are presented as they occur in memory. The octal code for the Control Console is to be 250.

The starting address for the program is 200.

An error routine is assumed to start in location 2100.

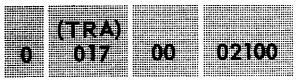
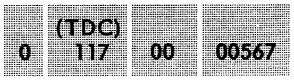
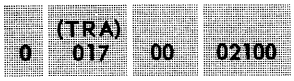
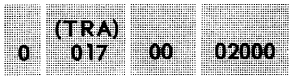
Another portion of the program in the Central Processor is assumed always to be in location 2000.

The sequence of communication line signals is given in conjunction with the program commands. The communication system automatically transmits each of the GRN and RED characters and each of the interrupt request signals. None of these signals need be programmed. An assignment (as used below) of the interrupt lines connected to the CC-10 has been assumed. The instruction or signal under the heading "Central Processor" comes earlier in time than the entry on the same line under the heading "CC-10."

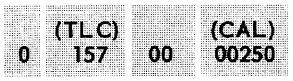
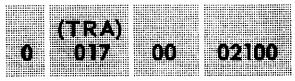
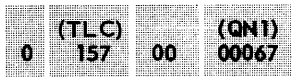
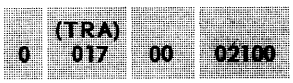
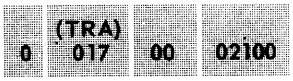
An interrupt request, initiated by the depressing of the BEGIN MESSAGE switch by the operator, is transmitted to the Central Processor on interrupt line number two. The interrupt service routine transfers control to location 200.*

		Central Processor	CC-10	
200	0 (TLC) 157 00 (CAL) 00250	CAL	GRN	The TLC command transmits the "Call" to the control console with call number 250. The Control Console answers GRN and enters the CALLED state.
201	0 (TRA) 017 00 02100			This command is the error exit for a non-GRN response.
202	0 (TLC) 157 00 (QN2) 00066	QN2	RED	The TLC command transmits the Query Interrupt (QN2) instruction to the Control Console. The answer is RED. The interrupt indicator in the CC-10 is reset.
203	0 (TRA) 017 00 00205			This command is the exit for a non-GRN response. (A RED response from the command in location 202 causes the command in location 203 to be executed.) A RED response to QN2 is expected. A GRN response is considered an error in this case.

*In example 8, it is assumed that the Interrupt Service Routine transfers to the type-in routine as soon as it is discovered that line one or line two (the CC-10 interrupt lines in this example) has sent an interrupt. Otherwise the queries for interrupt in the example would always be answered with GRN.

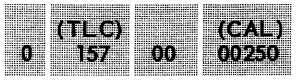
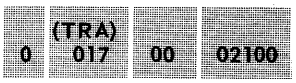
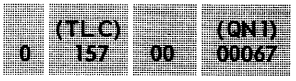
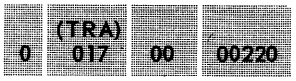
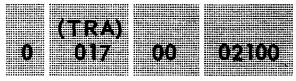
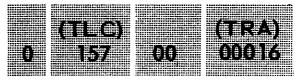
204				This command is an error exit for a non-RED response.
205		567	GRN	The TDC command transmits the Unlock Keyboard data character 567 to the CC-10. The CC-10 answers GRN.
206				This command is the error exit for a non-GRN response.
207				This TRA command transfers the Central Processor to another portion of the program while the keyboard is being physically unlocked.

The keyboard of the Control Console is unlocked. The Control Console then transmits an interrupt request on interrupt line number one. The interrupt service routine transfers control to location 208. (To tell the operator that the keyboard is unlocked, the TYPE-IN light is lit.)

208		CAL	GRN	The TLC Command transmits the "Call" to the control console with call number 250. The Control Console answers GRN and enters the CALLED state.
209				This command is the error exit for a non-GRN response.
210		QN1	RED	The TLC command transmits the Query Interrupt (QN1) instruction to the Control Console. The answer is RED. The interrupt indicator in the CC-10 is reset.
211				This command is the exit for a non-GRN response. (A RED response from the command in location 210 causes the command in location 211 to be executed.) A RED response to QN1 is expected.
212				This is an error exit for a non-RED response.

The above five commands are needed to turn the interrupt indicator off in the CC-10 to allow the interrupt to be sent for the type-in of the next character.

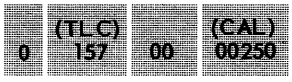
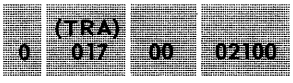
(A) The operator then can type a character. The octal code for the character is placed in the holding register of the CC-10 and an interrupt request is transmitted on interrupt line number one. The Interrupt Service Routine transfers control to location 215.

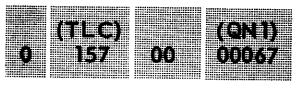
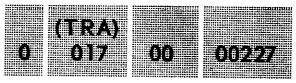
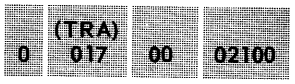
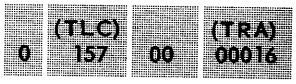
215		CAL	GRN	The TLC command transmits the "Call" to the Control Console with call number 250. The Control Console answers GRN and enters the CALLED state.
216				This command is the error exit for a non-GRN response.
217		QN1	RED	The TLC command transmits the Query Interrupt instruction to the Control Console. The answer is RED. The interrupt indicator in the CC-10 is reset.
218				This command is the exit for a non-GRN response. (A RED response from the command in location 216 causes the command in location 217 to be executed.) A RED response to QN1 is expected.
219				This command is an error exit for a non-RED response.
220		TRA	ddd	The TLC command transmits the Transmit Data Character (TRA) instruction. The data character currently in the holding register is transmitted into the least significant nine bit positions of the Line Response register.

The input character is in the Line Response register and can be used as the programmer desires.

Return to point (A) until the end of the type-in.

Then an interrupt request, initiated by depressing the END MESSAGE switch by the operator, is transmitted to the Central Processor on interrupt line number one. Depressing the END MESSAGE switch locks the keyboard. The Interrupt Service Routine transfers control to location 222.

222		CAL	GRN	The TLC command transmits the "Call" to the Control Console with call number 250. The Control Console answers GRN and enters the CALLED state.
223				This command is the error exit for a non-GRN response.

224		QN1	RED	The TLC command transmits the Query Interrupt instruction to the Control Console. The answer is RED. The Interrupt Indicator in the CC-10 is reset.
225				This command is the exit for a non-GRN response.
226				This is an error exit for a non-RED response.
227		TRA	570	The TLC command transmits the Transmit Data Character (TRA) instruction. The character transmitted is 570, the End of Message code.

After processing the 570 character, the Central Processor (under program control) returns to another portion of the program.

9. TYPE-IN EXAMPLE PROGRAM

The following example program will allow the type-in of any number of characters from the typewriter on the CC-10. The characters typed in are placed in memory beginning with location 1000. The program is assumed to start in location 200. The minimum ISR is assumed to be in memory for use with this example.

To initiate the transfer of control to this program, the TYPE-IN button on the console must be depressed. This causes an interrupt request and the ISR is entered. The ISR then transfers control to location 200.

LOCATION	F (31)	M (30-29)	OP (28-27 to 21)	I (20 to 15)	A Field (14 to 0)	u	Comments
200	0		157	00	00250		Send CAL
201	0		017	00	02100		Error exit
202	0		157	00	00066		Send Query interrupt number 2
203	0		017	00	00205		Transfer
204	0		017	00	02100		Error exit
205	0		117	00	00167		Send keyboard unlock
206	0		017	00	02100		Error exit
207	0		017	00	00207		Wait for interrupt
210	0		157	00	00250		Send CAL
211	0		017	00	02100		Error exit
212	0		157	00	00067		Send Query interrupt number 1
213	0		017	00	00213		Wait for type-in
214	0		017	00	02100		Error exit
215	0		157	00	00250		Send CAL
216	0		017	00	02100		Error exit
217	0		157	00	00067		Send Query interrupt number 1
220	0		017	00	00222		Transfer
221	0		017	00	02100		Error exit
222	0		157	00	00016		Send TRA
223	0		017	00	00225		Transfer
224	0		017	00	02100		Error exit
225	0		072	02	00377		Pickup (line response)
226	0		131	00	00170		= End message
227	0		017	00	02000		Yes, out to new program
230	0		173	01	01000		No, place character in location
231	0		002	01	00001		Add 1 to index 01
232	0		017	00	00213		Transfer to wait for next type-in

Explanation of the Type-in Example Program.

200	0		157	00	00250	Send CAL
201	0		017	00	02100	Error exit
202	0		157	00	00066	Send Query interrupt number 2
203	0		017	00	00205	Transfer
204	0		017	00	02100	Error exit
205	0		117	00	00167	Send keyboard unlock
206	0		017	00	02100	Error exit
207	0		017	00	00207	Wait for interrupt

The commands in locations 200 through 207 are used to determine whether the request type-in interrupt has occurred and to unlock the keyboard so that the operator can type in a character. The command in location 200 sends the call to the CC-10. The command in location 202 sends the interrupt query for the number two interrupt. If the answer to the query indicates that there was no interrupt, the command in location 204 transfers control to the error routine.

The command in location 205 transmits the unlock-keyboard line character to the CC-10. The command in location 207 is used to wait for the interrupt request that will be sent after the keyboard is unlocked.

The ISR will be entered when the interrupt occurs. The ISR will determine that the CC-10 sent the interrupt request and will transfer to location 210.

210	0		157	00	00250	Send CAL
211	0		017	00	02100	Error exit
212	0		157	00	00067	Send Query interrupt number 1
213	0		017	00	00213	Wait for type-in
214	0		017	00	02100	Error exit

The sequence of commands in locations 210 through 214 used only to reset the interrupt indicator on the CC-10 so that the interrupt request can come from the CC-10 when the operator types in a character. The command in location 213 is used to wait for the type-in interrupt request.

215	0		157	00	00250		Send CAL
216	0		017	00	02100		Error exit
217	0		157	00	00067		Send Query interrupt number 1
220	0		017	00	00222		Transfer
221	0		017	00	02100		Error exit
222	0		157	00	00016		Send TRA
223	0		017	00	00225		Transfer
224	0		017	00	02100		Error exit
225	0		072	02	00377		Pickup (line response)
226	0		131	00	00170		= End message
227	0		017	00	02000		Yes, out to new program
230	0		173	01	01000		No, place character in location
227	0		017	00	02000		Yes, out to new program
230	0		173	01	01000		No, place character in location
231	0		002	01	00001		Add 1 to index 01
232	0		017	00	00213		Transfer to wait for next type-in

The sequence of commands in locations 215 through 232 picks up the typed character, checks for the end of type-in and stores the typed character in memory. The command in location 215 sends the call to the CC-10, the command in location 217 sends the query for interrupt and the command in location 222 sends the TRA instruction, which causes the CC-10 to send the typed character to the Line Response register.

The command in location 225 picks up the typed in character from the Line Response register and places it in the accumulator. The command in location 226 checks to see if the character is the End Message character. If so, the command in location 227 transfers out of the type-in routine. If not, the command in location 230 stores the character in location 1000. The command in location 231 adds one to the contents of index 01 so that the next character will be stored in location 1001 and the transfer in location 232 returns to the wait-for-type-in command in location 213.

Bendix Computer Division

THE Bendix CORPORATION

December 1961

BET-12611