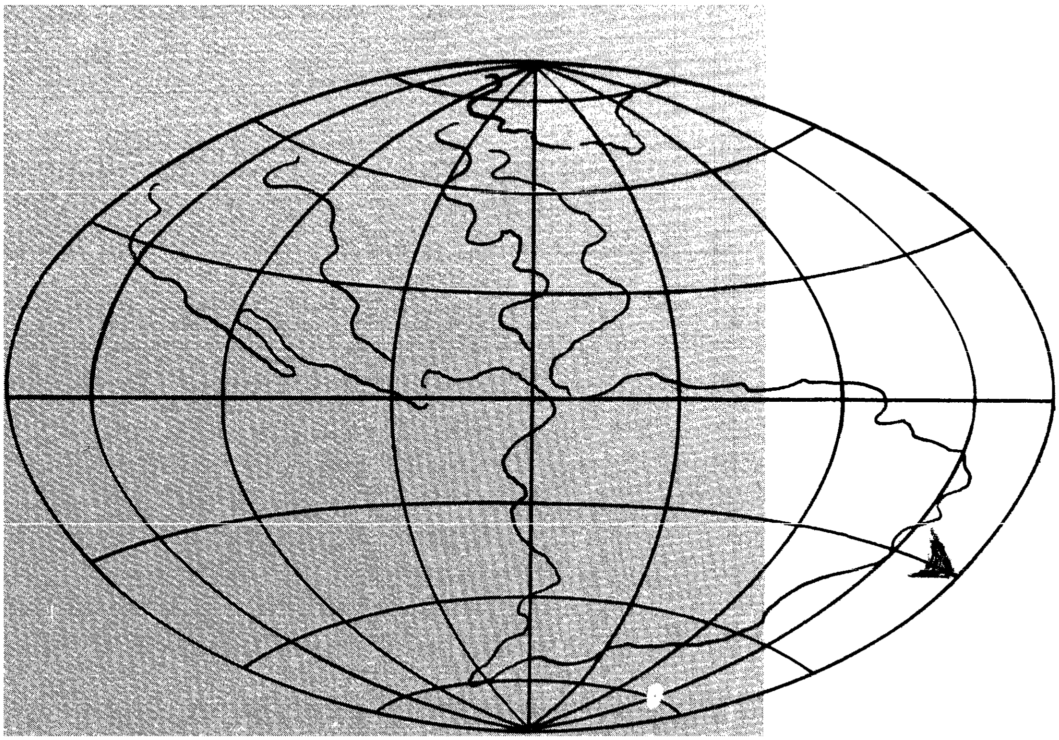


UNIVAC® 1105 SYSTEM

PRELIMINARY

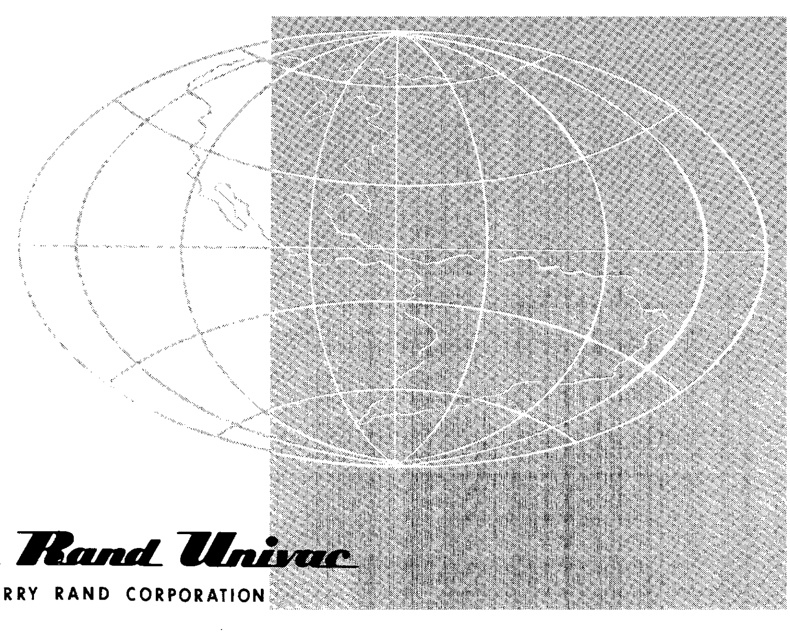
PROGRAMMING MANUAL

Remington Rand Univac
DIVISION OF SPERRY RAND CORPORATION

UNIVAC® 1105 SYSTEM

PRELIMINARY

Programming Manual



Remington Rand Univac

DIVISION OF SPERRY RAND CORPORATION

PREFACE

The UNIVAC 1105 system is a high speed computing system capable of handling both scientific and data processing applications. The 1105 provides all the outstanding features of the 1103A computing system with the addition of two buffer storage units and a second magnetic tape control unit. These additional features permit the central computer to communicate with a maximum of 20 UNISERVO II tape handlers in concurrent operations of compute-read-read, compute-write-write, or compute-read-write.

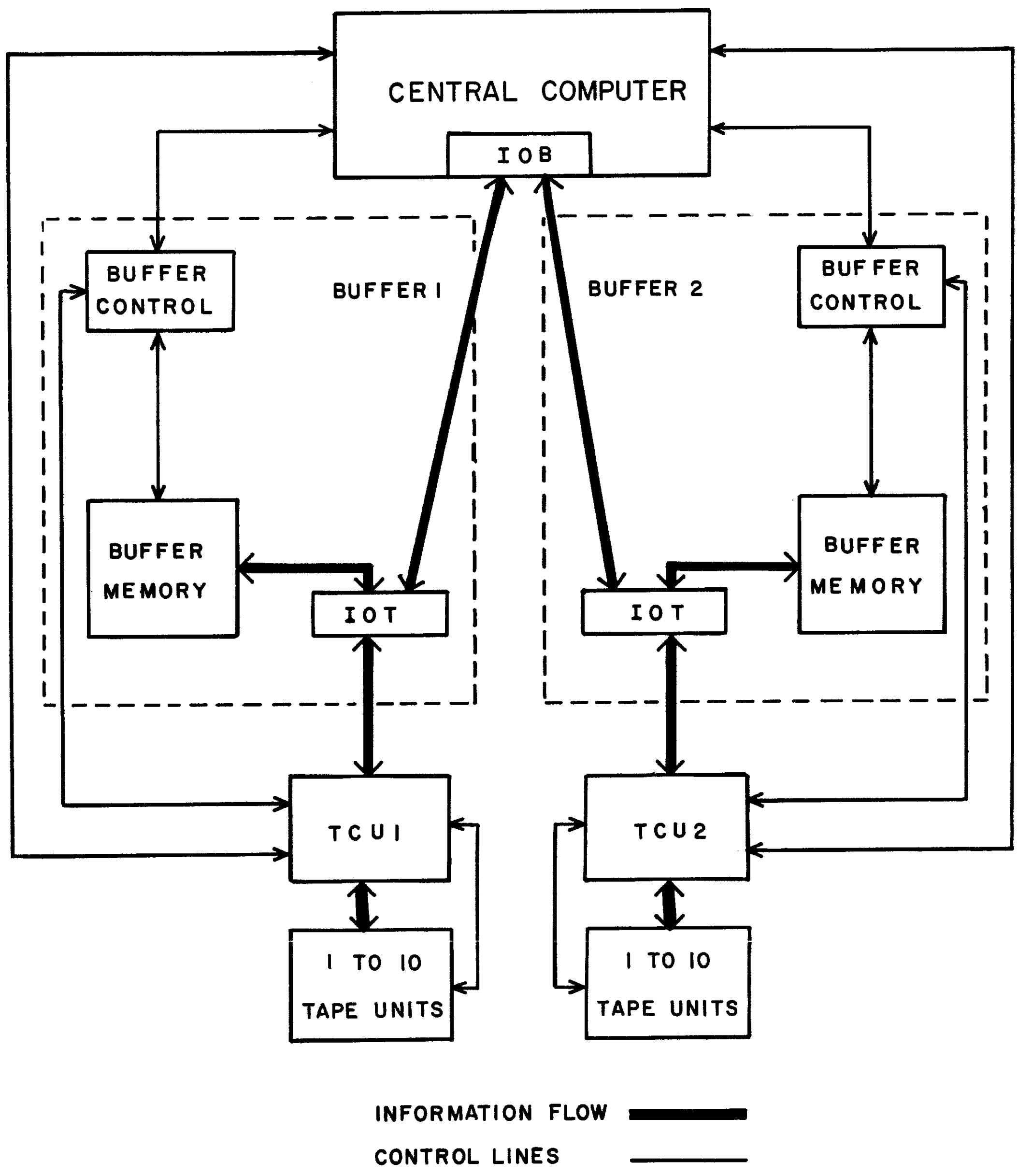
This manual assumes knowledge on the part of the reader of the basic 1103A and is devoted primarily to an explanation of buffer storage.

TABLE OF CONTENTS

DIAGRAMS

DATA TRANSFER

BUFFER — TAPE CHANNELS



GENERAL

The UNIVAC 1105 computing system provides two input-output channels for computer communication with magnetic tapes. As shown on the preceding page, the two channels are completely independent except for common use of IOB, a 36-bit input-output register within the central computer. Each of two buffer storage units has its own 36-bit Input-Output Transfer register, IOT, and an associated Tape Control Unit, TCU. Each Tape Control Unit controls up to 10 UNISERVO tape units, the tape units associated with buffer 1 being completely independent of buffer 2, and vice versa.

Communication between a buffer storage unit and its associated tape units imposes no restrictions upon simultaneous operation of the other buffer storage unit. Buffer 1 can be receiving data from magnetic tape, or transmitting data to the tape, while buffer 2 is similarly involved in either of these transfers. Moreover, during the transfer of data between either or both buffers and magnetic tape, the computer is completely free for computation. The same principle of concurrent operation applies to other tape operations such as move forward, move backward, rewind, etc.

Of the overall time consumed by the transmission of data between the computer and magnetic tape, only an approximate 5% of the fastest transfer time is used for computer instructions to direct the exchange of information. The remaining 95% is available for computation.

Either of two modes of tape operation is possible with the 1105: normal (buffer) mode or bypass mode. The normal mode of tape operation utilizes buffer storage whereas the bypass mode removes the buffers from the flow of information between the computer and magnetic tape. The bypass mode is thus likened to tape operation on the 1103A computer. Selection of the mode is accomplished by programming appropriate code bits in External Function instructions. Operation with one buffer unit in the normal mode can be simultaneous with operation bypassing the other buffer.

The normal mode of operation can be utilized for reading and writing blocks of both fixed and variable length, provided that the length of a variable block does not exceed 720 characters. To handle variable blocks of more than 720 characters, or information recorded in the continuous data input format, the bypass mode of operation is utilized.

RECORDING STANDARDS

UNITAPE metallic tape is used by the tape handling equipment in recording and reading information. Information is recorded as magnetized areas in eight channels across the width of the tape. Data bits are recorded in six of these channels; one channel contains the parity check bit; and one channel contains the sprocket, or timing, bit.

A row of eight binary digits across the width of a tape is termed a line, and six data bits are termed a hexabit character. In terms of a 36-bit computer word, six lines are required to record a word. A block (of fixed block length) consists of 720 consecutive lines or 120 computer words; a blockette consists of 120 consecutive lines or 20 computer words. A "dead space" (in which no information is recorded) of 1.2 or 2.4 inches exists between blocks of fixed length, and a dead space of 0.0, 0.1, or 1.2 inches exists between blockettes. Spacing between blocks of variable length is 1.2 inches. When tape movement is stopped, the stop occurs such that the read/write head is positioned over the dead space between blocks.

A UNISERVO tape handler is a unit of equipment comprising a read/write head, an erase head, a bad spot detector, and tape handling mechanism, such as the tape reel mount, reel drives, etc. In a reading operation, the read/write head detects "1's" on the forward or backward moving tape. During a writing operation the erase head is also activated so that the tape in its forward passage past the erase head has "0's" recorded on its entire width.

Since it is difficult to produce a flawless reel of magnetic tape, the manufacturer employs a tape checking device which seeks out the inherent bad spots, e.g., tape splices or areas which test below the required reading level. Identification is made of these areas by perforating the tape along the longitudinal center preceding, through, and following the bad spot. The bad spot detector on the tape unit automatically interrupts reading, writing, or moving operations until the undesirable tape area passes the read/write head. The length of a reel of magnetic tape, as specified by the manufacturer, denotes the feet of good recording surface regardless of the number of bad tape areas contained therein.

UNISERVO II tape handler operates with reels of ten, eight, or six inch diameter (2400 feet or less). Lines are recorded on tape at densities of either 128 lines per inch or a nominal 200 lines per inch. (Recording of variable length blocks is at the high density only.) With the high density recording of fixed length blocks, separated by 1.2 inch and with no separation of blockettes, a 1500 foot reel of tape can store up to approximately 450,000 computer words or 3750 blocks. A 2400 foot reel of tape ean store up to approximately 720,000 computer words or 6000 blocks. Thus, up to 14,400,000 36-bit words may be stored on 20 reels and processed by the UNIVAC 1105 system without changing tapes.

BUFFER STORAGE

Buffer memory is a transistorized magnetic core memory with a capacity of 120 36-bit words. Each buffer has a 36-bit input-output transfer register (IOT) through which the memory is loaded or unloaded, a word at a time. Buffer memory does not distinguish between receiving data from the computer and receiving data from the tape - it merely accepts information from IOT. Similarly, a buffer does not distinguish between transmitting data to the tape and transmitting data to the computer. Thus, the direction of data transfer must be established by External Function instructions.

LOAD STATE

A buffer is said to be in the "load state" if the memory is ready to accept data: this condition is a requirement for reception of data. Both buffers are set automatically to the load state by depressing the Master Clear button on the computer console. Either of the buffers is set to the load state by depressing its individual Clear button* on the auxiliary console. A previous operation which completely unloaded the buffer memory also automatically switches the buffer to the load state. If only a partial block (fixed or variable length) is unloaded by a buffer to the computer, the switch to the load state is not automatic. An External Function instruction must be programmed to clear the remaining words from the buffer memory and set the buffer to the load state.

UNLOAD STATE

A buffer is said to be in the "unload state" if the memory has previously received data, and is now ready to transfer it elsewhere. This condition is a requirement for transferring data from the buffer. If the buffer was completely filled by a previous load operation, the switch to the unload state is automatic. However, in the case where less than 120 words are transferred to a buffer from the computer, the switch to the unload state is not automatic: an External Function instruction must be programmed to set the buffer to the unload state. The switch to the unload state is automatic after reading a variable block, of 120 words or less, from a tape to a buffer.

BUFFER ACTIVITY

A buffer is defined as being "active" when the buffer memory is engaged in a data transfer between itself and magnetic tape. Hence, a buffer is active when

(1) buffer memory is receiving data from the tape being read, or
(2) buffer memory is transmitting information to the tape for writing.

Buffer activity is concluded when

(1) the buffer is switched from the load to the unload state, or
(2) the buffer is switched from the unload to the load state,

following completion of the respective data transfers listed above.

The completion of buffer activity is disclosed by the program interrupt, or it may be detected by the program, using an amplified Manually Selective Jump instruction.

While reading or writing tape in free-run operation, buffer activity is resumed for each successive block, as shown in the following sequences.

---

* To set a single, individual buffer to the load state, use the following procedure: set the Disconnect Master Clear switch for buffer 1 or buffer 2 to its abnormal position; depress the Clear button for buffer 1 or buffer 2; return the Disconnect Master Clear switch to its normal position; and, unless the computer is operating in the test mode, depress the Clear A Fault button. All mentions in this text of depressing a buffer Clear button imply following this procedure.

BUFFER ACTIVITY WHILE READING

(Free Run / Single Block)

EF Read Tape - read code to TCU----------> buffer becomes active
Tape acceleration
First block of data read from tape
Switch from load to unload state----------> completion of buffer activity
Buffer unloaded to computer
Switch from unload to load state--------------> buffer activity resumed UNLESS
1) Stop code included in EF Read Tape, or
2) EF Stop executed after EF Read Tape and prior to unload-to-load switching.

Second block read from tape
Switch from load to unload state--------> completion of buffer activity
Buffer unloaded to computer
Switch from unload to load state--------------> buffer activity resumed

. . .

Last desired block read from tape
Switch from load to unload state--------> completion of buffer activity
EF STOP → Buffer unloaded to computer
or → Switch from unload to load state

BUFFER ACTIVITY WHILE WRITING

(Free Run / Single Block)

Buffer loaded from computer
Switch from load to unload state*
EF Write Tape - write code to TCU---------> buffer becomes active
Tape acceleration
First block written on tape
Switch from unload to load state--------> completion of buffer activity
Transfer is completed if
1) Stop code included in EF Write Tape, or
2) EF Stop executed at this time or previously.

Buffer loaded from computer
Switch from load to unload state------------> buffer activity resumed
Second block written on tape
Switch from unload to load state---------> completion of buffer activity

. . .

Buffer loaded from computer
Switch from load to unload state-----------> buffer activity resumed
Last desired block written on tape
Switch from unload to load state--------> completion of buffer activity
EF Stop

NOTE: If the EF Write Tape is executed prior to the item starred, the buffer becomes active at this point, i.e., upon switching to the unload state.

PROGRAM INTERRUPT

The program interrupt feature is available during fixed and variable block length operation, normal mode use of the 1105. The interrupt is selected at the programmer's option for inclusion in the External Function instructions for reading, writing, or moving the tapes. The use and effect of the program interrupt with the buffer/tape system is equivalent to the interrupt available with the 1103A computing system. In brief, the interrupt feature arranges a program jump to the instruction at storage address 00002(F_3). A Return Jump instruction at this location would set aside for future reference the address currently held in the Program Address Counter, i.e., the address of the instruction which would have been executed had the interrupt signal not been received. The programmer should recall that in some cases, the interrupt signal does not effect an immediate interruption of the main program.

When reading magnetic tape, the program interrupt effects automatic interruption of the main program after a block of information has been read from the tape to the buffer. The interrupt signal is emitted upon completion of buffer activity, i.e., the interrupt signal occurs after the buffer has been switched from the load state to the unload state. The interrupt thus indicates that the buffer can now be unloaded to the computer.

When writing magnetic tape, the program interrupt effects automatic interruption of the main program after a block of information has been written on tape from the buffer. The interrupt signal again indicates completion of activity in as much as it occurs after the buffer has been switched from the unload state to the load state.

When moving magnetic tape, exclusive of a read or write operation, the program interrupt effects automatic interruption of the main program after the tape has been moved the specified number of blocks.

The interrupt feature may be chosen for use with both buffers, either buffer, or neither buffer. After emission of one interrupt signal by buffer control, all subsequent interrupt signals initiated within 0.5 milliseconds are ignored. If the first signal interrupts the main program immediately then, 500 microseconds of subsequent computation are free from imminent interruption.

INSTRUCTIONS

External Function instructions are used to select the buffer to be used, to establish either fixed block or variable block mode of operation, to start tape movement and data transfers, and to otherwise perform controlling functions. All External Function instructions which involve the tapes and the tape control units must specify the Tape Control Unit (TCU) to be selected:

Select TCU 1 bit is IOB_{30}; Select TCU 2 bit is IOB_{31}.

All External Function instructions which involve buffer memory and buffer control must specify the buffer to be selected:

Select Buffer 1 bit is IOB_{27}; Select Buffer 2 bit is IOB_{28}.

Transmissions from the tape to a buffer are initiated by executing an EF Read Tape instruction which begins tape movement and subsequent data transfer. Transmissions from the buffer to the tape are initiated similarly by executing an EF Write Tape instruction. External Read and External Write instructions are required only for transmissions of data through IOB, i.e., for transmissions between the computer and a buffer.

The instructions presented subsequently are required to effect data transfer between the computer and magnetic tape, via the buffer/tape system. A complete table of IOB select bits for these instructions is given on page 11. The relationship and order of the instructions are shown in the diagrams on pages 14 and 15.

MJ TEST FOR BUFFER ACTIVITY

The function of the Manually Selective Jump instruction, MJjv, for the 1103A is as follows: "If j is 0, take (v) as the next instruction. If j is 1, 2, or 3, and a corresponding jump selection was made manually, take (v) as the next instruction." This instruction is extended for use with the buffer/tape system to include j factors of 4, 5, 6, or 7. The extended instruction interprets conditions established by buffer control as being equivalent to manual settings of j = 4, 5, 6, or 7. (The use of j = 6 or 7 is explained in subsequent paragraphs.)

The normal condition for the j factors of 4 and 5 has them set to their non-jump status, i.e., not equal to 4 or 5. Only during buffer activity are these factors in their jump status. Upon completion of buffer activity, the $j \neq 4$ or 5 condition is re-established. Note that a Master Clear and a buffer Clear retain the $j \neq 4$ and 5 condition.

A judiciously programmed MJjv, j = 4 or 5, can be used to determine when computer instructions are needed to transfer data between a buffer and the computer, in either direction. Such an MJjv executed during buffer activity yields a program jump to (v). For example, a jump condition of j equal to 4 indicates that buffer 1 is active - involved in either (1) reading the tape, or (2) writing the tape. In case (1), after reading is completed, and buffer 1 has been switched from the load to the unload state, j is returned to its non-jump condition. An MJ4v executed at this point does not jump to (v), thus indicating the need to unload this buffer. The $j \neq 4$ condition remains in effect until this buffer again becomes active. In case (2), after writing is completed, and buffer 1 has been switched from the unload state to the load state, j is returned to its non-jump condition. An MJ4v executed at this point does not jump to (v), indicating the need to load this buffer if writing is to be continued. The $j \neq 4$ condition remains in effect until this buffer again becomes active.

A j factor of 5 is similarly used in testing for buffer 2 activity.

During concurrent use of both buffers, it may be desirable to program two consecutive MJ instructions (one with j = 4 and one with j = 5) to determine which buffer is currently available for data transfer. During concurrent operation of both buffers, both using the program interrupt feature, two consecutive MJ instructions must be programmed to determine which buffer signalled completion of activity.

It has been stated that normally the j factors are equal to 4 or 5. In Bypass operation of either one or both buffers, the buffer(s) being bypassed are never active. In as much as an active buffer is the usual means of establishing a jump condition, MJ4v never jumps to (v) if executed while bypassing buffer 1, and/or MJ5v never jumps to (v) if executed while bypassing buffer 2.

One other case wherein the j factors 4 and 5 are set to their jump condition is during a move tape operation (exclusive of reading or writing), normal mode, fixed or variable block length. While moving tape on any tape unit connected with buffer 1, an MJ4v jumps to (v). Thus, this instruction can be used to test for completion of tape movement in anticipation of reading or writing the same tape, or in anticipation of any subsequent tape operation with any tape unit connected to buffer 1. An MJ5v is used to check for tape movement on a tape unit connected with buffer 2. (In only one situation can data transfers - between the computer and the buffer testing as being active - be underway while tape is moving the specified number of blocks. This situation is described on page 16, "temporary suspension of buffer loading or unloading.")

EF WRITE BUFFER

This External Function instruction is programmed in a computer-to-buffer data transfer routine. The function of this EF is to select the particular buffer which is to receive data, and prepare the selected buffer for the subsequent transmission of words from the computer.

The Write Buffer bit is IOB_8. One of the Select Buffer bits must also be provided by this instruction.

EXTERNAL WRITE

External Write (EWjv) instructions are executed following the EF Write Buffer instruction. An External Write instruction is required for each word to be transferred to the buffer. If 120 words are to be written on the tape, 120 EW instructions are executed. The length of a variable block is determined by the number of EW instructions executed, i.e., the number of words sent to the buffer, since the buffer is completely emptied when the buffer-to-tape transfer is effected. The External Write instructions can be executed under the control of a Repeat instruction since buffer storage can accept words from IOB as fast as the repeated EW can transmit words to IOB. Consequently, the number of External Write instructions is governed by the quantity "n" in the Repeat instruction code, RPjnw.

EF END TRANSFER

If less than 120 words are transferred in a computer-to-buffer routine, an End Transfer instruction must be executed after the last External Write. The function of this instruction is to switch the buffer from the load state to the unload state. This instruction is necessary only in writing a block of variable length.

The End Transfer bit is IOB_9. A Select Buffer bit must also be provided by this instruction.

EF WRITE TAPE

The EF Write Tape instruction initiates tape movement and subsequent writing of block(s) of data on the tape. Data is received automatically from buffer storage, one word at a time. This instruction, executed once only, furnishes the tape control unit selected with all the information required for writing any number of blocks.

This instruction must provide the Select TCU bit, and specify the writing density, the interblock spacing, the interblockette spacing, fixed block or variable block mode operation, and the tape unit to be used. If only one block of data (fixed or variable length) is to be transferred from the buffer, the stop code may be provided by this instruction. If the use of the interrupt feature is desired, the interrupt bit should also be provided by this instruction.

EF READ TAPE (FORWARD OR BACKWARD)

This External Function instruction is programmed to initiate tape movement and data transfer from the tape to buffer storage. Data is transferred, one word at a time to the buffer, after the tape has been accelerated to its free-running speed. This instruction, executed only once, provides for the automatic transmission of any number of blocks, fixed or variable length, to the buffer.

If tape movement is in a backward direction, the last word in the block is encountered and read first, and hence is stored in the first position in the buffer. The words of the block are thus stored in reverse order in the buffer. Correct storage of the block in the computer is effected by reading the buffer "backward" during the buffer to computer transfer. The procedure is automatic and of no concern to the programmer, since the appearance of the block in computer storage is the same whether read backward or forward.

The EF Read Tape instruction must provide a Select TCU bit, the Read Forward or Read Backward bits, if required, the variable block bits, and the tape unit number. If only one block of data (fixed or variable length) is to be read, the stop bits can be included in the code. If the interrupt feature is to be used, the interrupt bit is also provided by this instruction.

MJ TEST FOR READ ERROR

A Manually Selective Jump instruction, MJjv with j = 6 or 7, checks for an error incurred during reading fixed or variable length blocks, normal mode operation. The mechanics of extending the MJ instruction are explained under "MJ Test for Buffer Activity."

Detection of a sprocket error or parity error in reading a tape to buffer 1 sets the j factor of 6 to its non-jump condition. Either of these read errors incurred in reading to buffer 2 sets j factor of 7 to its non-jump condition. These non-jump conditions of $j \neq 6$ or $j \neq 7$ are established upon end of block detection. Upon execution of MJ6v or MJ7v then, (v) is referenced for the next instruction except when an error has been detected while reading. An MJ6v instruction is programmed to follow completion of buffer 1 activity, as detected by an MJ4v instruction and/or a program interrupt (if chosen). An MJ7v instruction follows buffer 2 activity similarly.

No computer faults result from read errors. Tape movement is automatically stopped upon detection of the end of the block containing the error. When reading blocks of fixed length, normal operation, the type of detected error can be determined by reading the Buffer Word Counter, BWK. (See EF Read Count instruction.) If the detected error is a parity error only, BWK has a count of 120. If BWK $<$ 120, then less than 720 recorded lines were read from a block having at least 714 lines. In case of this error, the normal end-of-block indication is generated; however, buffer storage contains only 119 words. If 127 $>$ BWK $>$ 120, it can be assumed that more than 720, but no more than an additional six, lines were sensed before the space between blocks was detected. Recovery from this error can be attempted by re-reading at low gain. A count of 127 in BWK indicates that 727 or more lines were read. In case of this error, end of block detection occurs with the first block or blockette space following the 727th line, whereupon tape movement is stopped automatically. The stop generated will be in the unrecorded area between blocks or blockettes unless the blockette space is 0.1 inch. Probable causes of this error could be faulty recording (inasmuch as a block space is non-existant between any of the lines 714, 715, ..., 726, 727),or attempting erroneously to read a tape recorded with variable length blocks.

During normal operation, no indication of the nature of a read error is sent to IOA. During bypass operation in reading both blocks of fixed and variable length, IOA receives indication of the parity error and the sprocket error (mod 6 error during variable mode). Upon detection of a read error while reading variable blocks in normal operation, the programmer may desire to change to bypass operation to determine the type of error.

EF READ BUFFER

This External Function instruction is programmed in a buffer-to-computer data transfer routine. The function of this instruction is to select a buffer, and prepare the selected buffer, for subsequent transmission of words to IOB. Provided that the buffer is ready for unloading, the first word from buffer memory is transmitted to IOB with the execution of this instruction. Successive words are sent to IOB when IOB signals that it is ready to accept them.

The Read Buffer bit is IOB_7. One of the Select Buffer bits must also be provided by this instruction.

NOTE

Assume the EF Read Buffer instruction is executed before indication of completed buffer activity, i.e., during data transfer from the tape to the buffer. Until the buffer is ready to be unloaded to the computer, another instruction referencing IOB creates an IOB lockout and a computer stall. Also, only if the MJ Test for Read Error follows reading a block does a program jump to (v) indicate that the block was read without error. For these reasons it is advised not to execute the EF Read Buffer until completion of buffer activity.

EXTERNAL READ

External Read instructions (ERjv) are executed following the EF Read Buffer instruction. As each word is received by IOB from the buffer, an External Read must be executed to transmit the word to computer storage. Every word stored in buffer memory need not be transferred to the computer. However, the number of External Reads executed must not exceed the number of words in the buffer. The External Read instructions can be executed under the control of a Repeat instruction, RPjnw, with the quantity "n" governing the number of ER's executed.

EF READ COUNT

The EF Read Count instruction provides a means for obtaining the contents of the Buffer Word Counter. The Buffer Word Counter, BWK, tallies a count of the number of words received by the buffer memory, not in excess of 120. (A count above 120 indicates a block with more than 720 lines, as described under "MJ Test for Read Error"). Upon execution, the EF Read Count transmits the contents of BWK to the u-address portion of IOB. An External Read instruction transfers the count to computer storage.

The EF Read Count instruction is required when blocks of unknown length have been read from tape. When a variable block of unknown length is read, this EF determines the number of words received by the buffer. As stated previously, the number of External Read instructions must not exceed the number of words stored in the buffer memory. Following computer storage, the word count is available for use as a counter. For example, the word count may be "masked" directly into a Repeat instruction, RPjnw, to serve as "n" in governing the number of External Reads.

The Read Count select bit is IOB_6; a Select Buffer bit must also be provided by this instruction.

EF CLEAR BUFFER

When less than the number of words in the buffer memory are read into the computer, this instruction follows the last External Read. The EF Clear Buffer is required when the programmer chooses to transfer only part of a block (fixed or variable length) to the computer from the buffer. This instruction accomplishes the actions effected automatically when the buffer is completely emptied, i.e., the buffer memory is cleared, the Buffer Word Counter is reduced to zero, and the buffer is switched to the load state. (A computer Master Clear and buffer Clear have the same effect.)

The Clear Buffer bit is IOB_{10}; one of the Select Buffer bits must also be provided by this instruction.

EF STOP

An EF Stop instruction is executed to terminate tape reading and writing. This instruction, executed after reading or writing a block, brings tape movement to a halt in the space following this block. The EF Stop need not be executed within any time limitation after the last desired block is read or written, as a temporary automatic tape stop is imminent within a certain interval. (See Automatic Tape Controller, page 18.)

Consider first, stopping a free run read or write operation, fixed block length, normal mode. Reading: The program interrupt (if chosen) and/or no jump on MJ indicates the following: the end of block has been detected, 120 words have been read from the tape, and the buffer is ready for unloading. During reading then, the EF Stop is executed either at this indicated time prior to buffer unloading, or following buffer unloading. Writing: A program interrupt (if chosen) and/or no jump on MJ indicates the following: the buffer has been emptied of its contents, the (last) block of 120 words has been written on the tape, and the buffer is ready for re-loading. The EF Stop is executed at this time, i.e., after unloading the buffer to the tape.

Consider stopping a free run read or write operation, variable block length, normal mode. Reading: The program interrupt (if chosen) and/or no jump on MJ is indicative of end of block detection, as in the previous paragraph. The EF Stop is executed either at this time or following buffer unloading. Writing: The EF Stop is executed following a program interrupt (if chosen) and/or no jump on MJ, these being indicative of an emptied buffer, as quoted in the previous paragraph. Note that in the normal mode, blocks of variable length may be written without stopping between blocks.

To read a single block - fixed or variable length - and stop, the EF Stop might be executed immediately following the EF Read Tape instruction (or at a time thereafter as in free run reading). Tape movement is halted following detection of the end of the first block. To write a single block - fixed or variable length - and stop, the EF Stop might be executed immediately following the EF Write Tape instruction (or at a time thereafter as in free run writing). Tape movement is brought to a halt after the buffer is emptied of its contents - of 120 words in writing a fixed length block, or 120 words or less in writing a variable length block.

Only when initiating single block reading or writing should an EF Stop be executed at a time other than after an interrupt and/or the failure to jump on MJjv.

The usual procedure in reading or writing a single block is to include the stop bits in the code provided by the EF Read Tape or EF Write Tape instruction. In this case, reading or writing is terminated, as is tape movement, upon completion of reading or writing one block of either fixed or variable length. (In bypass operation, stop bits cannot be provided by the EF Write Tape for writing a single block of variable length.)

The stop bits are $IOB_{23, 22}$. An EF Stop instruction must provide these bits and a Select TCU bit. No tape unit selection need be nor should be provided by an EF Stop instruction.

MJ TEST FOR END OF RECORD

A Manually Selective Jump instruction, MJ6v or MJ7v, in addition to being used as a check for a read error, can be used to detect the end of record on tape, with the stipulation that the tape is being read during normal mode operation. The end of record indication informs the programmer of a four inch (nominal) dead space following the last word read. This unrecorded space may indicate end of tape, end of recording on the tape, or a break between groups of recorded data.

Upon detection of this dead space, tape movement is stopped automatically, j is set $\neq$ 6 or 7 (for buffer 1 or 2, respectively), the buffer is switched from the load to the unload state, and an interrupt is signalled, if this optional feature was selected. All indications are that a block of data has been received by the buffer except for the fact that the Buffer Word Counter has not advanced beyond its count of zero. To distinguish between a read error and an end of record indication, the procedure is as follows:

Failing to jump to (v) on MJ6v or MJ7v, program an EF Read Count instruction, an External Read to transfer the word count to storage, and a zero check on the word count. Only if the word count is zero has an end of record area been found.

The programmer may be aware of the end of recorded data, by knowledge of the number of records or blocks on the tape, or otherwise. If his program is coded to terminate reading prior to tape movement into the end of record dead space, then, any failures to jump on MJjv, j$\neq$6 or 7, indicate reading errors.

To return a buffer to the load state following end of record detection, program an EF Clear Buffer, or depress the Master Clear button or one of the buffer Clear buttons on the console.

EF BYPASS

This instruction enables the programmer to "bypass" or disregard the buffers in data transfer between the tape and the computer. The buffers are not affected in any way by tape instructions following the EF Bypass instruction. Either buffer can be bypassed. One buffer may be operating in the normal mode while the other buffer is being bypassed. Also, both buffers may be in the bypass condition simultaneously. However, in this last case, reading and/or writing cannot occur simultaneously because of common useage of IOB. The Bypass mode can be used to read or write blocks of more than 120 words, and it may be used in determining the nature of a read error incurred during normal mode operation, while reading variable blocks.

The Bypass select bit is IOB_{11}; one of the Select Buffer bits must also be provided by this instruction. The Bypass mode may also be selected manually at the control console.

A buffer is returned to normal operation by a computer Master Clear, the appropriate buffer Clear, or one of the following instructions: EF Clear Buffer, EF End Transfer, EF Read Count, EF Write Buffer, EF Read Buffer. Which instruction is used is determined by the operation desired and by the state, load or unload, in which the buffer was left at the time of bypassing it. If the buffer was left in the load state, any of the instructions listed are useable except the EF Read Buffer. If the buffer was left in the unload state, any of the instructions are useable except the EF Write Buffer.

The memory of a buffer being bypassed can be used as temporary storage. Data from the computer or magnetic tape can be held in the buffer until normal mode operation is resumed. Resume normal operation as follows:

1) To transfer this data to the computer, program an EF Read Buffer instruction.
2) To transfer this data to the tape, program an EF End Transfer or EF Read Count instruction preceding the EF Write Tape instruction.

BYPASS OPERATION

In most instances, bypass operation of the 1105 is equivalent to operation of the 1103A magnetic tape system. A few characteristics of bypass operation are noted briefly in the following paragraphs.

During bypass operation, fixed block length mode, a sprocket error detected while reading from tape sends a "1" to IOA_3; a parity error is indicated by a "1" in IOA_0. An External Read of IOA must follow the External Read of the last word in each block. Both errors stop tape movement following the block with the error. To stop tape movement, barring either error, program an EF Stop instruction after the External Read of IOA.

During bypass operation, variable block length mode, errors detected during reading are indicated in IOA. A mod 6 error sends a "1" to IOA_3; a parity error sends a "1" to IOA_0. IOA receives indication of either of these read errors only upon detection of the end of the block. Neither of these read errors stops tape movement automatically. End of block detection in itself sends a "1" to IOA_1. Detection of the end of recording sends a "1" to IOA_2. Tape movement is stopped automatically upon end of record detection. An External Read of IOA must precede each External Read of IOB. Only if IOA is zero should IOB be read. IOA not equal to zero is indicative of the end of the record, or the end of the block with or without parity error and/or mod 6 error. To program a stop of tape movement, follow the check of IOA with an EF Stop instruction. Tape movement can be stopped only between blocks.

To read a single block, either fixed or variable length, include the stop code in the EF Read Tape instruction, or follow the IOA checks, as discussed above, with an EF Stop instruction.

To terminate writing, fixed or variable block length, program an EF Stop instruction to follow the last External Write instruction. In fixed block writing, the last External Write instruction is the 120th. In variable block length writing, the EF Stop, in effect, determines the length of the block in as much as writing and tape movement are terminated immediately, thereby creating a block space.

To write a single block of fixed length, include the stop code in the EF Write Tape instruction. In writing a block of variable length, program an EF Stop instruction after the External Write for the last word in the block. Writing must be stopped after each block of variable length. Unlike writing variable blocks during normal operation, writing in the free-run mode is impossible during bypass operation.

EXTERNAL FUNCTION BIT ASSIGNMENTS

IOB_{31}	=	1 - Select TCU 2
IOB_{30}	=	1 - Select TCU 1
IOB_{28}	=	1 - Select Buffer 2
IOB_{27}	=	1 - Select Buffer 1
IOB_{26}	=	1 - Select Magnetic Drum Zone Switch*
IOB_{24}	=	1 - Program Interrupt
IOB_{23}-IOB_{22}		Select Rewind or Stop Tape
	=	01 - Rewind
	=	10 - Rewind with Interlock
	=	11 - Stop Tape
IOB_{21}		Select Block Space
	=	0 - Select 1.2" Interblock Space
	=	1 - Select 2.4" Interblock Space
IOB_{20}-IOB_{19}		Select Blockette Space or Variable Block Option
	=	00 - Select 0" Interblockette Space
	=	01 - Select 0.1" Space
	=	10 - Select 1.2" Space
	=	11 - Select operation in variable block length mode or continuous data input mode.
IOB_{18}-IOB_{16}	=	100 - Switch to continuous mode if in variable block mode and vice versa.
		Select Tape operation
	=	001 - Read Forward
	=	010 - Move Forward
	=	011 - Write Forward (high density)
	=	101 - Read Backward
	=	110 - Move Backward
	=	111 - Write Forward (low density)
IOB_{15}	=	1 - Select Zone B*
IOB_{15}-IOB_{12}		Select Tape Unit or Reading Level
	=	0001 - Select Tape Unit 1
	=	0010 - Select Tape Unit 2
	=	etc.
	=	1010 - Select Tape Unit 10
	=	1101 - Select Normal Gain
	=	1110 - Select High Gain
	=	1111 - Select Low Gain
IOB_{11}	=	1 - Select Bypass
IOB_{10}	=	1 - Select Clear Buffer
IOB_{9}	=	1 - Select End Transfer
IOB_{8}	=	1 - Select Write Buffer
IOB_{7}	=	1 - Select Read Buffer
IOB_{6}	=	1 - Select Read Count
IOB_{11}-IOB_{0}		Select number of blocks to be moved in a Move Forward or Move Backward operation
	=	0 --- 01 - Move 1 block
	=	0 ---010 - Move 2 blocks
		. . .
		etc.

---

* The magnetic drum on the 1105 has two zones providing a total of 32,768 addressable storage locations. In zone A, 16,384 words are addressable; in zone B, another 16,384 words are addressable. Addresses 40000 through 77777 are used to reference words in both zones. Zone A is referenced following a computer Master Clear or an EF instruction providing IOB_{26}, the select bit for the magnetic drum zone switch. Zone B is referenced following a EF instruction providing IOB_{26} and IOB_{15}, the select bit for zone B.

Following are examples illustrating the use of the External Function bit assignments.

TAPE INSTRUCTIONS FIXED BLOCK

Function	Buffer/ TCU Number	Direction of Tape Movement	Number of Blocks	Blockette Spacing (inches)	Block Spacing (inches)	(v) of EF-v
Write high density tape on tape unit No. 1	1	Forward	Free run	0.0	1.2	01 00006 10000
					2.4	01 00106 10000
				0.1	1.2	01 00026 10000
					2.4	01 00126 10000
				1.2	1.2	01 00046 10000
					2.4	01 00146 10000
Write low density tape on tape unit No. 1	1	Forward	1 block	0.0	1.2	01 00616 10000
					2.4	01 00716 10000
	2	Forward	Free run	0.1	1.2	02 00036 10000
					2.4	02 00136 10000
				1.2	1.2	02 00056 10000
					2.4	02 00156 10000
Write high density tape on tape unit No. 10 (with interrupt)	2	Forward	Free run	0.0	1.2	02 01007 20000
Read tape unit No. 1	1	Forward	Free run			01 00002 10000
Read tape unit No. 2	1	Forward	1 block			01 00602 20000
Read tape unit No. 7	1	Backward	Free run			01 00012 70000
Read tape unit No. 1	2	Backward	1 block			02 00612 10000
Move tape unit No. 6 N blocks*	2	Forward	$N=n_1n_2n_3n_4$			02 00004 $6n_1n_2n_3n_4$
Move tape unit No. 1 N blocks*	1	Backward	$N=n_1n_2n_3n_4$			01 00014 $1n_1n_2n_3n_4$

TAPE INSTRUCTIONS VARIABLE BLOCK

Write high density tape on tape unit No. 4	1	Forward	1 block	0.0	1.2	01 00666 40000
Read tape unit No. 3	2	Forward	Free run			02 00062 30000
	1	Backward	Free run			01 00072 30000
Move tape unit No. 3 N blocks*	1	Backward	$N=n_1n_2n_3n_4$			01 00074 $3n_1n_2n_3n_4$
Stop tape unit**	1					01 00600 00000

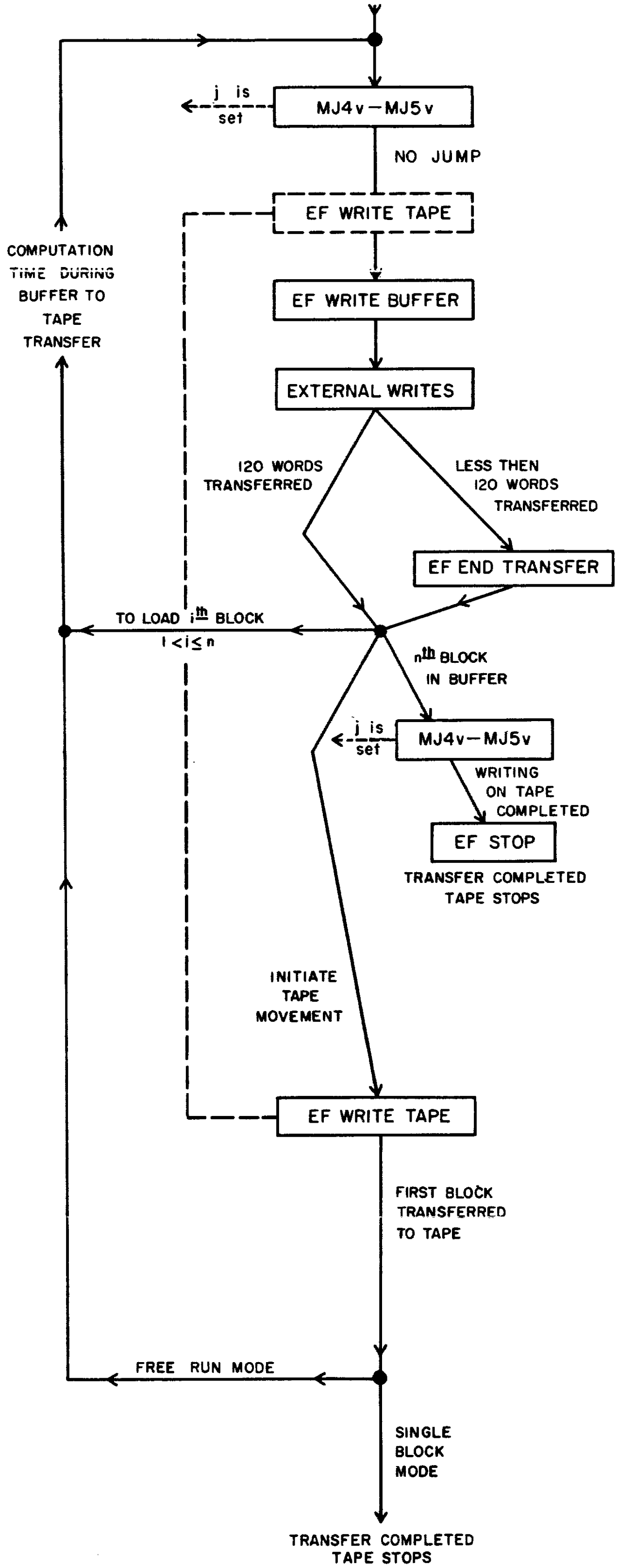
BUFFER INSTRUCTIONS

	Buffer No. 1	Buffer No. 2
Bypass Buffer	00 10000 04000	00 20000 04000
Clear Buffer	00 10000 02000	00 20000 02000
End Transfer	00 10000 01000	00 20000 01000
Write Buffer	00 10000 00400	00 20000 00400
Read Buffer	00 10000 00200	00 20000 00200
Read Count	00 10000 00100	00 20000 00100

* Stop codes must not be selected in a Move operation.

** To stop a tape, the tape unit number need not be given. To avoid an erroneous tape unit number selection, select only TCU number and stop code.

DATA TRANSFER FROM COMPUTER STORAGE TO MAGNETIC TAPE



COMPUTER TO BUFFER TRANSFER

Check for buffer activity with MJjv instruction. If the interrupt feature is being used, the MJ instruction determines which buffer is inactive and ready to receive data. Programmer must ascertain that buffer is in load state.

Execute EF Write Buffer instruction to select buffer and connect IOB output to IOT input.

Execute an EW instruction for each word to be transferred from magnetic core storage to IOB. The transfer of each word from IOB to IOT, and hence to buffer storage, is automatic. As each word is received by the buffer, the word counter is advanced.

After 120 words are transferred, buffer is automatically set to unload state. If less than 120 words are transferred, execute EF End Transfer instruction to set buffer to unload state.

STOPPING TAPE MOVEMENT FREE RUN MODE

For the free run mode, the program must keep a record of the number of blocks to be transferred. The EF Stop instruction is executed after the last block has been written on the tape.

BUFFER TO TAPE TRANSFER

Execute EF Write Tape instruction to start tape movement. If only one block is to be transferred from the computer to the tape, include the stop code in this EF instruction. Also, the interrupt bit, if desired, is included in this instruction. This EF can be executed preceding the EF Write Buffer instruction. Programming thusly hastens writing of the first block on tape, since the tape is accelerating during the computer to buffer transfer. The EF Write Tape instruction need be executed only once, regardless of the total number of blocks to be written (single block only or free run mode for writing n blocks).

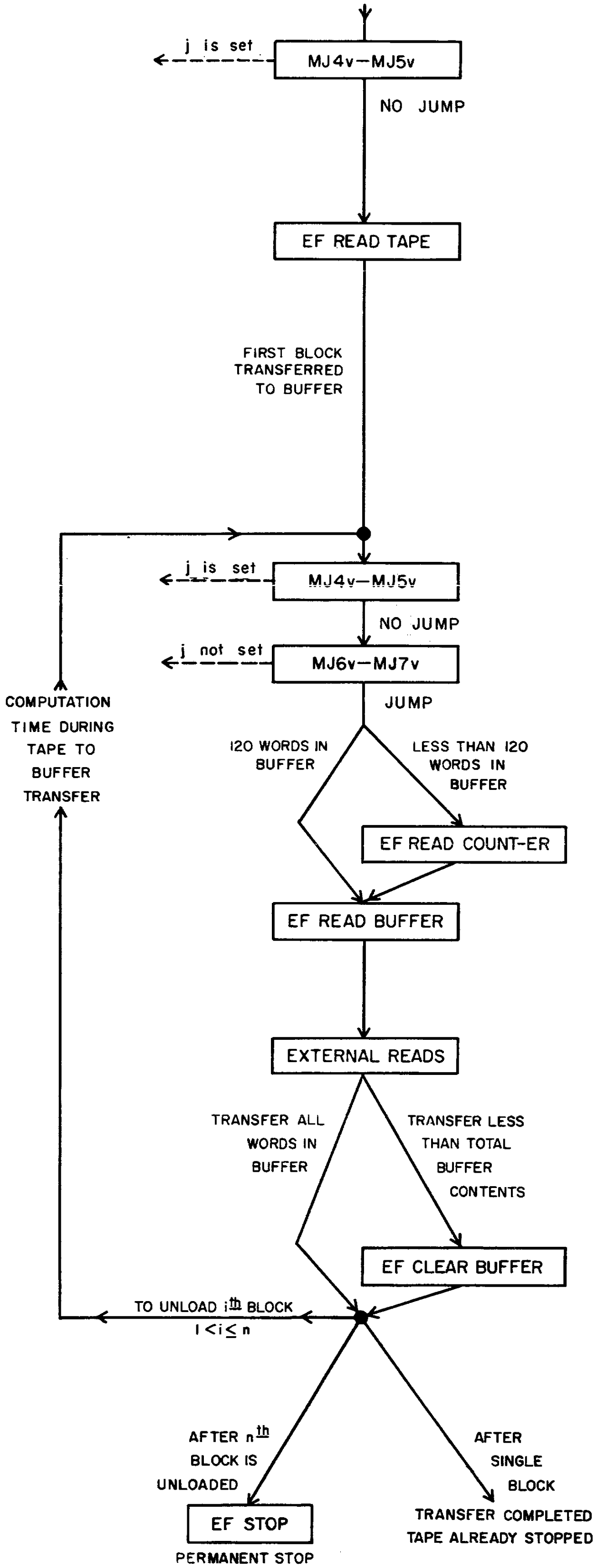
Transmission proceeds automatically after tape acceleration to full speed. The word counter is reduced by one for each word written.

Termination is automatic when word counter is reduced to zero. At this time, the buffer is automatically set to the load state.

STOPPING TAPE MOVEMENT SINGLE BLOCK MODE

If only one block is to be transferred from the computer to the tape, the stop code is included in the EF Write Tape instruction. Tape movement is stopped automatically then, after the transfer of the buffer contents to the tape.

DATA TRANSFER FROM MAGNETIC TAPE TO COMPUTER STORAGE



TAPE TO BUFFER TRANSFER

Check for buffer activity with MJjv instruction. Buffer must be in load state.

Execute EF Read Tape instruction to start tape movement and tape reading. This instruction need be executed only once, regardless of the total number of blocks to be read (single block only or free run mode for reading n blocks). If only one block is to be read from tape, include the stop code in the EF instruction. Also, the interrupt bit, if desired, is included in this instruction.

Transmission proceeds automatically. With each word received, the word counter is advanced. Termination is automatic when word counter is 120, or when an end of block signal is received from variable block length tape. At this time, the buffer is automatically set to its unload state. If the interrupt feature is being used, the interrupt signal is emitted after the switch to unload is completed.

BUFFER TO COMPUTER TRANSFER

Check for buffer activity with MJjv instruction. Programmer must ascertain that buffer is in unload state. If the interrupt feature is being used, the MJjv (j = 4 or 5) determines which buffer is inactive and in the unload state. If an error occurred in reading the block from tape to buffer, jump to an error check routine.

If the buffer contains less than 120 words, an EF Read Count and an External Read instruction must precede the EF Read Buffer instruction. The stored word count is used later in the program. Execute EF Read Buffer instruction to connect IOT output to IOB input. This instruction (or the EF Read Count), if initiated (unadvisedly) during the tape to buffer transfer, does not become effective until now.

Transmission of each word to IOB is automatic. With each transmission, the buffer word counter is reduced by one. Execute an External Read instruction to transmit each word from IOB to computer storage. The number of ER instructions must not exceed the number of words stored in the buffer.

Transmissions to IOB cease when buffer is empty, i.e., when word counter is reduced to zero. At this time, the buffer is set to load state. If less than the number of words in the buffer is desired, the last ER must be followed by an EF Clear Buffer instruction to clear the buffer and set it to the load state.

STOPPING TAPE MOVEMENT

If only one block is to be transferred from the tape to the computer, the stop code is included in the EF Read Tape instruction. Tape movement is stopped automatically then after the transfer of one block to the buffer from the tape.

For the free run mode, the program must keep a record of the number of blocks to be transferred. The EF Stop instruction is executed either preceding or following (as shown here) the transfer of the last block to the computer.

TEMPORARY SUSPENSION OF BUFFER LOADING OR UNLOADING

A special case exists which allows buffer loading from the computer to be suspended temporarily while reading from another piece of equipment. In other words, it is possible to transfer a few words from the computer, read another piece of equipment, or the other buffer, and then continue loading the first buffer from the computer. The sequence of instructions used for this type of operation is as follows:

(1) EF Write Buffer (buffer 1)
(2) Repeated External Write instructions
(3) Instructions required to read other equipment, possibly buffer 2. (Writing with other equipment is not possible here; moving or rewinding tape, any tape unit, is possible.)
(4) EF Write Buffer (buffer 1)
(5) Repeated External Write instructions

Note that this sequence does not include an EF End Transfer instruction after step (2). The EF Read Count instruction can be inserted between steps (2) and (3) or between steps (3) and (4). This instruction informs the programmer of the current word count in the buffer and aids in determining the number of External Write instructions to be executed in step (5) if the write buffer operation is to be continued at this time to its conclusion. If desired, successive suspensions of buffer loading can be programmed provided that each resumption of writing to the buffer is initiated by an EF Write Buffer instruction.

Also possible is the case whereby buffer unloading to the computer is suspended temporarily while writing to another piece of equipment. Using the following sequence of instructions, it is possible to read a few words from the buffer, write with another piece of equipment, or the other buffer, and then continue reading data from the first buffer to the computer. The sequence of instructions is as follows:

(1) EF Read Buffer (buffer 1)
(2) Repeated External Read instructions
(3) Instructions required to write to other equipment, including buffer 2. (Reading from other equipment is not possible; moving or rewinding tape, any tape unit, is possible.)
(4) EF Read Buffer (buffer 1)
(5) Repeated External Read instructions.

Note that this sequence does not include an EF Clear Buffer instruction after step (2). Step (3) requires an EF instruction to initiate writing to other equipment. This instruction clears IOB, thereby destroying the last word transferred from the buffer to IOB. The loss of this word should be kept in mind when buffer reading is resumed in step (5): one less External Read is required to complete unloading of the buffer. Successive suspensions of buffer reading can be programmed provided that each resumption of reading is initiated by an EF Read Buffer instruction. If buffer reading should be suspended following the External Read which transferred the next-to-last word to computer storage, no resumption of buffer reading is required or possible since the last word from the buffer is destroyed.

The following sequence of operations is possible: writing (reading) of buffer 1 is suspended, reading (writing) of buffer 2 is undertaken (and completed or suspended), and writing (reading) of buffer 1 is resumed. Of course, the appropriate Manually Selective Jump instructions should be executed, and the programmer must determine that the buffers are in the correct state (unload or load).

TIMING

TIMING FOR DATA TRANSFERS

The following intervals are defined to facilitate optimum programming of input/output operations. For those times depending upon the tape speed, the figures are based on a free-running speed of 100 inches per second. ALL TIMES QUOTED ARE THEORETICAL. From a practical standpoint, a minimum deviation of 10% should be allowed from these preliminary times given.

DATA TIME

This time may be considered as the time required to traverse the information contained in one block, exclusive of block and blockette spacings. A nominal recording density of 200 lines per inch results in a data time of 36 milliseconds for a block of 720 lines, or 120 computer words. A recording density of 215 lines per inch, as achieved on Univac II, results in a data time of approximately 33.5 milliseconds. A density of 250 lines per inch, as expected ultimately on the UNIVAC 1105 system, would result in a data time of approximately 28.8 milliseconds.

BLOCKETTE SPACE TIME

The optional blockette spaces are 0.0, 0.1, and 1.2 inches, requiring 0.0, 1.0, or 12 milliseconds, respectively, to traverse.

BLOCK SPACE TIME

The optional block spaces between fixed blocks are 1.2 inches and 2.4 inches. Times required to traverse these spaces are 12 milliseconds and 24 milliseconds, respectively.

TAPE TRANSFER TIME

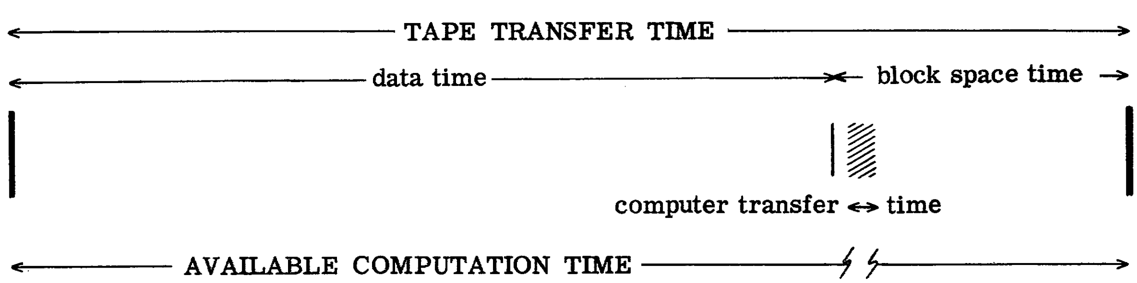
The transfer rate of words between the tape and the buffer is limited only by timing factors of the tape system. Assuming that the tape is in the free-run mode, the tape transfer time is the summation of the data time, block space time, and blockette space time. With a recording density of 200 lines per inch, 1.2 inch block spacing, and 0.0 blockette spacing, the transfer rate for words between the tape and buffer storage is 2500 words per second.

COMPUTER TRANSFER TIME

The transfer rate of words between the buffer and the computer is computer-limited only. Using repeated External Read instructions, words are transferred from IOB to computer storage at the rate of one word every 16 microseconds. Thus, a block of 120 words can be transferred from the buffer to the computer in approximately 1.9 milliseconds. Using repeated External Write instructions, words are transferred from computer storage to IOB at the rate of one every 20 microseconds. Thus, a block of 120 words can be transferred from the computer to the buffer in approximately 2.4 milliseconds.

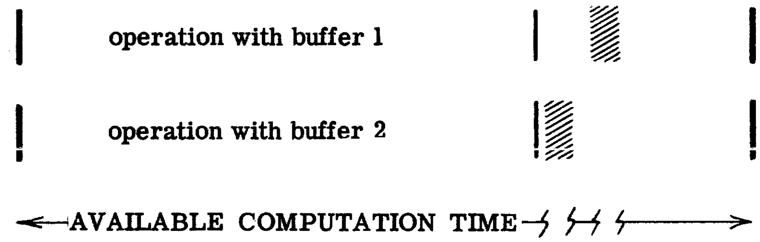
AVAILABLE COMPUTATION TIME

From a tape moving continuously in the free-run mode, 120 computer words are transferred to computer storage every 48 milliseconds (tape transfer time, assuming 200 lines per inch recording, 1.2" block spacing, and 0.0" blockette spacing). An uninterrupted average computation time of approximately 46 milliseconds is thus available when operation involves one buffer. Pictorially, the relationship of times can be shown as follows:



Thus, the available computation time during reading is equal to 48 ms - 1.9 ms = 46.1 ms; during writing, 48 ms - 2.4 ms = 45.6 ms.

To illustrate concurrent operation of both buffers, two such diagrams can be superimposed, exactly coincident as shown below, or displaced to any degree. How the 1105 lends itself readily to concurrent operations and computation is easily seen here.



If the interrupt feature is being used with both buffers during their concurrent operation, the interrupt signal from the second operation is lost if the second interrupt follows the first interrupt within 0.5 milliseconds. The lost interrupt, indicating completion of buffer activity, is detected by checking for a j factor not equal to 4 or 5. For this reason, in case the concurrent operations should exactly coincide timewise, or be displaced by less than 0.5 milliseconds, a Manually Selective Jump instruction, checking for the completion of activity of one buffer, should always be programmed to follow a "computer transfer routine" for the other buffer.

PROGRAMMED START AND STOP TIMES

The start time is the time elapsing between the execution of the EF Write Tape instruction, or the EF Read Tape instruction, and the writing or reading of the first word in the block. The start time, including tape acceleration time, is approximately 17 milliseconds (29.0 ms for 2.4 inch block space). The stop time is the time elapsing between the execution of the EF Stop instruction and the moment whereupon another EF instruction can be executed (referencing the same buffer system, any tape unit). This stop time, during which tape movement is brought to a complete halt, is approximately 8.5 milliseconds.

AUTOMATIC TAPE CONTROLLER (ATC)

The Automatic Tape Controller eliminates the task of programming data transfers between the computer and the buffer within time limitations imposed by the tape transfer rate. The Automatic Tape Controller automatically effects a temporary stop of tape movement if the data transfer between the computer and the buffer is not initiated at a time appropriately coincident with tape reading or writing. Specifically, during a reading operation (tape to buffer to computer), the tape is stopped after a nominal 2.5 millisecond lapse of time following detection of the end of the block, if, during this time, an EF Read Buffer instruction has not been executed. During a writing operation (computer to buffer to tape), the tape is stopped after a nominal 2.5 millisecond lapse of time following writing the last word of the block, if, during this time, an EF Write Buffer instruction has not been executed. The execution of the respective EF Read Buffer or EF Write Buffer instruction, after ATC has stopped tape movement, initiates a re-start of the tape. Accordingly, these instructions can be programmed at the convenience of the programmer, allowing him as much time as he desires between successive transfers of data between the buffer and the computer.

If the EF Read Buffer or EF Write Buffer is executed within 10 milliseconds following initiation of the ATC stop (between 2.5 and 12.5 milliseconds following detection of the end of the block), the re-start of tape movement is detained until this quoted interval has elapsed. A protective lockout, as required by the clutch and brake mechanisms of the tape unit, is active during this time. This lockout does not delay the execution of the EF instruction.

Upon re-starting the tape with an EF Read Buffer or EF Write Buffer instruction, approximately 9.5 milliseconds elapse (21.5 ms elapse for 2.4 inch block space) before the reading or the writing of the first word in the next block. During this time the transfer of data between the buffer and the computer must be completed in order that (in reading) the buffer can accept more data from the tape, or (in writing) the buffer has ready more data for transcribing on the tape.

The Automatic Tape Controller also eliminates the need for executing the EF Stop instruction within certain time limits in order to stop tape movement after reading or writing a specific block. The EF Stop can be executed to effect a "permanent" halt of tape movement at any time following the ATC temporary stop.

ATC stops tape movement 2.5 milliseconds after the end of the block is detected. (End of block detection occurs 250 microseconds after the last word is read in fixed block mode, and, in variable block mode, 600 microseconds after the last word is read. If prevention of this automatic tape stop is desired, the EF Read Buffer instruction must be executed within 2.34 milliseconds (2.5 ms - 160 microseconds) after the interrupt is effected.

TIMING DETAILS COMPUTER-TO-TAPE DATA TRANSFER

As shown in the diagram on page 14, the EF Write Tape instruction may precede the EF Write Buffer instruction. In programming thusly, care must be taken that the buffer is switched to the unload state before the tape start time is consumed: upon expiration of the start time, the tape expects to receive data. The start time is 17 milliseconds (for 1.2 inch block spacing) within which time the EF Write Buffer and Repeat-External Write instructions (and EF End Transfer instruction) should be executed. The failure to program within this time results in a fault.

Any amount of time is available between the EF Write Tape instruction and the <u>second</u> EF Write Buffer instruction. (See ATC discussion.) If the automatic tape stop, as effected by ATC, is not desirable, then the time allowable between the EF Write Tape and the second EF Write Buffer is determined as follows:

Tape starting time	=	17 ms (for 1.2" block space)
Data time	=	36 ms (for 200 lines per inch recording)
Time allowed by ATC	=	2.5 milliseconds

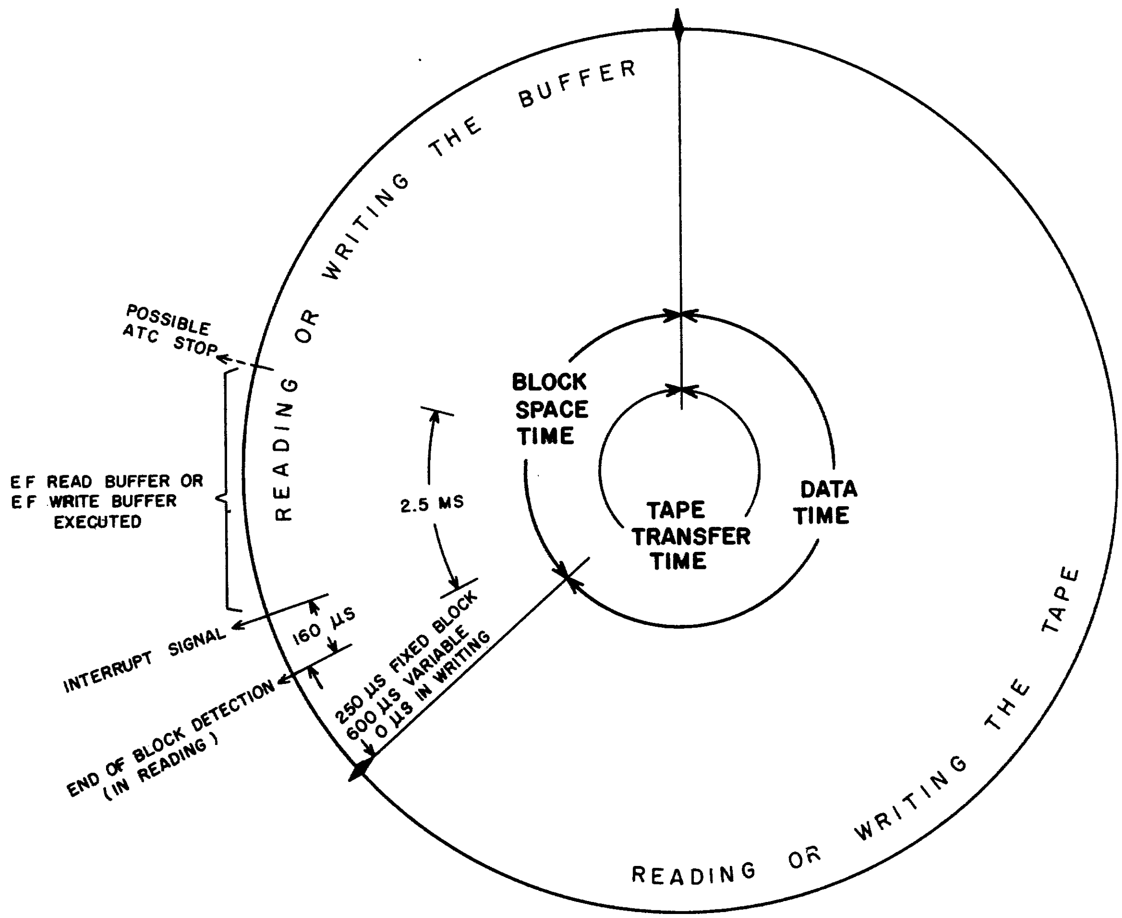
If the interrupt feature is being used, ATC does not stop the tape if the EF Write Buffer is executed within 2.34 milliseconds after the interrupt signal is received by the program. (The ATC timing of 2.5 milliseconds is started with the end of block indication; the interrupt signal is not emitted until the buffer has been switched from its unload to load state, a process which consumes 160 microseconds.)

TIMING DETAILS TAPE-TO-COMPUTER DATA TRANSFER

The MJjv instruction, j = 4 or 5, jumps to (v), i.e., not into the "Read Buffer routine" until the buffer is set to the unload state. (As with all MJjv instructions, if a jump to the v address does not occur, then the next consecutive instruction is executed.) The time consumed and available between the EF Read Tape instruction and the moment whereupon the buffer is switched to the unload state, is determined as follows:

Tape starting time	=	17 milliseconds (for 1.2 inch block spacing)
Data time	=	36 milliseconds (for 200 lines per inch recording)
End of block detection time	=	250 microseconds (fixed block mode)
	=	600 microseconds (variable block mode)
Switching time from load to unload state	=	160 microseconds

Of course, the MJ instruction, j = 4 or 5, can be executed at any time during this interval. If the interrupt feature is being used, the interrupt signal is emitted after the time above is consumed, i.e., after the buffer is switched from the load to unload state. With or without the interrupt in effect, MJjv does not enter the "Read Buffer routine" until the buffer is switched to the unload state.



SUMMARY OF TIMES

NORMAL OPERATION, FREE RUN MODE

FAULT DETECTION

BUFFER FAULTS

Each of the buffer faults listed subsequently is indicated on the auxiliary control console. Each of the buffer faults is a computer B fault, leading to an immediate computer stop. If tape is being read or written at the time, a buffer fault leads to a transfer fault.

A select fault results from an attempt to load the buffer from the computer at a time the buffer is in the unload state. For example, the following sequence results in a select fault: an EF Write Buffer, 120 External Writes or less than 120 EW's followed by an EF End Transfer, and another EF Write Buffer. Also, a select fault occurs when an attempt is made to unload the buffer when the buffer is in the load state. In this case, the select fault would most likely be the result of malfunctioning equipment.

As a checking device a buffer counts the words as they enter the memory. If a word has not been stored in its proper memory location, the buffer initiates an address fault. (This fault is not initiated if a block is less than 20 words in length.) This fault would be generated by malfunctioning equipment.

A buffer IO fault is similar to an IOB Read Fault with the exception that the register involved is IOT, the in-out transfer register for a buffer. Similar to the IOB Read Fault, Class I, is the fault incurred should IOT receive a word from tape while still retaining the previous word from tape. This fault occurs while reading variable blocks, normal mode operation, if an attempt is made to load more than 120 words into a buffer. The fault is detected upon receipt of the second extra word from tape, in as much as IOT still holds the first extra word. Similar to an IOB Read Fault, Class II, is the fault occurring if a word in IOT, destined for IOB or magnetic tape, is not removed before IOT receives a word from tape. Generally, the buffer IO fault would be incurred by malfunctioning equipment.

TRANSFER FAULT

For each data channel between tape and the computer, there is a transfer fault indicator on the auxiliary control console. Illumination of either of these indicators notifies the programmer of one of the faults listed subsequently. These faults are either directly generated by faulty operation of the buffer and/or tapes, or are generated indirectly by a computer stop during buffer/tape operation. For every stop of computer operation, whether it be a programmed stop, a manual stop, or a fault stop, the buffer/tape system is notified. If a tape is currently being read or written, a transfer fault is incurred, and tape movement is halted at the end of the current block.

The transfer faults noted are computer B faults, leading to an immediate computer stop, if computer operation has not already been halted. Upon noting a transfer fault, inspection of the fault lights in the B Fault group on the computer console and the code word in the Tape Control Register, TCR, should enable the programmer to determine the cause of the fault. Recovery from all B faults requires a computer Master Clear. Following are the conditions generating transfer faults.

As with the Univac Scientific computer, an IOB Read Fault (Class I) occurs if IOB receives a second word from an external source before the first word received has been removed.

As with the 1103A tape system, a no information fault occurs if a word has not been received by the tape control unit at the time TCU is ready to transcribe a word on the tape. Indication of this fault is given by the MT light in the B Fault group on the computer console.

If a UNISERVO tape handler loses power while writing a tape, a write fault occurs. A light on the malfunctioning tape unit indicates this fault.

UNIVAC® - The FIRST Name in Electronic Computing Systems

U-1513

Printed in U.S.A