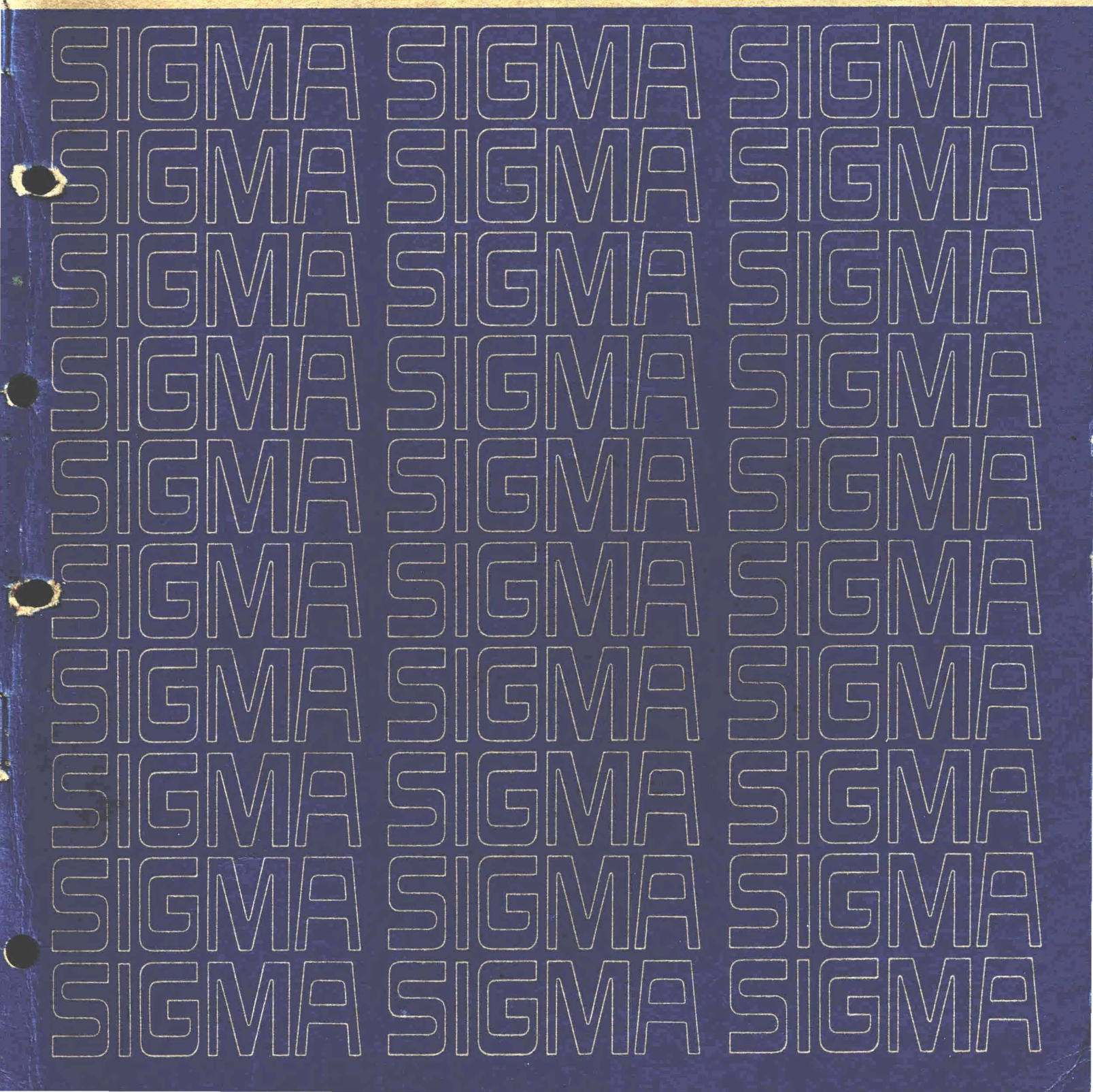
XDS

Xerox Data Systems

TECHNICAL MANUAL

MAGNETIC TAPE CONTROLLER MODEL NO. 7320



XDS 901561A
PRICE: $6.75

TECHNICAL MANUAL

MAGNETIC TAPE CONTROLLER

MODEL NO. 7320

July 1969

Scientific Data Systems A XEROX COMPANY 701 South Aviation Blvd./El Segundo, California 90245 (213) 772-4511

LIST OF EFFECTIVE PAGES

Total number of pages is 204, as follows:

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont.)

LIST OF ILLUSTRATIONS (Cont.)

LIST OF TABLES

RELATED PUBLICATIONS

The following publications contain information not included in this manual, but necessary for a complete understanding of the Magnetic Tape Controller Model 7320.

Publication Title	Publication Number
Magnetic Tape Station Model 7322, Technical Manual	901084
Magnetic Tape Station Model 7323, Technical Manual	901176
Power Supply Model PT20, Technical Manual	901157
Peripheral Equipment Tester Model 7901, Technical Manual	901004
Sigma 5 and 7 9-Channel Magnetic Tape System Test, Diagnostic Program Manual	901119
Sigma 5 and 7 9-Channel Magnetic Tape Test, Diagnostic Program Manual	901129
Sigma 2 9-Channel Magnetic Tape System Test, Diagnostic Program Manual	901149
Sigma 2 Magnetic Tape Test (9-Channel 75/150 Ips), Diagnostic Program Manual	901166
Diagnostic Control Program for Sigma 5 and Sigma 7 Computer Peripheral Devices, Reference Manual	900712
Diagnostic Control Program for Sigma 2 Computer Peripheral Devices, Reference Manual	900839
SDS Sigma Computer Systems/Interface Design Manual	900973
SDS Sigma 9-Track Magnetic Tape System, Reference Manual	900977
Multiplexing IOP, Models 8271/8471, Technical Manual	901515

SECTION I
INTRODUCTION

1-1 INTRODUCTION

This publication defines and describes the operation, programming considerations, theory of operation, installation, maintenance and performance testing of the Magnetic Tape Controller Model 7320 (figure 1-1), that is used in the Magnetic Tape Systems Models 7320/7322 (75 ips) and 7320/7323 (150 ips). These systems are designed and manufactured by Scientific Data Systems, El Segundo, California.

1-2 SCOPE OF MANUAL

The purpose of this publication is to provide the reader with a comprehensive description of the magnetic tape controller (controller) and its functions. The contents of the sections are:

a. Section I: An explanation of the related publications, a physical and a functional description, and the important characteristics of the controller are contained in this section.

b. Section II: Programming information and tape system instructions and responses are described in detail, and typical instructions are illustrated.

c. Section III: A detailed operational description of the controller is given on a block diagram level.

d. Section IV: A detailed theory of operation using equations, flow charts, block diagrams, and logic diagrams to describe fully the controller functions is presented in this section.

e. Section V: A list of applicable logic equations for the controller and a definitive description of each logic term are given.

f. Section VI: This section contains a list of applicable engineering drawings.

g. Section VII: This section contains specifications and leading particulars, module location charts, and installation data.

h. Section VIII: Information necessary for the maintenance of the controller is provided.

i. Section IX: An illustrated parts breakdown and a parts list are included in this section.

1-3 RELATED PUBLICATIONS

The publications listed in the front matter of this manual and described below contain information that is necessary for a complete understanding of the controller:

a. The magnetic tape station technical manuals describe the operation and the function of the stations used with the controller to make up a magnetic tape system.

b. The PT20 power supply technical manual describes the capabilities and the performance specifications of the power supply used in the magnetic tape systems.

c. The peripheral equipment tester (PET) technical manual describes the test equipment used in offline testing of the magnetic tape systems.

d. The 9-Track Magnetic Tape System Reference Manual provides additional operation and programming information.

e. The Sigma magnetic tape system diagnostic program manuals and the Sigma diagnostic control program manuals specify and illustrate programming instructions for the magnetic tape systems.

f. The Sigma computer systems interface design manual describes the input/output interface for Sigma series computers.

1-4 GENERAL DESCRIPTION

The controller is an integral part of a magnetic tape system and provides control for the station in which it is installed.

The tape system can be expanded to include seven other stations which are controlled by the one controller in the basic system. In automatic operation, the controller handles all operating commands and read or write data transfers for all stations in the tape system. The controller routes data to be written from the computer to the selected station, and routes data to be read from the selected station back to the computer. Figure 1-2 is a simplified block diagram of a magnetic tape system.

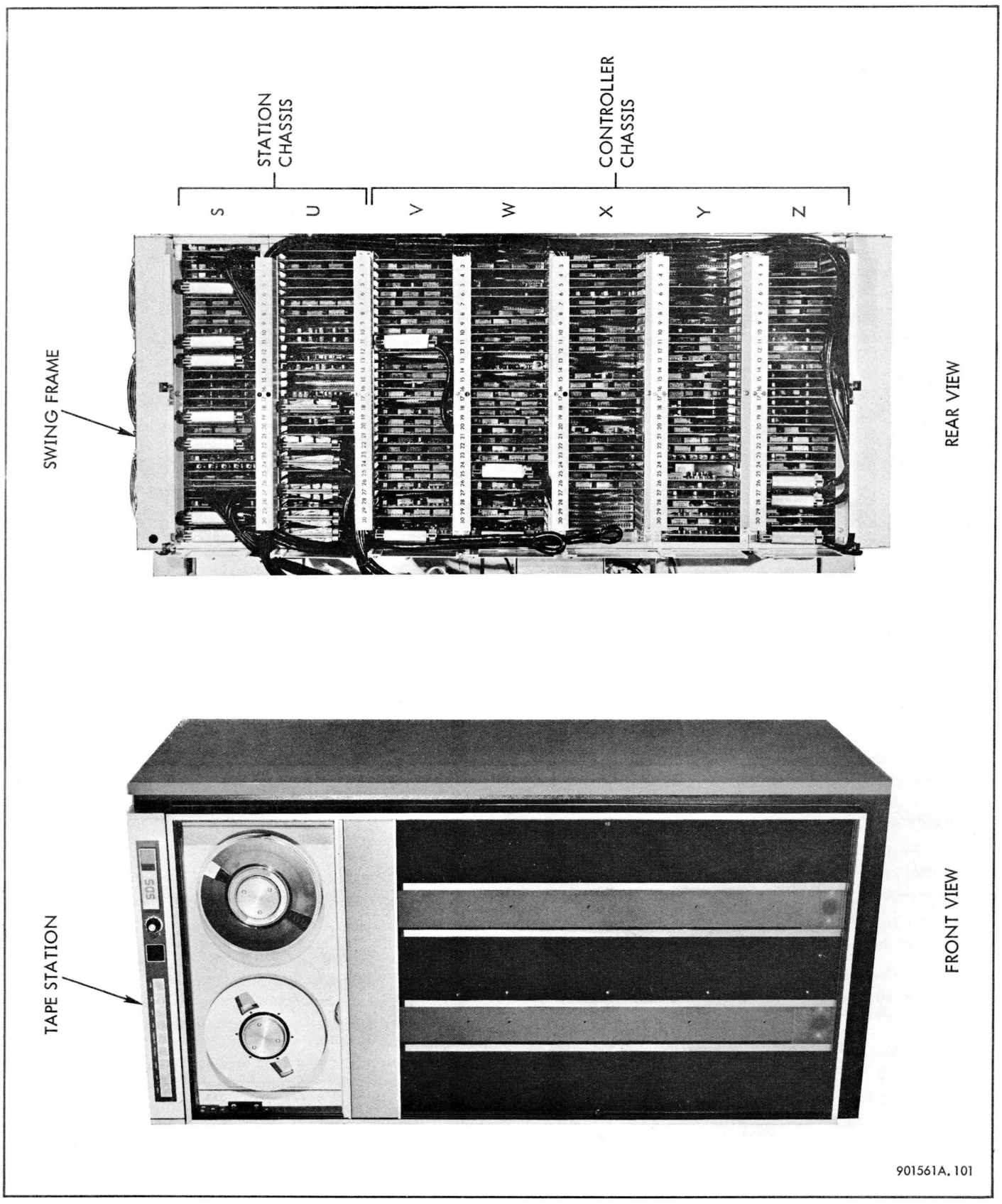


Figure 1-1. Magnetic Tape System Model 7320/7322 and Model 7320/7323

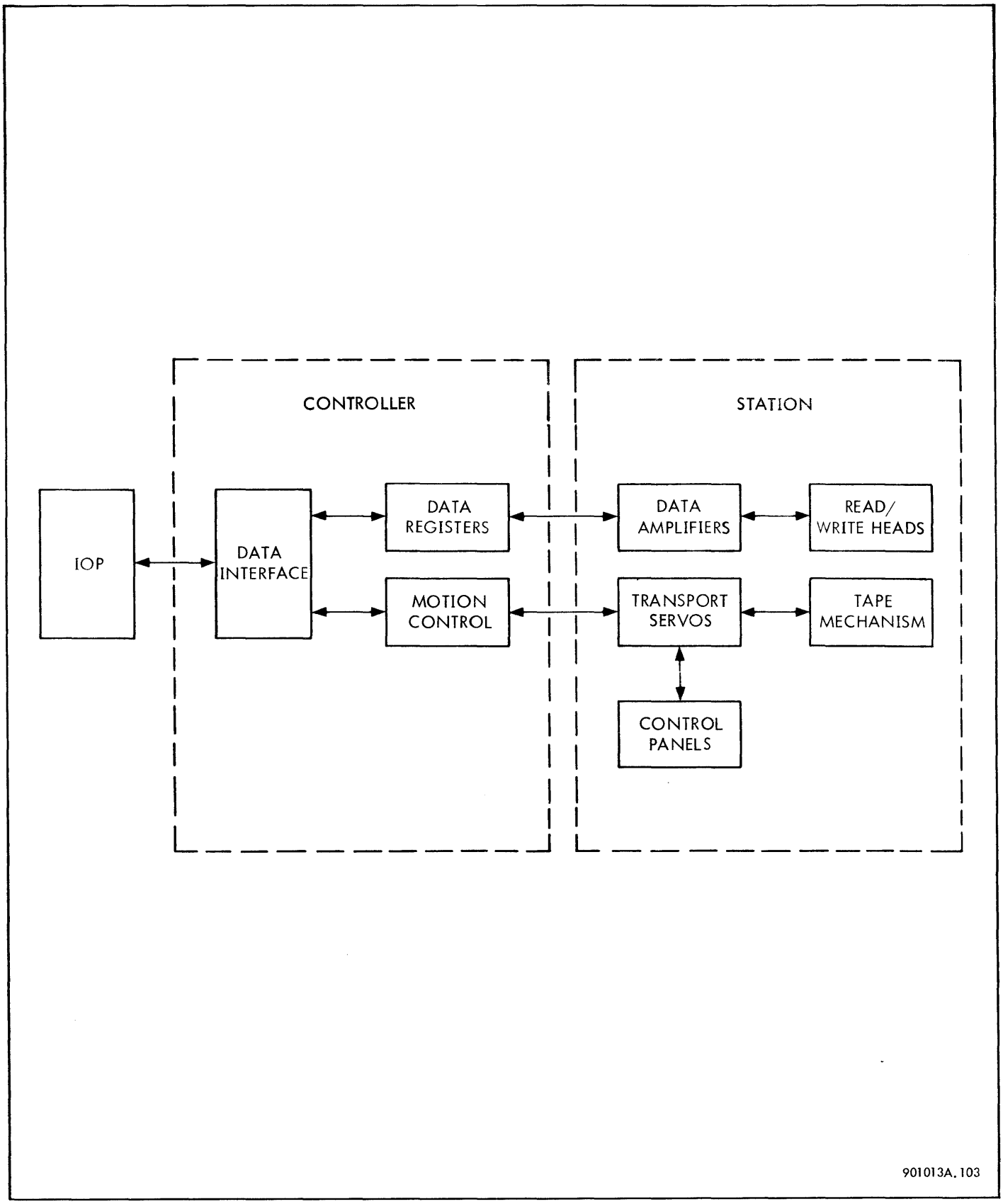


Figure 1-2. Magnetic Tape System, Block Diagram

SECTION II
OPERATION AND PROGRAMMING CONSIDERATIONS

2-1 INTRODUCTION

This section contains programming information for the Magnetic Tape Systems Models 7320/7322 and 7320/7323, and a description of the various functions and operations of the tape system. An explanation of the orders and the instruction responses necessary to perform the functions and operations is also given. The term device used in the description is SDS terminology for any type of peripheral equipment.

2-2 OPERATING CONTROLS, ADJUSTMENTS, AND INDICATORS

Operating controls, adjustments and indicators are located on the stations and are detailed in SDS publications No. 901084 and No. 901176.

2-3 OPERATING INSTRUCTIONS

When operating in automatic mode, the computer controls all tape system operations through the controller. When operating in manual mode, tape speed and direction are controlled from the auxiliary control panel on the station. For a detailed description of operating instructions, refer to SDS publications No. 901084 and No. 901176.

2-4 PROGRAMMING CONSIDERATIONS

The programming information in this manual is applicable to the 7320/7322 and 7320/7323 magnetic tape systems, used with the Sigma-series computers. The basic programming of writing and reading on magnetic tape is essentially the same as for any other combination of input and output devices that use interlace control. Since tape is also used as an external storage device, rather than strictly as an input/output device, certain other functions are unique to programming magnetic tape systems. These functions are described in detail in the following paragraphs.

2-5 SYSTEM CHARACTERISTICS

The main characteristics to be considered in programming the Tape Systems Model 7320/7322 and Model 7320/7323 are as follows:

a. Nine recording channels (eight data, one parity)

b. 75 ips (Model 7320/7322) and 150 ips (Model 7320/7323) tape speed

c. 800 bpi (60,000 and 120,000 characters per second) recording density

d. One byte transfer time = 16.7 μs and 8.35 μs

e. No minimum record length; 12 bytes minimum for IBM compatibility

f. Nominal interrecord gap = 0.6 in.

2-6 MAGNETIC TAPE SYSTEM I/O INSTRUCTIONS

The CPU communicates with the magnetic tape system through the use of the following five instructions:

Instruction Name	Mnemonic
Start Input/Output	SIO
Halt Input/Output	HIO
Test Input/Output	TIO
Test Device	TDV
Acknowledge I/O Interrupt	AIO

2-7 I/O Addressing

An eight-bit I/O address must be provided by the I/O processor to select a device for the SIO, HIO, TIO, and TDV instructions. For the AIO instruction, the device selected by the priority chain to process its interrupt presents its eight-bit address to the I/O processor. The assignment of an address for the controller is performed by setting up switches on a module in the subcontroller portion of the controller. The individual device number for each station is selected by a rotary switch located on the operator control panel of the station and is displayed by a segmented display indicator. The controller responds to each I/O instruction with a set of condition codes and with status information.

2-8 SIO Description

The CPU executes an SIO to initiate an input or an output operation with a device specified by the I/O address. At the same time, the controller that is addressed sets up the condition codes (CC1 and CC2) in the IOP for the CPU to examine, as follows:

CC1	CC2	Interpretation
0	0	I/O address recognized and SIO accepted (tape unit has advanced to busy condition)

CC1	CC2	Interpretation
0	1	I/O address recognized but SIO not accepted (tape unit was already busy or device interrupt is pending)
1	0	Tape unit is attached to a busy selector IOP. Not applicable to Sigma 2
1	1	I/O address not recognized

I/O address recognition indicates that the addressed device exists and is online. The controller accepts the SIO and indicates SIO successful only if it is not busy with a previous operation, if the station that is addressed is operational and ready, and if neither the controller nor the addressed station has an interrupt pending. (See paragraph 2-45.)

2-9 HIO Description

The addressed device immediately halts all current operations upon the execution of an HIO. The controller itself, if busy, returns to the ready state (or has its pending interrupt reset) only if the HIO is addressed to the active device that has a transmission interrupt pending in the controller. HIO is used only in special cases, and the tape positioning of a busy device halted in this manner is undefinable. The condition code settings for HIO are as follows:

CC1	CC2	Interpretation
0	0	I/O address recognized and tape unit not busy when halt occurred
0	1	I/O address recognized but tape unit busy when halt occurred
1	1	I/O address not recognized

2-10 TIO Description

The CPU uses TIO to gain access to the status information of the selected IOP, the controller, and the station. The operations of the IOP, controller, and station are not affected by the use of TIO. The condition codes for TIO are as follows:

CC1	CC2	Interpretation
0	0	I/O address recognized and SIO can currently be accepted (tape unit in ready condition with no device interrupt pending)
0	1	I/O address recognized but SIO cannot currently be accepted
1	0	Tape unit is attached to a busy selector IOP. Not applicable to Sigma 2
1	1	I/O address not recognized

2-11 Status Information for SIO, HIO, and TIO

Common status information is provided for the CPU whenever it executes an SIO, HIO, or TIO instruction. The controller and the station supply the information that is contained in bit positions 0 through 7 of the status response byte. See table 2-1.

2-12 TDV Description

The CPU uses TDV to obtain a more detailed status report from the selected device, and the execution of TDV does not alter the operation of the addressed IOP, the controller, or the station. The condition code settings for TDV are as follows:

CC1	CC2	Interpretation
0	0	I/O address recognized
1	0	Tape unit is attached to a busy selector IOP (not applicable to Sigma 2)
1	1	I/O address not recognized

2-13 TDV Status Information (See table 2-2.)

2-14 AIO Description

The CPU executes an AIO to acknowledge an I/O interrupt and to identify the source and the cause of the interrupt. The condition code settings for AIO are as follows:

CC1	CC2	Interpretation
0	0	Normal interrupt (channel end or zero byte count) condition present
0	1	Unusual interrupt (unusual end) condition present
1	1	No interrupt condition present

2-15 AIO Status Information (See table 2-3.)

2-16 INPUT/OUTPUT OPERATIONS

An input/output operation is initiated when the device accepts an SIO instruction from the CPU. The controller

Table 2-1. Status Response for SIO, HIO, and TIO

Bit Position	Bit Status	Description
0	1	Interrupt Pending. Either the addressed device has requested an interrupt, or the controller has set an interrupt, after having received a request from the IOP that was associated with a previous data transmission to the addressed device. An SIO is not possible while a controller interrupt is pending, but command chaining may occur
1, 2	0, 0	Device Ready. The selected station is operational and is not connected to the controller for an operation
	0, 1	Device Not Operational. The addressed device has developed some condition that does not allow it to proceed
	1, 0	Not used
	1, 1	Device Busy. The addressed station is operational and is currently connected to the controller for an operation or is rewinding
3	1	Automatic Mode. A successful SIO is possible. The device is under program control
	0	Manual Mode. A successful SIO is possible, but the device delays any order that requires tape movement until the operator sets it in the automatic mode
4	1	Unusual End. The controller has encountered an unusual condition since having accepted the last order (see paragraph 2-40)
5, 6	0, 0	Device Controller Ready. The addressed controller, if online, is in a standby state but may have an interrupt pending
	0, 1	Not used
	1, 0	Not used
	1, 1	Device Controller Busy. The addressed controller is online and is currently engaged in performing an operation
7		Not used

Table 2-2. Status Response Information for TDV

Bit Position	Bit Status	Description
0	1	Rate Error (Data Overrun). The addressed controller detected a data transfer rate error during the last read or write operation. This condition is caused by an equipment malfunction or by the total I/O data rate exceeding the system limits. When a rate error is encountered during writing, dummy characters are written, if necessary, until further bytes are available. At this time normal writing resumes
1	1	Write Permitted. Writing as well as reading may be performed

(Continued)

Table 2-2. Status Response Information for TDV (Cont.)

Bit Position	Bit Status	Description
1	0	Write Protected. The addressed device is write protected; only reading may be performed. Inserting a write enable ring on the tape file reel puts the device in a write permitted state
2	1	Write Protect Violation Error. The last order received is to write, and the device selected is write protected
3	1	End of File. Either the last record read or spaced was a tape mark record or the last reading or spacing has come to a halt at the load point (beginning of tape) marker
4	1	Incorrectable Read Error. The last record that was read resulted in a transmission error, and the error is of the type that cannot be corrected through a read with the correction process
5	1	Load Point. The addressed device is positioned at the load point marker
6	1	End of Tape. The selected device has the tape positioned past the end of tape marker. This signal is of significance during writing, because it gives a general idea of how much tape is left on the reel for further writing
7	1	Rewind Online. The addressed device is in the process of rewinding in the automatic mode

Table 2-3. Status Response Information for AIO

Bit Position	Bit Status	Description
0	1	Rate Error (Data Overrun). The addressed controller has detected a data transfer rate error during the most recent read or write operation. This condition is caused either by an equipment malfunction or when the total I/O data transfer rate exceeds the system limits. If a rate error is encountered during writing, dummy characters are written, if necessary until further bytes are available. At this time normal writing resumes
1	0	Controller Interrupt. An interrupt is generated in response to the IOP's request through a terminal order
	1	Device End Interrupt. An interrupt is generated when either device has just completed a rewind and an interrupt order execution or a device has just been put on the automatic mode by an operator after setting the ATTENTION indicator by pressing the control switches
2	1	Write Protect Violation Error. The last order received is to write, but the device selected is write protected

(Continued)

Table 2-3. Status Response Information for AIO (Cont.)

Bit Position	Bit Status	Description
3	1	End of File. Either the last record read or spaced was a tape mark record or the last reading or spacing has come to a halt at the load point marker
4	1	Noncorrectable Read Error. The last record read resulted in a transmission error, and the error is of the type that cannot be corrected through a read with the correction process

then proceeds to request an order from the IOP. The order byte is interpreted by the device as shown in table 2-4.

Table 2-4. Input/Output Order Byte

STATUS OF BIT POSITIONS								ORDER
0	1	2	3	4	5	6	7	
				0	1	0	0	Sense
				1	1	0	0	Read backward
				0		0	1	Write
				0		1	0	Read forward
	0	0	0			1	1	Set correction
	0	0	1	0		1	1	Rewind and interrupt
	0	1	0	0		1	1	Rewind offline
	0	1	1	0		1	1	Rewind
	1	0	0	D*		1	1	Space record
	1	0	1	D†		1	1	Space file
	1	1	0			1	1	Set erase
	1	1	1	0		1	1	Write tape mark

*Space record forward if D = 0; space record reverse if D = 1

†Space file forward if D = 0; space file reverse if D = 1

2-17 Hexadecimal Order Bytes

The hexadecimal order configuration for the magnetic tape system responses is as follows:

Order	Hexadecimal Configuration
Write	01
Read forward	02
Read backward	0C
Rewind	33
Rewind and interrupt	13
Rewind offline	23
Space record forward	43
Space record backward	4B
Space file forward	53
Space file backward	5B
Set erase	63
Write tape mark	73
Sense	04
Set correction	03

2-18 Order Descriptions

The operations associated with orders used for the magnetic tape system are described in paragraphs 2-19 through 2-24. Additional effects of particular orders are further outlined in the descriptions of the responses to I/O instructions and at the causes of the unusual condition status.

2-19 READ. The tape can be read either forward or reverse, as specified by the order. It is the programmer's responsibility to select the proper byte address for beginning data transfer. In Sigma 5 and 7 the read-reverse order transmits the bytes into memory in a descending sequence, exactly in

the reverse order to that of the read-forward. In Sigma 2, the read-reverse order transmits the bytes into memory in an ascending order, causing the record in memory to be reversed. The tape is started in the appropriate direction, and data is read until the interrecord gap is detected. This is also true for the tape mark record, although no data is transmitted in this instance.

2-20 WRITE. A normal data record, or a tape mark, is written when a write tape mark order is requested. Only forward motion is allowed for the write operation. Tape motion and data transfer terminates when the byte count reaches zero. For compatibility purposes, a minimum of 12 bytes per record should be recorded. Normal writing cannot take place if the tape unit is in the write protect mode. This condition is achieved by removing the write ring from the tape reel. The FILE PROTECT light on the operator control panel then lights, indicating to the operator that writing cannot take place.

2-21 SPACE. Four separate orders allow for implementing any type of spacing maneuver. The four orders are: space record forward, space record reverse, space file forward, and space file reverse. The last two orders traverse the tape over records until the tape mark of the record being spaced over is detected.

2-22 REWIND. Three separate rewind orders are available for different terminating functions. The rewind order effects a standard rewind to the load point, causing a device busy during the rewind. The rewind and interrupt order operates similarly, except that an I/O device end interrupt is requested when the tape is stored at the load point. The third rewind is similar to the first, with no interrupt, except that it switches the device to the manual mode at the start of the rewind.

2-23 ERASE. The only function of the erase order is to set an indicator in the device. When the next write (not a write tape mark) order is received, an erasure of 3-1/2 inches precedes the recording of data onto the tape. This particular order takes the least amount of time to be implemented; therefore, the programmer must be aware of the timing involved.

2-24 SENSE AND SET CORRECTION. This sequence of orders is used to correct data read from the tape after a read order that resulted in a transmission data error. After the read, if the read response is set but the noncorrectable read error response is reset, a correction can be effected.

A determination that the error is not correctable means that more than one channel has caused a parity error and that the sense-set correction procedure cannot be effective. For an explanation of this procedure, it is assumed that a record has been read that produced a read error response and that the record is correctable. Therefore, at least one byte in the record produced a parity error and only one channel caused that error. The sequence of orders needed for a correction would then be: sense, set correction, and read the same record again, either in the opposite direction or by spacing over it and repeating the same read. When the tape record is read for the second time, upon detecting a parity error, that channel bit in the error byte is reversed in polarity (since its reversal caused the parity error).

The operations associated with a typical SIO are shown in figure 2-1.

2-25 Order Execution

2-26 SENSE. A byte containing the track-in-error information of the preceding read operation is transferred by the device controller to the IOP.

2-27 READ FORWARD. The selected device moves the tape forward to the next interrecord gap, and the tape is allowed to coast to a stop if no additional order to move forward is issued. The information that is read is sent to the I/O channel until either the count done signal is received or an interrecord gap is detected. Reading over a tape mark sets up an unusual end condition, and no data is transmitted.

2-28 READ REVERSE. The selected device moves the tape backward to the next interrecord gap, and the tape is allowed to coast to a stop in the gap if no additional order to move backward is issued. The data bytes that are read are transmitted to the channel in the reverse order from which they were recorded. Data transmission ends when either the IOP signals count done or the device detects the interrecord gap. Reading over a tape mark or attempting to read backward at the BOT marker sets up an unusual end condition, and no data is transmitted.

2-29 WRITE. The selected device activates the erase head and moves the tape forward. After generating a gap, data received from the IOP is written sequentially on the tape until the IOP signals either count done or IOP error halt. The redundancy check characters are written after the last data byte. The read-after-write process checks the entire record currently being written for correct lateral and longitudinal parity, and, when it detects the gap at the the end of the record, the tape is allowed to coast to a stop. An attempt to write on a write-protected file results in an unusual end, and the write operation is aborted entirely.

2-30 SET CORRECTION. The track-in-error information byte (provided by the I/O channel) is transferred back to the controller. This order, if followed directly by a read order, enables the read with the read-with-correction feature to take place.

2-31 REWIND. The selected device rewinds tape to the load point if it is not resting at the load point when the order is issued. The controller ends the order execution when the device begins to rewind.

2-32 REWIND AND INTERRUPT. The execution of this order is identical to that of the rewind, except that the

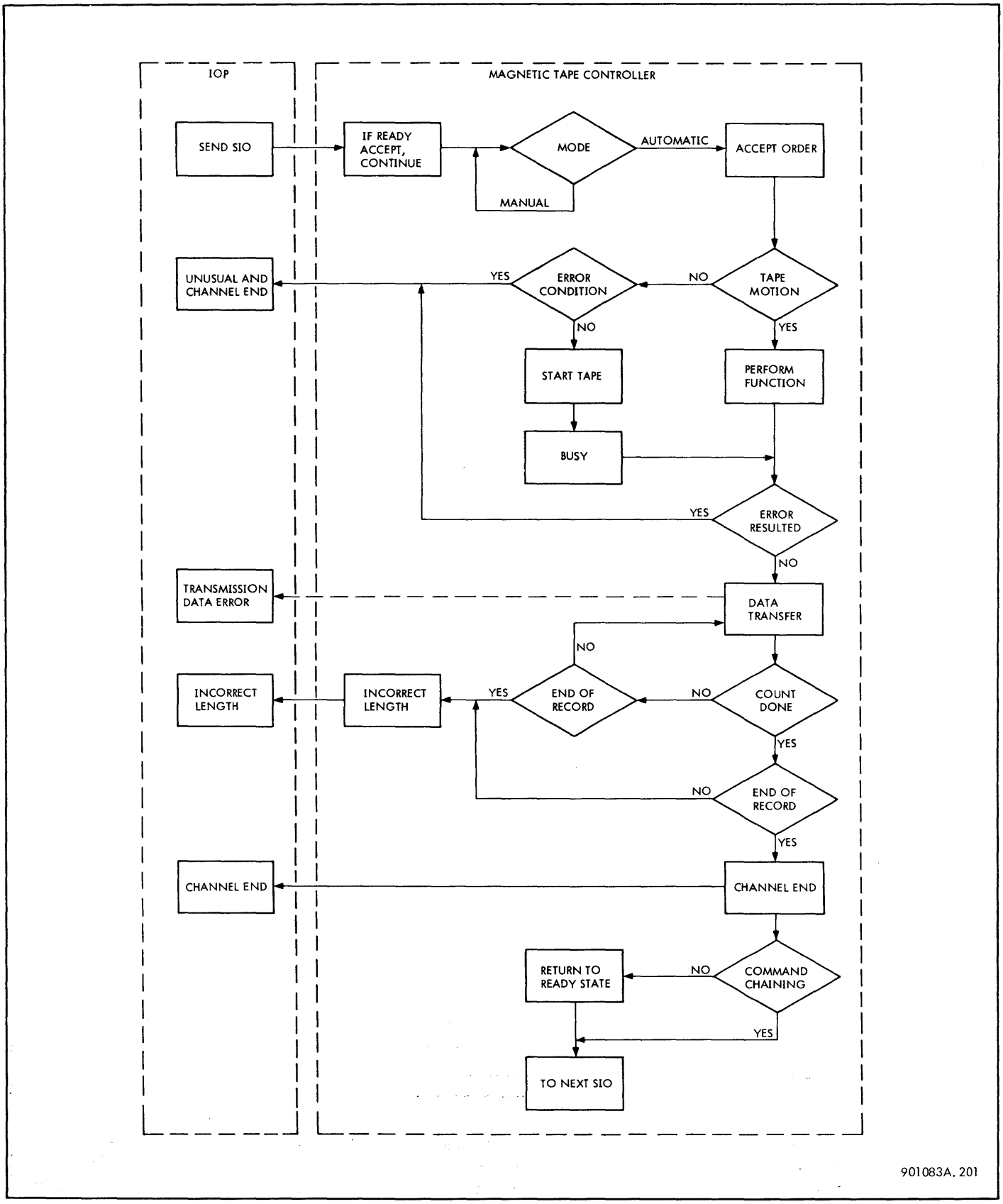


Figure 2-1. SIO Operation, Simplified Flow Diagram

lected device generates a device interrupt when the device becomes ready at the load point.

2-33 REWIND OFFLINE. The execution of this order is identical to that of rewind, except that the selected device switches itself to the manual mode when it starts rewinding.

2-34 SPACE RECORD FORWARD, SPACE RECORD REVERSE. The selected device moves the tape forward or backward to the next interrecord gap. No data transfer occurs. Spacing over a tape mark record or BOT marker sets the unusual end condition. The tape is not allowed to travel in the reverse direction beyond the BOT marker.

2-35 SPACE FILE FORWARD, SPACE FILE REVERSE. The selected device moves the tape forward or backward to the interrecord gap beyond the next tape mark record or to the load point. If a space file backward order is issued to a device which is already positioned at the load point, the controller sets the unusual end and aborts the order entirely.

2-36 SET ERASE. The selected device controller sets the erase flip-flop but no tape motion is initiated. If this order is followed directly by a write or a write tape mark order, an erasure of approximately 3-1/2 inches of tape takes place prior to the writing process. The set erase order is not necessary at the load point, since it is inserted automatically by the device.

2-37 WRITE TAPE MARK. The selected device turns on the erase head and moves the tape forward. After generating a gap, a tape mark is written. The read-after-write process checks the lateral and the longitudinal parities; when it detects the gap, the tape is allowed to coast to a stop. No data transfer occurs in this order execution. An attempt to write a tape mark on a write-protected device sets the unusual end, and the order is aborted entirely.

2-38 Termination of Order Execution and Error Indications

The controller, upon completing an order execution, signals the IOP by sending a channel end or an unusual end, or both, with error indications accumulated during the order execution. Error indicators are cumulative in the event of command chaining.

2-39 CHANNEL END. Channel end is reported at the end of each order execution except when the following conditions occur:

a. An order out service cycle terminates with an IOP error halt indication.

b. A read backward, a space record backward, or a space file backward order is issued to a device that is already positioned at the load point.

c. A write or write tape mark order is issued to a device that is write-protected.

2-40 UNUSUAL END. Unusual end is reported whenever the controller encounters one of the conditions noted in the previous paragraph, or when one of the following conditions occur:

a. The IOP signals an IOP error halt (except in the terminal order following order in).

b. A tape mark record is detected during a read operation.

c. A space record forward or a space record reverse order execution is given over a tape mark record.

d. The device switches to nonautomatic after the tape motion is initiated, except for rewind orders.

e. A reverse order occurs at the load point.

The device and the controller return to ready after reporting an unusual end or after the IOP halts in a terminal order following an order in.

2-41 TRANSMISSION ERROR. A transmission error report is made at the end of a read or a write operation if the controller has detected either the data transfer rate error or the data error, or both. The data error consists of lateral and longitudinal parity errors in both read and read-after-write; it also consists of the CRC mismatch error during read.

2-42 INCORRECT LENGTH. An incorrect length indication is issued after a read only, if either the count done is signalled before the complete record has been read, or the complete record is read without sensing count done.

2-43 I/O INTERRUPTS

Two types of interrupts may be generated by the tape system. One results from the interrupt bit in a terminal order, and the other originates at the device itself. The latter is termed a device interrupt, and the former is considered a controller interrupt uniquely associated with the last active device. Device interrupt is generated when the device is positioned at the load point and is ready to perform an operation either at the conclusion of a rewind and an interrupt order or at the time that the device is switched to the automatic mode after the ATTENTION button is pressed.

2-44 DATA REPRESENTATION

Records are written on tape in byte sequence. Four bytes can be transmitted from each word in memory from Sigma 5 and 7, and two bytes from Signa 2. Each byte contains the eight bits corresponding to the eight channels of data on tape and an odd-parity bit that is developed in the IOP and is written on tape as the ninth channel. Figure 2-2 illustrates the translation of memory information into data on tape.

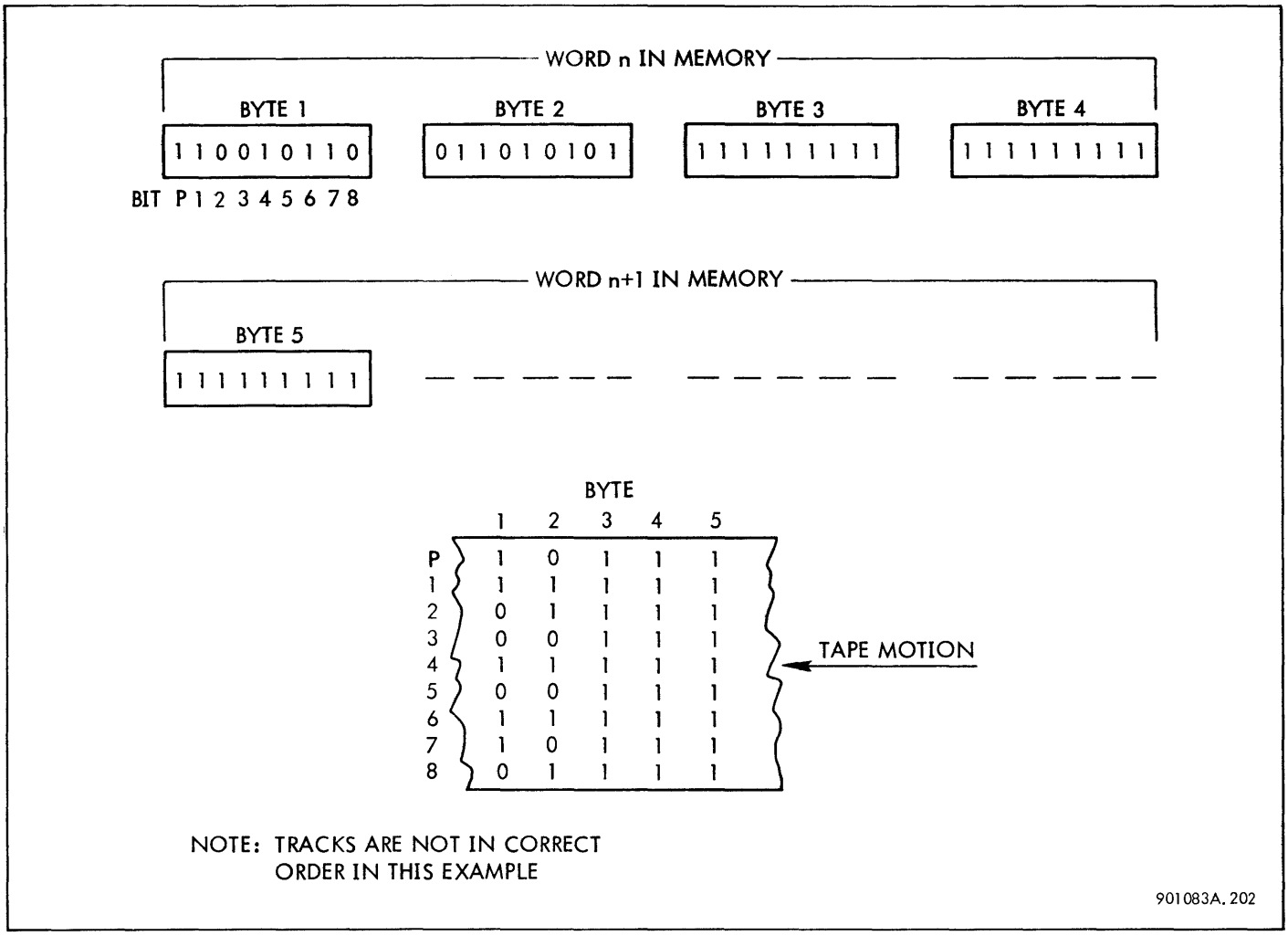


Figure 2-2. Data Translation, Memory to Tape

The write order reaches the byte address of the first word in memory from which data is to be transferred. The read forward order reaches this same byte address location. The read reverse order (Sigma 5 or 7) must reach the byte address plus the number of bytes to be transferred to have the data placed in memory in the same relative position as for the read forward order. In a read reverse order, the data is transmitted into memory in the reverse sequence to the read forward order.

2-45 MAGNETIC TAPE SYSTEM STATES

Several status levels describe the various states that the tape system can be in. These states comprise a priority chain that must be satisfied before the device can operate in conjunction with the IOP. The states are as follows:

a. Device address recognizable

b. Device operational

c. Device automatic

d. Device ready

e. Device controller ready

The SIO, HIO, and TIO instructions provide the current status of the addressed device and of its controller. Following is a brief explanation of the possible conditions and the modes of the tape system.

2-46 Device Address Recognized

This condition exists unless the device does not have its power on or unless the UNIT SELECT switch on the tape station console is not set to the correct address.

2-47 Device Operational

The device is operational if all vacuum and interlock requirements are satisfied. A nonoperational condition exists if the vacuum falls too low, if the window or door is opened, or if the tape either goes off the reel or breaks.

2-48 Device Automatic

This condition exists when the operator presses the START switch on the operator control panel after ensuring that all other conditions exist for a successful tape operation. The manual mode results when the operator presses the RESET switch on the operator control panel or when a nonoperational condition arises. The rewind offline order switches the device to the manual mode. The controller, if ready, can accept an SIO for this device even if it is in the manual mode, but it waits until the device becomes automatic. This situation forces the controller into a busy condition.

2-49 Device Ready

The device is considered ready if it is in a condition to accept and to operate an SIO instruction. The device must be operational and not busy. It may be either in the manual or in the automatic mode and must still be considered ready.

2-50 Device Controller Ready

The device controller has only two states, ready and busy. It becomes busy when it accepts an SIO, and remains so until released in the order in state. It then becomes ready again and can accept another SIO when conditions allow. Figure 2-3 is a simplified block diagram of the ready and the busy states in online (automatic) and offline (manual) operation.

2-51 DATA TRANSFER MODE

Only one mode of data transfer is used with the magnetic tape system. This mode is a direct memory to tape data transfer discussed in paragraph 2-44. The command doublewords for the read and the write orders specify the byte address in memory as well as the byte number to be transferred. An incompleted word can be transferred as the last word in the record. The IOP obtains and transfers the data into and out of memory. For a write order, data is transmitted from memory onto tape until the byte count is decreased to zero. The proper tape gap is developed before and after the recording of data. Channel end is sent when count done is encountered in the write order. For read orders, the data transfer terminates when either the count done is received or the end-of-record gap is sensed on the tape.

An incorrect length response results if both the count done and the end-of-record indications are not sensed together at the termination of the order. Parity checking takes place during the data transfer; that is, in the data read in for read orders, and in the read-after-write sequence for the write orders (including the write tape mark order). A resulting parity error sets the transmission data error response.

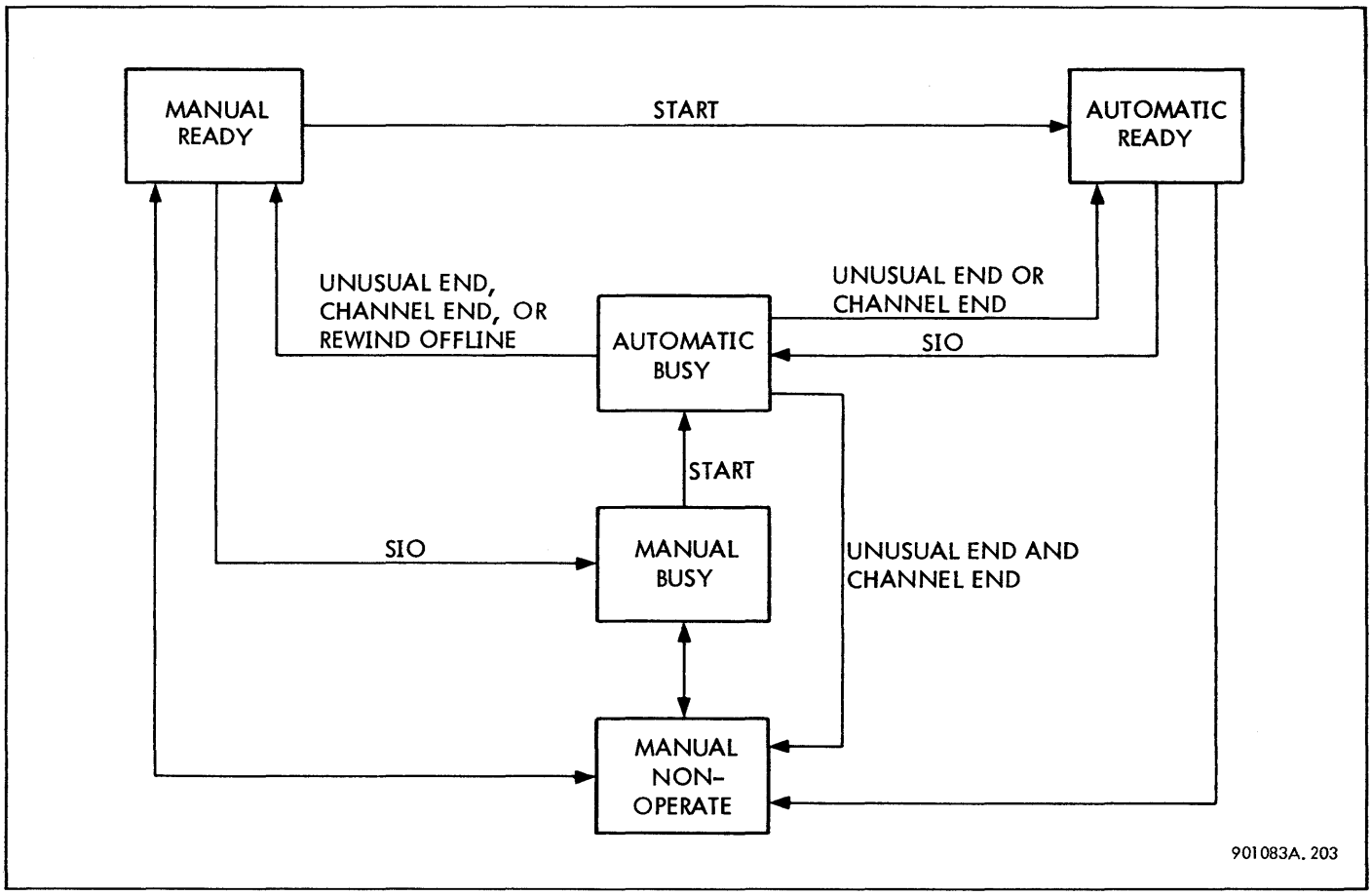


Figure 2-3. Magnetic Tape Unit Ready and Busy States, Simplified Block Diagram

SECTION III
FUNCTIONAL DESCRIPTION

3-1 INTRODUCTION

This section contains a general description of the magnetic tape system and of the tape data format, and a functional description, on a block diagram level, of the various controller functions.

3-2 GENERAL DESCRIPTION

3-3 MAGNETIC TAPE SYSTEM

The magnetic tape system communicates with the Sigma computer through the input/output processor (IOP) that controls its operation. The magnetic tape system performs the following operations:

a. Read in the forward or reverse direction. When the tape system receives a read order from the IOP, it initiates either forward or reverse tape motion as determined by the program. One byte (eight data bits plus a parity bit) is transmitted to the computer for every character that is read. Reading continues until the interrecord gap is encountered. The redundancy check characters (encountered either first or last, depending upon the direction of tape motion) are not transmitted to the computer but are checked by the controller. If no further orders are received, after the entire record has been read (including the redundancy check characters), the tape comes to a stop with the read heads located in the interrecord gap.

b. Write in the forward direction. When the tape unit receives a write order, it initiates a forward tape motion (writing is not permitted in the reverse direction). One nine-bit character (eight data bits plus a parity bit) is written for each byte that is received from the computer. Writing continues until the desired number of characters, as determined by the program, has been recorded. The tape station then receives both the cyclic and the longitudinal redundancy check characters for recording. The validity of the recorded information is guaranteed since a read after write operation is performed on each recorded character. The longitudinal redundancy check character is also read and is compared in the controller with the information in the record just written. The tape coasts to a stop, if no further orders are received after the tape station has read the longitudinal check character.

c. Space over a record in the forward or reverse direction. When the tape system receives a space record order, it spaces forward or reverse over one record, as determined by the program. If no further orders are received, the tape comes to a stop with the heads located in the gap following the record.

d. Space over the file in a forward or a reverse direction. When the tape system receives a space file order, it spaces forward or reverse, as determined by the program, until a tape mark is encountered. The tape then stops with the tape heads located in the gap following the tape mark. If no tape mark is encountered in the reverse direction, tape motion ceases when the load point is encountered. The controller detects the tape mark.

e. Erase on writing. Upon receipt of an erase order, the tape system sets an erase condition that causes approximately 3-1/2 inches of tape to be erased preceding the next write operation.

f. Rewind either online or offline. When the tape station receives a rewind order, it moves the tape in the reverse direction. The tape comes to rest when the load point is encountered. The tape system can be ordered to generate an interrupt when the load point is reached or to switch to the manual mode after rewind is started. The tape system then requires operator intervention to be switched back online. This order is used when the operator has finished processing one reel of information and wants to change that reel.

3-4 TAPE DATA FORMAT (See figure 3-1.)

The data format conforms to the IBM 2400 nine-track data format. Information is recorded on nine parallel tracks along the length of the tape at a recording density of 800 bytes per inch, with odd parity checking. The information is in bytes, and each byte consists of eight data bits and one parity bit. A column of bits is formed across the width of the tape. The nonreturn-to-zero (NRZ) method is used to record the bits on the tape. By this method, a change in the direction of the magnetic flux on the tape occurs when a one bit is recorded. No change in flux direction takes place when a zero bit is recorded.

The information on tape is arranged in records. Each record may contain any number of bytes, limited only by the length of the usable portion of the tape. Records are separated on tape by an interrecord gap of approximately 0.6 inch in length, which is automatically generated during a write operation. An initial gap of approximately 0.94 inch in length is inserted between the load point marker and the first record when the writing process is initiated. At the end of each record, three blank characters are written, followed by a cyclic redundancy check (CRC) character. Three more blank characters are then written, followed by a longitudinal redundancy check (LRC) character.

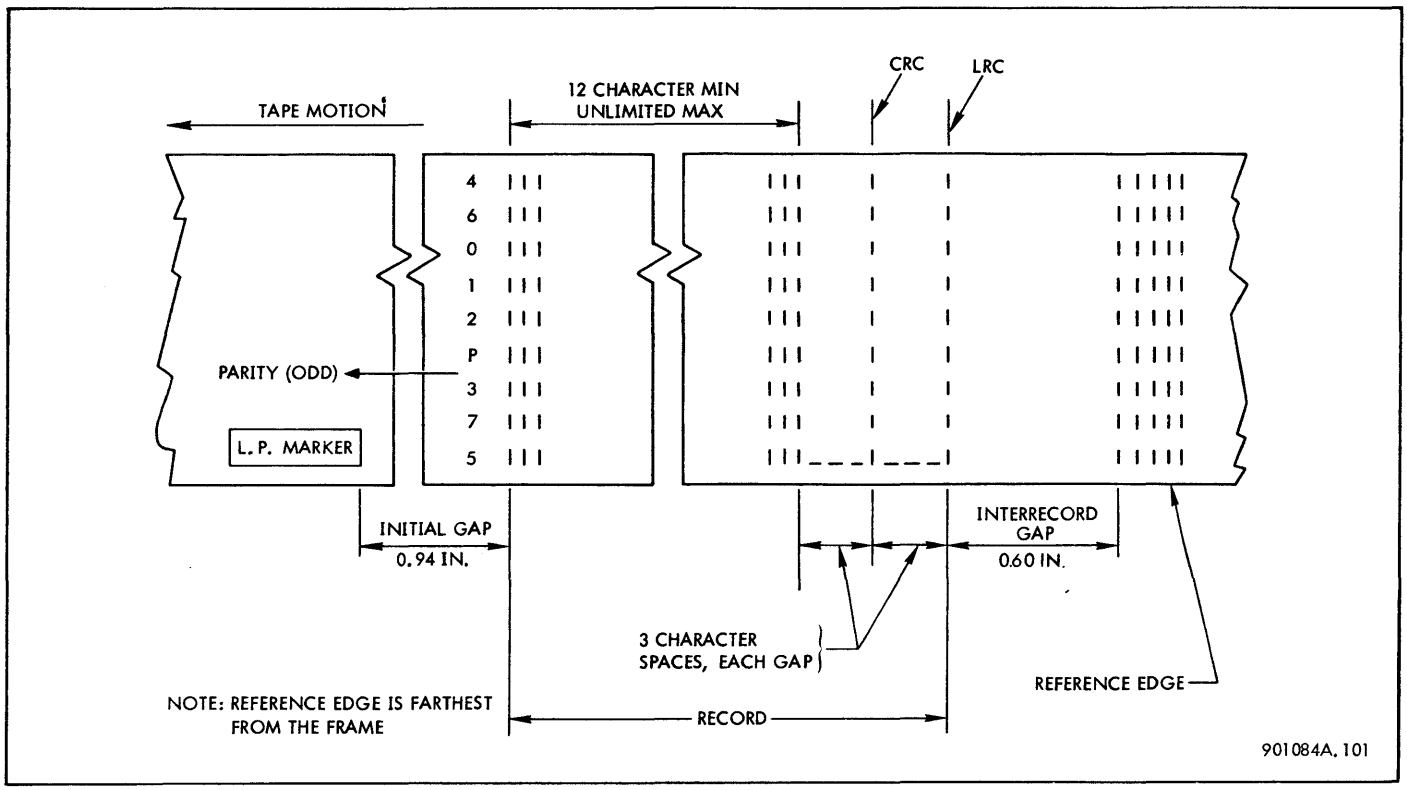


Figure 3-1. Tape Data Format

3-5 Tape Mark Record

A file of information consists of a group of one or more consecutive records. Adjacent information files are separated by a tape mark record. A tape mark record is a special single byte record that consists of a tape mark character, equivalent to a hexadecimal 13 (00010011), followed by seven blank characters and an LRC character. A number of files may be written on a reel of tape. The tape mark character following the last file on the tape also serves to indicate the end of recorded information on the tape reel.

3-6 Checking Record Validity

During the write operation, information on the tape is immediately read back for checking. This process is known as a read-after-write check and is accomplished through the use of a two-gap read write head. The information on the tape is checked during the read-after-write operation and also during a normal read process.

3-7 BYTE PARITY CHECK. During a read forward, or a read backward, or a read-after-write process, each column of bits across the tape that forms a byte is checked for odd parity (with the exception of the CRC and the LRC characters). Whenever an error is found, the error pattern register is modified for later use.

3-8 CYCLIC REDUNDANCY CHECK (CRC). The CRC character is initially generated from the input data and is recorded during the write operation. The CRC character is checked only during the read forward or read backward operations, and not during read-after-write. The CRC character is checked in the following manner: During the read forward or the read backward operation, the CRC character is generated again from the data that is being read. This CRC character is then compared with the CRC character that was generated and is written on the tape in the write operation. If the two characters do not agree, a track-in-error determination is attempted with the aid of the error pattern register. If the track-in-error is determinable, indicating that all errors occurred in a single track, the error is termed a correctable read error.

3-9 LONGITUDINAL REDUNDANCY CHECK (LRC). The parity of each track (or channel), up to and including the CRC character, is automatically checked during the write operation. This data is used to determine the LRC character in order to make even the parity of each track over the entire record. During a read forward, a read backward, or a read-after-write operation, an LRC character is generated from the data that is being read. This LRC character is then compared to the LRC character that was written on the tape in the write process.

3-10 Tape Markers

Tape markers (reflective strips) are placed on the tape in conformance with IBM tape specifications to allow the tape system to sense the beginning (load point) and the end (end

Wie? (Bildungsgesetz)

of tape) of the usable portion of the tape. At least 10 feet of tape must be allowed between the beginning of the reel and the load point marker. This portion of the tape is used as a leader.

The tape is loaded by manually threading the leader and by pressing the LOAD switch while the tape transport is in the manual mode. The loading cycle automatically positions the tape at the load point.

After the tape has been moved forward by reading, writing, or spacing, any rewind operation repositions the tape on the load point. Unloading the tape is accomplished by manually unwinding the leader after the tape is positioned at the load point.

Approximately 14 feet of tape, including 10 feet of leader, is reserved between the end of tape marker and the end of the reel of tape. The end of tape sensor provides a status signal that is observed by the program while writing.

3-11 File Protect Device

A file protection device is provided to avoid an accidental erasure of previously recorded information. This device consists of a plastic (write enable) ring that must be in the circular groove in the transport side of the reel to enable writing on the tape in that reel. When the write enable ring is not in the reel, writing is inhibited, and the file is protected against accidental writing.

3-12 CONTROLLER (See figure 3-2.)

The controller handles data transfer to and from the CPU through the IOP and controls tape motion by directing the proper operating orders to the station(s). The three main functions of the controller are: interface, data, and motion control. The three functions occur within eight conceptual states determined by the state counter. Figure 3-3 is a simplified block diagram of the controller states, and paragraphs 3-13 through 3-21 define them.

The interface function is described in paragraphs 3-22 through 3-29 and data transfer in paragraphs 3-30 through 3-62. The orders concerned with motion control are covered in paragraphs 2-18 through 2-37.

3-13 States

3-14 00F00U. The controller enters the 00F00U (idle or ready) state when it is initially energized. It remains in this state until it accepts an SIO from the IOP. The SIO is given to the controller, and it advances to state 01F01U.

3-15 01F01U. In state 01F01U, the controller requests service from the IOP by raising signal CSL. The controller is connected for service, and the order for the function to be performed is given to the controller. After the order is received, the controller advances to state 01F02U.

3-16 01F02U. The controller enters one of three phases from state 01F02U. These three phases are:

a. Data input phase for sense

b. Data output phase for set correction

c. Device select phase

In the sense phase, the set correction process is initiated and the track-in-error byte is transferred from the controller to the IOP. The controller then advances to state 01F00U (order input).

In the set correction phase, the track-in-error byte is transferred from the IOP to the controller; then the controller advances to the order input state, 01F00U.

In the device select phase, the controller separates valid and invalid orders and connects the device for operation. The controller then advances to state 01F03U (wait for device proceed).

3-17 01F03U. The 01F03U state is used to either initiate a rewind order or advance to the next state for a read, write, or space function. If a rewind order is initiated, the controller advances to state 01F00U (order input).

If a rewind order is not initiated, the controller advances to state 03F03U-02U (data transfer).

3-18 03F03U-02U. The 03F03U-02U state is used for three functions: read, space file or space record, and write.

3-19 03F01U. The read, write, and space record or space file orders are terminated in state 03F01U. The read and the write orders are held true by the controller when it advances to the next state, 03F00U (station finalization). If errors are detected, an indication of this condition is also carried to the next state.

3-20 03F00U. In this state, the read and the write orders are either carried through to the next state, 01F00U (order input), or the operation is terminated on an error indication.

3-21 01F00U. The 01F00U state is the order input state. The controller completes an operation in this state and sends an operational status report to the IOP. The status report is followed by a terminal order.

A command chaining order sends the read and the write orders to the 01F01U (order output) state. If there is no command chaining order, the controller advances to state 00F00U (idle or ready) and waits for the next order.

3-22 Interface

The IOP time-shares interface signals with all device controllers, thereby allowing them to operate simultaneously. (The terms device controller and controller are

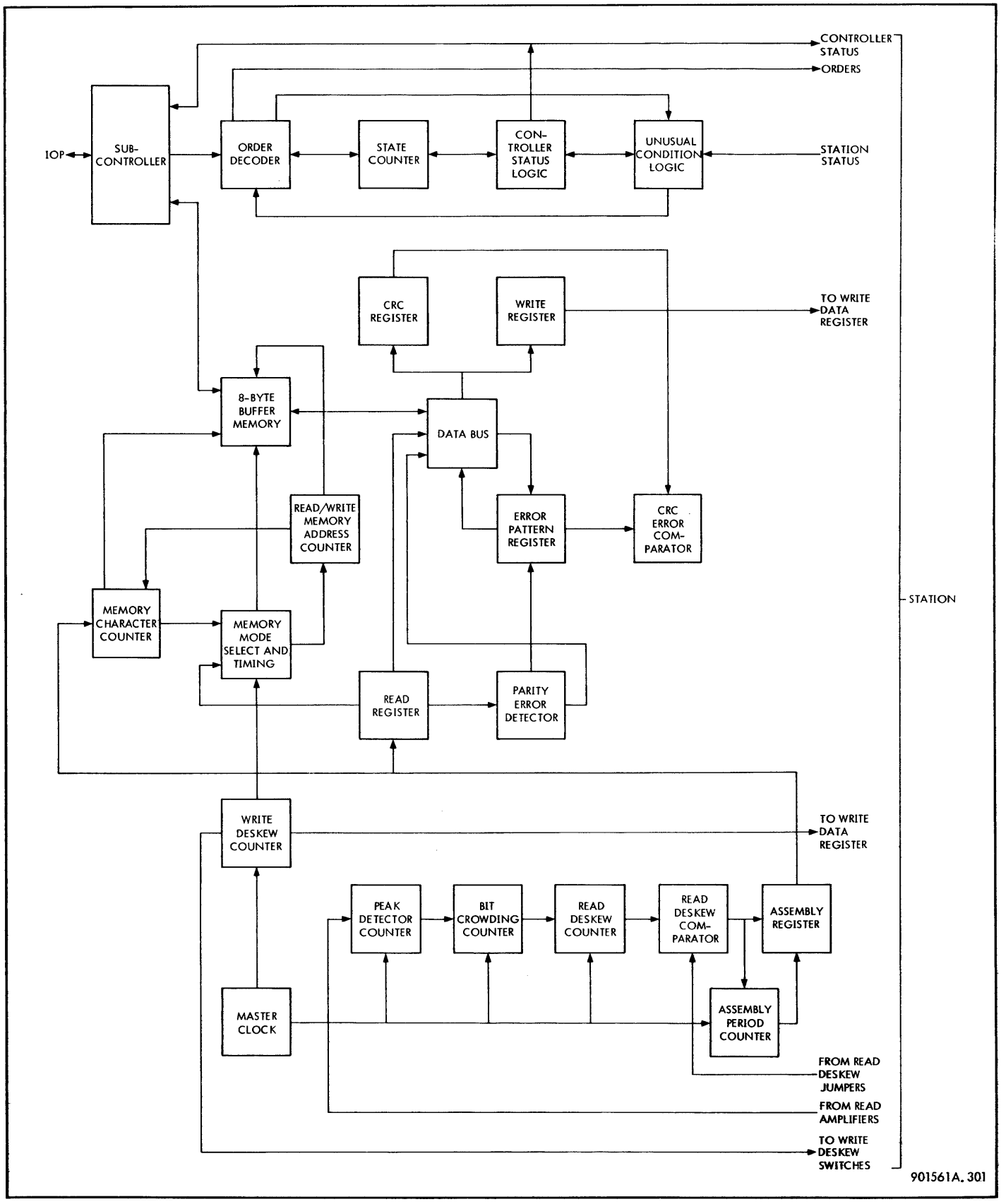


Figure 3-2. Controller, Functional Diagram

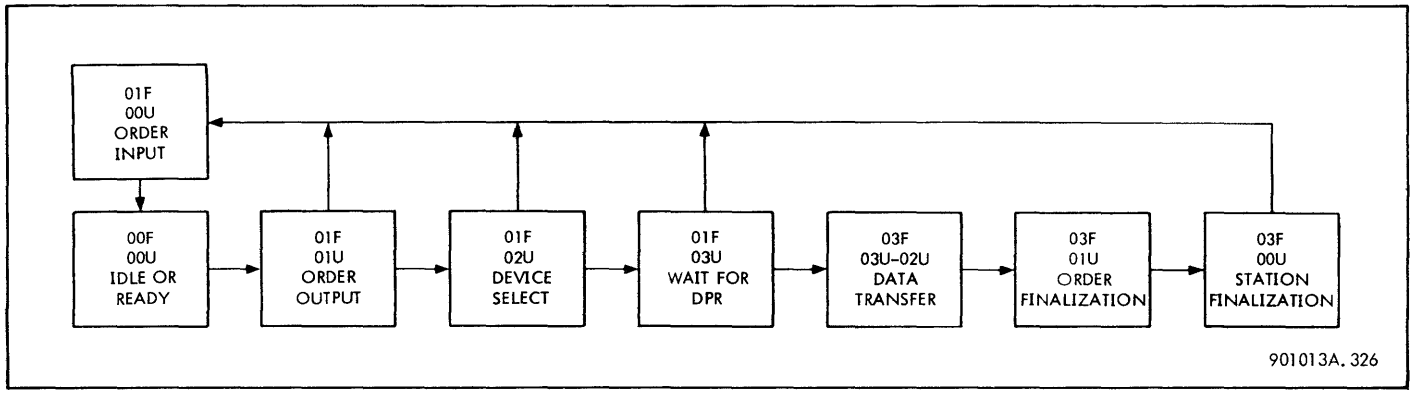


Figure 3-3. Magnetic Tape Controller States, Simplified Block Diagram

synonymous.) The interface signals are handled by the subcontroller portion of the controller. Figure 3-4 is a simplified block diagram of the subcontroller.

3-23 SUBCONTROLLER. The subcontroller contains the following logic and components:

a. Cable receivers and drivers to interface the eight-bit data path from the IOP.

b. Logic to determine priority during ASC and AIO operations.

c. Eight switches to select the controller address, and logic to compare the address with IOP output address during SIO, HIO, TIO, and TDV instructions.

d. Service connect flip-flop (FSC) and logic to indicate that the controller is connected for service.

e. Relay logic, under remote control, for connection or removal of the subcontroller during system on-off operations.

The subcontroller interface lines and their status for various system functions are shown in table 3-1.

The interface signals for the complete tape system are listed in table 3-2. Included in this list are the signals for IOP to controller, controller to station, and station to transport. Signal direction is indicated by the arrows. Most terms have the same name on both sides of the interface, but both names are shown where a change occurs.

3-24 COMMUNICATION AND SERVICE CYCLES. The interface signals travel in both directions between the IOP and the controller. Signals from controller to IOP initiate communication cycles, and those from IOP to controller initiate service cycles. A communication and service cycle sequence is shown in figure 3-5.

The IOP raises different function lines during a communication cycle from instructions executed by the CPU. The following functions are involved:

a. Start or halt each station under program control (SIO and HIO).

b. Status testing of controller and/or of station (TIO and TDV).

c. Interrupt acknowledgement (AIO).

d. Transfer of orders or data from or to the IOP (ASC).

3-25 SIO, HIO, TIO, and TDV Communication Cycles. The SIO, HIO, TIO, and TDV communication cycles are similar, and the SIO cycle is shown in figure 3-5.

3-26 Operational Sequence After SIO Instruction. The operations that take place during a communication and service cycle sequence are described in the following paragraphs.

Once a device controller has been activated by an SIO instruction, it may make service calls to the IOP until it halts or is halted by a halt (HIO) instruction. A device controller makes a service call by raising a specified line to the IOP, and it may do this at any time without regard to the state of the IOP. For example, the IOP may be occupied communicating with the CPU, the main memory, or some other device controller. A single line is used by all the device controllers to make service calls.

As soon as the IOP is free, it scans the service call line until it detects the service call request. The IOP does not know which controller is requesting service, since any number of controllers could be requesting service simultaneously. The IOP then issues a function strobe (FS) signal with a function indicator, acknowledge service call (ASC), which is sent to all controllers attached to the IOP.

A separate, hard-wired priority chain is connected from controller to controller. This chain is especially important when more than one controller is calling for service. The highest priority controller that has requested service places its device address on the eight function response lines and acknowledges the FS signal. All other controllers must then stay off

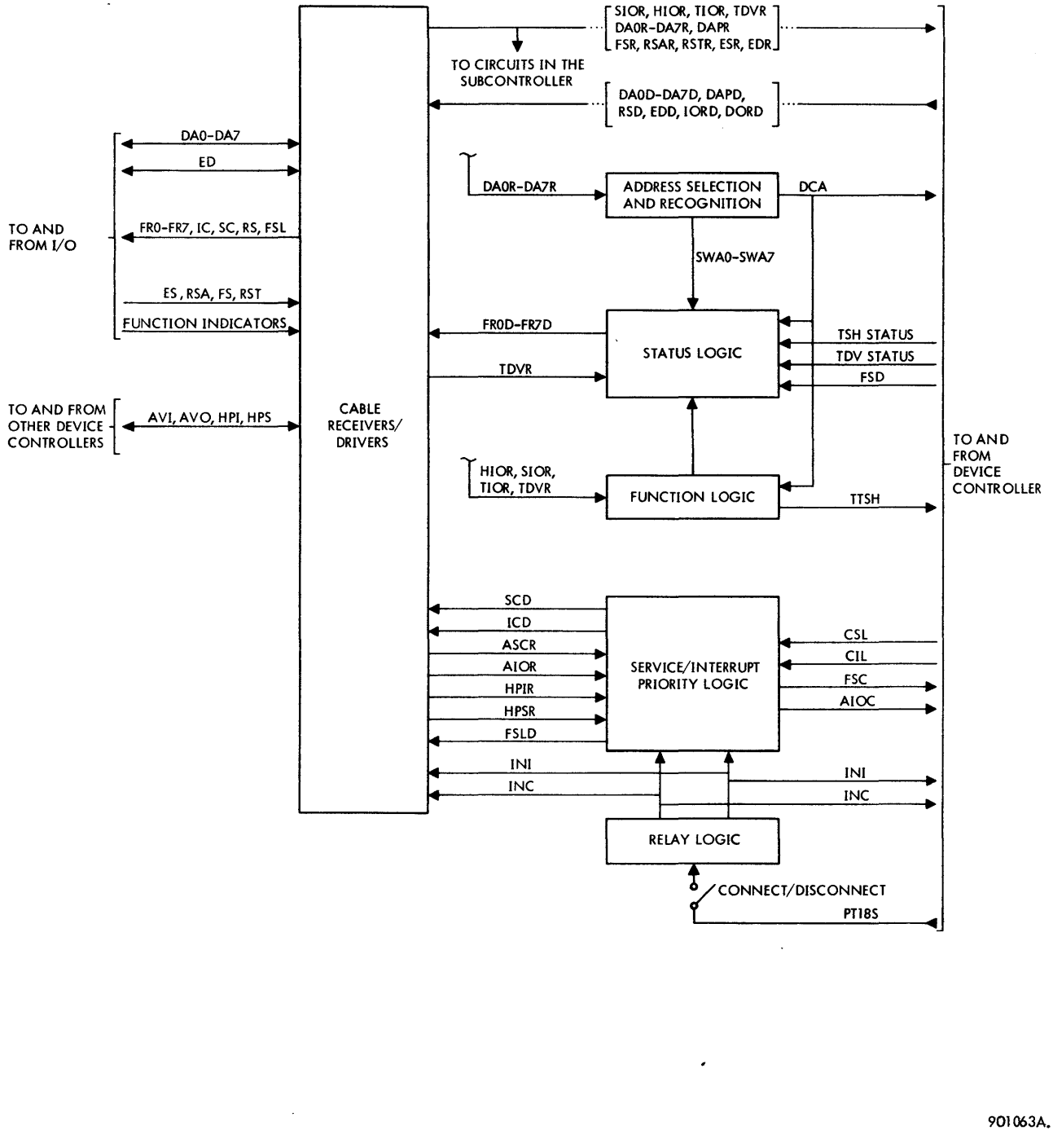


Figure 3-4. Subcontroller, Simplified Block Diagram

Table 3-1. Subcontroller Interface (Continued)

FUNCTION	DATA LINES								FUNCTION RESPONSE LINES								I/O ORDERS						DESCRIPTION
	DA0	DA1	DA2	DA3	DA4	DA5	DA6	DA7	FR0	FR1	FR2	FR3	FR4	FR5	FR6	FR7	DOR	IOR	DAP	PCD	ED	ESR	
Service (ASC)													Device Number				1						Order (0 ⇒ data)
																		1					Output (0 ⇒ input)
																					1		End data
					0	1	0	0									Order Out						Sense
					1	1	0	0															Read backward
					0		0	1															Write
					0		1	0															Read forward
		0	0	0			1	1															Set correction (to be used with next read)
		0	0	1	0		1	1															Rewind and interrupt (online)
		0	1	0	0		1	1															Rewind offline (manual mode)
		0	1	1	0		1	1															Rewind (rewinds to load point on automatic)
		1	0	0	D		1	1															Space record (D = 0 ⇒ forward; D = 1 ⇒ backward)
		1	0	1	D		1	1															Space file (D = 0 ⇒ forward; D = 1 ⇒ backward)
		1	1	0			1	1															Set erase (to be used with next write)
		1	1	1	0		1	1															Write tape mark
	1																Order In						Transmission error
		1																					Incorrect length
				1																			Channel end
					1																		Unusual end
				1													Order Out (Terminal)						IOP error halt

Table 3-1. Subcontroller Interface (Cont.)

FUNCTION	DATA LINES								FUNCTION RESPONSE LINES								I/O ORDERS						DESCRIPTION
	DA0	DA1	DA2	DA3	DA4	DA5	DA6	DA7	FR0	FR1	FR2	FR3	FR4	FR5	FR6	FR7	DOR	IOR	DAP	PCD	ED	ESR	
Service (ASC)														Device Number					1				Data parity
																				1			Check parity
																						1	End of service
SIO, HIO, TIO (Continued)		DCA				Device Number			1														Interrupt pending
										0	0												Device ready
										0	1												Device not operational
										1	0												Device unavailable – not used in mag tape system
										1	1												Device busy
												1											Automatic mode
													1										Unusual end
														0	0								Device controller ready
														0	1								Device controller not operational; not used in mag tape system
														1	0								Device controller unavailable; not used in mag tape system
														1	1								Device controller busy
																	1						Address recognition
																		1					System ready: SIO successful; HIO when device not busy; TIO when SIO can be accepted
TDV		DCA				Device Number			1														Rate error
																	1						Address recognition
																		1					Abnormal condition does not exist (device operational)
										1													Write permitted
											1												Write protect violation

Table 3-1. Subcontroller Interface (Cont.)

FUNCTION	DATA LINES								FUNCTION RESPONSE LINES								I/O ORDERS						DESCRIPTION
	DA0	DA1	DA2	DA3	DA4	DA5	DA6	DA7	FR0	FR1	FR2	FR3	FR4	FR5	FR6	FR7	DOR	IOR	DAP	PCD	ED	ESR	
TDV			DCA			Device Number						1											End of file
													1										Noncorrectable read error
														1									Load point
															1								End of tape
																1							Rewind online
AIO		1																					Device end
			1														1						AIO recognition
				1														1					No unusual end condition detected
					1																		Write protect violation
	1																						End of file
																							Noncorrectable read error
																							Rate error

Table 3-2. Magnetic Tape System Interface Signals

Transport	Station	Controller	IOP
Indicator, Device Ready	←DRDS		
Indicator, Device Busy	←DBSS		
Indicator, Attention	←ATNS		
Reverse/Forward	←REVS		
Actuate Motor	←ACMS		
Rewind/Offline	←ROFS		
Fast/Normal	←REWS		
Unit Select Switch 0C→	UNIT SELECT SW 0C		
1C→	1C		
2C→	2C		
3C→	3C		
4C→	4C		
5C→	5C		
6C→	6C		
Unit Select Switch 7C→	UNIT SELECT SW 7C		
ATTENTION CONTR→	ATTENTION CONTR		
START CONTR→	START CONTR		
OPERATIONAL CONTR→	OPERATIONAL CONTR		
FILE PROTECT→	FILE PROTECT		
EOT CONTR→	EOT CONTR		
LOAD POINT CONTR→	LOAD POINT CONTR		
LSPS→	LSPS		
HSPS→	HSPS		
25VWP1→	25VWP1		
25VWP2→	25VWP2		
POWER FAILURE IN*→	POWER FAILURE IN*		
POWER FAILURE OUT*→	POWER FAILURE OUT*		
	DV0R	←DV0D	
	⋮	⋮	
	DV7R	←DV7D	
	GNT	←GNT	
	RST	←RST	
	ATO, HLTD	←ATO, HLTD	
	AIOCD	←AIOCD	
	NTSH	←NTSH	
	CLK	←CLK	

*Power failure in and power failure out lines which short out through a relay contact when the power is applied to the transport are utilized only if the station and the controller occupy a common cabinet

(Continued)

Table 3-2. Magnetic Tape System Interface Signals (Cont.)

Transport	Station	Controller	IOP
	RVRS	← REV	
	WN1	← WN1	
	WN2	← WN2	
	WRT	← WRT	
	RASS	← FRS	
	DSS	← DSS	
	DST	← DST	
	DSL	← DSL	
	END	← END	
	DV0D →	DV0R	
	⋮	⋮	
	DV7D →	DV7R	
	INPS →	INPS	
	DRDS →	DRDS	
	DBSS →	DBSS	
	AUTS →	AUTS	
	ARGS →	ARGS	
	WPMS →	WPMS	
	BTSS →	BTSS	
	ENTS →	ENTS	
	WNDS →	WNDS	
	LSPS →	LSPS	
	HSPS →	HSPS	
	BOTS →	BOTS	
	INTS →	INTS	
	DSGS →	DSGS	
	DPRS →	DPRS	
		WDC45C	← WDC45C
		SRIPA	← SRIPA
		RDP	← RDP
		RPB	← RPB
		R0B	← R0B
		⋮	⋮
		R7B	← R7B
		TM	← TM
		BMT	← BMT
		EDB	← EDB

(Continued)

Table 3-2. Magnetic Tape System Interface Signals (Cont.)

Transport	Station	Controller	IOP
		LG	←LG
		RATE	←RATE
		DATE	←DATE
		NCORE	←NCORE
		READ →	READ
		WRITE →	WRITE
		RESIN, RES →	RESIN, RES
		REV →	REV
		SRIP →	SRIP
		DAPR →	DAPR
		DA0R →	DA0R
		⋮	⋮
		DA7R →	DA7R
		FCR →	RCR
		LD →	LD
		SEP →	SEP
		SRWFIN, RWFIN →	SRWFIN, RWFIN
		WTM →	WTM
		BOT →	BOT

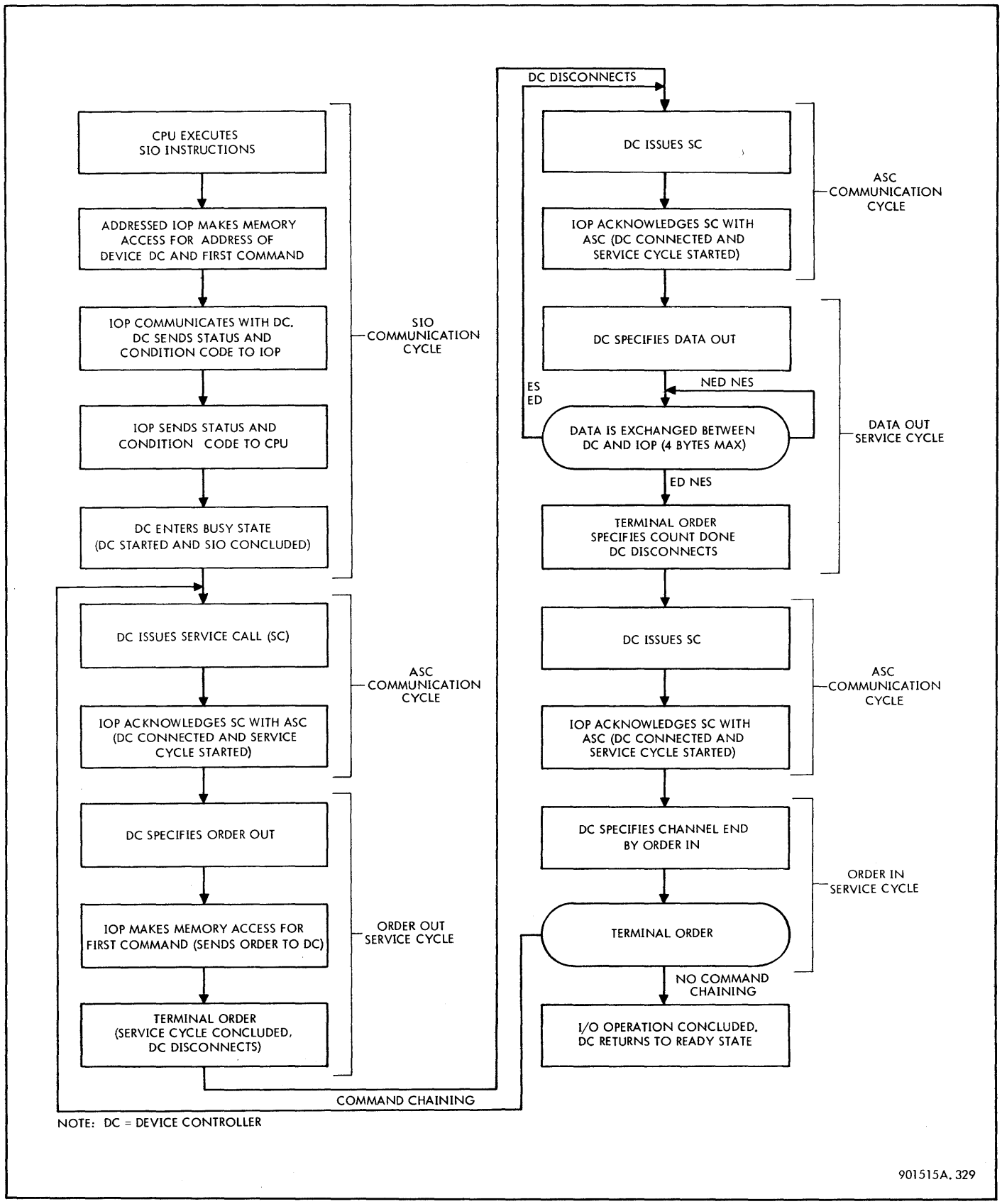


Figure 3-5. Communication and Service Cycle Sequence

ɜ lines. The active controller is connected to the IOP for ·vice, and it goes through a service cycle.

The IOP uses the address that is returned on the function response lines to select the fast memory location from which the required information is to be obtained for the subsequent service cycle to the connected controller. The controller then begins data transfer by raising request strobe (RS). During the time that RS is true, the controller also controls the data order request (DOR) and input/output request (IOR) signals. These two signals are decoded by the IOP to select one of four possible functions that may be performed by the IOP during that cycle.

The service cycle is terminated by two additional signals: end data (ED) and end service (ES). Signal ED may be controlled by either the IOP or the controller. ES is controlled only by the IOP.

The IOP generates an RSA signal in response to each RS signal generated by the controller. Along with RSA, the IOP either provides data (if the subcycle is for data out, or order out, or terminal order) or accepts the data provided by the controller (if the subcycle is for data in or order in).

Line ED may be driven by either the IOP or the controller. The controller may drive this line during order out, order in, sense, and set correction service cycles. The IOP may drive line ED for one of three reasons:

a. An error of some sort has been detected, and the IOP has been programmed to terminate operations when this error is detected.

b. A word boundary has been reached during data transfers, and the IOP would have to go back to the main memory for more data bytes.

c. Signal ED is received from the controller.

Signal ES is made true only by the IOP. It is raised for one of two reasons: The IOP or the controller has signaled ED, and there is no necessity for a terminal order; it is made true during the terminal order. In the first instance, line ES is driven during the same subcycle during which ED was driven, so they both appear together at the controller.

The controller, if it wishes to terminate the service cycle, drives line ED while the last data byte is being input or output. The controller must disconnect following the subcycle in which it sensed ES true.

3-27 Operational Sequence After HIO, TIO, or TDV Instructions. The operational sequences for the HIO, TIO, or TDV instructions are very similar to the operational sequence for the SIO instruction. The 11 low-order bits of the effective address are interpreted (as for the SIO) as specifying an IOP number and an eight bit device address. The primary characteristics of the four instructions are as follows:

a. Instruction SIO causes the addressed controller to be started (although the instruction may be rejected for certain reasons). It also provides the first command doubleword address to be stored in IOP fast memory.

b. Instruction HIO causes an unconditional halt of the controller and the peripheral device; it also resets any waiting interrupts.

c. Instructions TIO and TDV are used by the CPU to obtain a status report from the selected IOP, controller, and station. Instructions TIO and TDV differ only in that the status information returned to the CPU during TIO is general and is independent of the particular controller or device while the status information returned during TDV is specific for the particular device.

3-28 Device Selection. When the IOP raises the SIO, HIO, TIO, or TDV function indicators, the device number is supplied as a byte on the DA lines. The device number for controllers with multiple devices is in the following form:

DAxR Lines	0	1 2 3	4	5 6 7
Contents	1	Device controller number	Not used	Device number

DAxR lines 0, 1, 2, and 3 specify the device controller number, and lines 5, 6, and 7 specify the device number.

The contents of DAxR lines 0 through 3 are compared to the output of four toggle switches in the subcontroller. When a successful comparison is made, signal DCA is true.

The terms TSH and TTSH are generated in the subcontroller. After the IOP raises the function strobe (FS), the subcontroller raises either FSL (function strobe leading acknowledge), which indicates that the function response lines and the condition code lines may be strobed by the IOP or AVO (priority chain signal to the next lower priority controller), depending upon the state of signal DCA.

If the controller address does not agree with the address on the DAxR lines, signal DCA remains false, and AVO is sent to the next lower priority controller in the priority chain. If the controller address does agree with the address on the DAxR lines, signal FSL is made true, and the IOP strobes the function response lines for a status report. The controller is then connected to the IOP for service.

3-29 Priority Determination. When a device controller has been activated by an SIO instruction, it may make

service calls to the IOP until it halts itself or is halted by an HIO instruction. Since many controllers may be connected to one IOP, a priority chain is established to enable the IOP to determine which is the highest priority controller requesting service.

A priority cable is connected between each controller in the chain and between the lowest priority controller and the IOP. Four signals are carried on the cable:

a. HPI – High priority interrupt

b. HPS – High priority service

c. AVO – Available output

d. AVI – Available input

The first two signals (HPI and HPS) constitute a signal bus that is tapped and is driven by each controller. Signal AVO is an output from each controller that is sent to the next lower priority controller in the chain; it is a logical function of AVI (the input priority signal). Signal AVI is always true for the first (highest priority) controller in the chain. Signal AVI for each subsequent controller in the chain is equal to AVO from the preceding controller.

When more than one controller has made a service request at the same time, only one controller may be connected for service. When the IOP acknowledges the service call, the requesting controller with the highest priority responds to the IOP acknowledgement.

If a controller has no service request pending, it passes on signal AVO when it detects signal AVI. If a controller has a service request pending, the controller passes on signal AVO only if HPS is true, or if the controller's service request is of a low priority that the controller has not generated HPS itself. If a controller has a service request pending, the controller does not pass on signal AVO when it detects signal AVI, if it has generated HPS itself, or if HPS is not true.

If, during the execution of an SIO, TIO, TDV, or HIO, a controller does not sense its own address, it passes on signal AVO when it detects signal AVI.

For additional interface information, refer to the SDS Sigma Computer Systems/Interface Design Manual, publication No. 900973.

3-30 Data Transfers

Data transfers to and from the IOP occur during the read/write functions of the controller.

3-31 READ OPERATION. In the read operation, the controller receives input data from the station electronics over nine individual data lines. Each line contains a bit making up a nine-bit character of eight data bits and one parity bit. The read operation takes place in state 03F03U-02U.

3-32 Peak Detector Counter. Each input data line connects to a peak detector counter which starts a count operation when an input line goes true. The counter resets if the input signal is a short duration noise spike, but continues to count if the input is a normal data bit. When the input goes false, the counter stops and transfers the count to the bit crowding counter.

3-33 Bit Crowding Counter. The bit crowding counter accepts the input count from the peak detector counter through gating circuitry that divides the count in half. At the count of 29, the output of the bit crowding counter is transferred to the read deskew counter, if the previous bit in the read register was a one. The transfer takes place at a count of 31 if the previous bit in the read register was a zero.

3-34 Read Deskew Counter. The input signal from the bit crowding counter starts a count operation in the read deskew counter. The count is used to correct read signals which have skew errors. The output of the counter goes to the read deskew comparator.

3-35 Read Deskew Comparator. The read deskew comparator selects the period required to correct skew errors in a particular channel. When the tape unit is initially set up, each read channel is adjusted for minimum skew by removeable jumper wiring in each channel in the station. When the count in the read deskew counter equals the count set up by the jumper wiring in the station, the output of the comparator is true. The output of each read deskew comparator is connected to the set input of a flip-flop in the assembly register. All nine read deskew comparator outputs are connected in parallel to the input of the assembly period counter.

3-36 Assembly Period Counter. The assembly period counter specifies the period during which the assembly, and the read registers can accept input data from the read deskew comparators.

When the first bit from any one of the nine read deskew comparators arrives, the assembly period counter starts its count operation. As each bit arrives, it is also connected to the set input of flip-flop in the assembly register. When the assembly period is over (during which the assembly register can accept data for that particular character), the assembly period counter generates two signals. The first signal transfers the contents of the assembly register to the read register; the second signal resets the assembly register.

3-37 Assembly and Read Registers. Each input bit is connected to the set input of a flip-flop in the assembly register. The flip-flops are clocked, and are set during the assembly period if the set input is true. The flip-flop outputs are connected to the set inputs of the flip-flops in the read register. When the assembly period counter terminates the assembly period, it generates a signal that clocks the flip-flops in the read register. They are set to the condition at that time of the flip-flops in the assembly register. The assembly period counter then generates another signal that resets the assembly

register. The assembly register remains reset until the next character arrives.

3-38 Parity Error Detector. The output of the read register is connected to a data bus (paragraph 3-39) and to the parity error detector. The parity error detector samples each byte from the read register and detects any parity errors in that byte. The output of the parity error detector is connected to a data bus in conjunction with the read register for a read with correction order. In a normal read operation, the parity error detector output is loaded into the error pattern register in a particular manner.

3-39 Error Pattern Register. The error pattern register receives its input from the parity error detector in a read operation. As each parity error is detected, it is stored in the error pattern register in a distinct format that is used later to determine which channel contains the errors. In a read with correction operation, the error pattern register corrects the output of the channel that contains the errors. The output of the register is connected to a data bus. In a read after write operation, the LRC check is performed in this register.

3-40 Data Bus. There are 18 data bus circuits in the controller; nine read circuits and nine write circuits. Each read circuit receives input from the parity error detector, the error pattern register, and the read register. The data bus transfers data to the write register, the CRC register, the error pattern register, and the eight-byte buffer.

3-41 Memory Mode Select and Timing. Timing circuits are required to regulate the various functions that are being performed and to select the correct mode for data transfer between the IOP and the tape and between the tape and the IOP. The memory mode select and the timing circuitry perform this function.

It receives an input from the read register when the first character is read from the tape and initiates a timing sequence. The timing sequence clocks the read/write memory address counters and the eight-byte buffer memory for the read operation.

The memory mode select and the timing circuitry receives an input from the memory character counter in a write operation, and initiates a timing sequence that clocks the read/write memory address counters and the eight-byte buffer.

3-42 Read and Write Memory Address Counters. The read memory address counter selects the address of bytes that are read from the eight-byte buffer memory. The write memory address counter selects the address of bytes that are stored in the eight-byte buffer memory.

Each counter receives its clock input from the memory mode select and the timing circuitry. Each time one of the counters receives a clock signal from the timing circuitry, it changes its count and selects the next address for either a read or a write data transfer.

The output of each counter is connected through gating circuitry, along with the output of the memory mode select and timing circuitry, to the address inputs of the eight-byte buffer. Each counter also generates a clock signal that is connected to the clock inputs of the memory character counter.

3-43 Memory Character Counter. The memory character counter is an up-down counter that indicates the number of characters that the eight-byte buffer contains at any given time. It also initiates signals that cause data to be stored in memory or to be read from memory.

The counter uses two clock signals generated by the read and the write memory address counters. One clock signal causes the counter to count up as data is being stored in memory. The other clock signal causes the counter to count down as data is read from memory.

The counter generates a signal that calls for additional data to be stored in memory when the count falls to four or less in a write operation. It also generates a signal that causes data to be read from memory when the count is five or more in a read operation. In each case, a service call is raised.

3-44 Eight-Byte Buffer Memory. The eight-byte buffer memory stores up to eight bytes in a transfer of data between the IOP and the tape in a write operation or between the tape and the IOP in a read operation.

Each character is stored in memory at an individual address determined by the count of the read and write address counters.

The read register supplies input data to the memory in a read operation, and the IOP supplies input data to the memory in a write operation. The input data is presented to the buffer memory over nine individual data lines, gated with outputs from the memory mode select and timing circuitry. The address of the characters stored in memory is determined by the output of the write address counter and the memory mode select and timing circuits. The input character is clocked into memory by the memory mode select and the timing circuits.

The read memory address counter, the memory mode select, and the timing circuitry select the address of the characters read from memory. A clock signal is not required to read data from memory; it is only necessary to select the address of the character to be read.

3-45 Write Register. In a read operation, the write register checks the validity of the LRC character and determines which channel is in error in a track-in-error byte generation.

3-46 LRC Character Check. The input signals for the write register originate in the read register in a read operation. These signals are presented to the set inputs of the flip-flops in the write register on nine individual data lines from the data bus. The flip-flops in the write register are

clocked and follow the input data by one clock time. When the LRC character is read from tape, it resets all flip-flops in the write register if it is not in error. If it is in error, the register gives an error indication. In a read after write operation, the LRC check is performed identically in the E register.

3-47 Track-In-Error Determination. If the CRC register and the error pattern register do not compare after a record has been read, one or more channels on the tape contain errors. The write register is used in a track-in-error operation as a shift register to determine which channel (or track) contains the error. The first flip-flop in the series of nine in the write register is set, and the CRC register is shifted and is compared again with the error pattern register. If they agree, the write register indicates which channel is in error. If they do not agree, the write register is shifted once more, the CRC register is shifted, and the comparison is again attempted. If they agree, the write register indicates which channel is in error. If they do not agree, the operation is repeated again. If the marker bit that is being shifted through the write register is shifted completely through the register without a comparison being made, the error (noncorrectable read error) indicates that there are errors in more than one channel.

3-48 CRC Register. The CRC (cyclic redundancy check) register is a nine-stage shift register that generates a CRC character from data read from tape in a read operation.

The input signals for the CRC register originate in the read register in a read operation and are presented to the set inputs of the flip-flops in the CRC register on nine individual data lines from the data bus. As the record is read from tape, the register is shifted in a distinct pattern. After the CRC character is read from tape, the CRC register contains the final character that is to be checked for validity. If the record is correct, the register contains the pattern 111010111. The pattern is now compared in the CRC error comparator with the pattern in the error pattern register. The CRC error comparator indicates the state of the two registers. If they agree, a comparison signal is generated; if they do not agree, a noncompare signal is generated. If the two registers do not agree, the CRC register is alternately compared and shifted until they do agree or until the error is declared a noncorrectable read error.

3-49 Read Order Termination. When the read operation is finished, the IOP signals count done and the controller advances to state 03F01U (order finalization).

3-50 SPACE FILE ORDER. When the controller is given a space file order and the tape mark record is read from tape, the controller advances to state 03F01U (order finalization).

3-51 SPACE RECORD ORDER. When the controller is given a space record order and signal LG (long gap) is read from tape, the controller advances to state 03F01U.

3-52 WRITE OPERATION. In the write operation the controller receives input data from the IOP over nine individual data lines. Each line contains a bit making up a nine-bit character of eight data bits and one parity bit. The data lines are connected to the input of the eight-byte buffer memory where they are gated with signals from the memory mode select and timing circuitry. The write operation takes place in state 03F03U-02U.

3-53 Write Deskew Counter. The write deskew counter divides the output of the master clock signal to provide accurate reference timing signals for the write operation.

The write deskew counter input is the master clock signal which is divided down until the final output of the counter is a 60-kHz signal. The counter outputs are used to initiate the transfer of data to the write register and the eight-byte buffer memory, to initiate a transfer of data from the write register in the controller to the write data register in the station, and to control the operation of the flip-flops in the write data register in the station.

Four outputs of the counter are used to control the flip-flops in the write data register. These outputs control the set, reset, and clock input signals.

The counter generates control signal CWR, which is true when all stages of the counter are reset. This signal is sent to the memory mode select and to the timing circuitry. It is used to initiate a timing sequence that transfers a byte from the eight-byte buffer memory to the write register; from the write register, it transfers a byte to the write data register in the station.

Other signals are generated that are used in the write deskew comparator circuits in the station for the correction of write errors caused by skew.

3-54 Memory Mode Select and Timing. When a write operation is started for the first time, there is no data stored in the eight-byte buffer memory. The memory character counter is reset (indicating no data in memory), and signal BMT from the counter is true. (BMT is true whenever the memory character counter has a count of four or less in a write operation.) BMT initiates the service call signal CSL, and the IOP answers and puts data on the nine data lines to the eight-byte buffer memory.

A timing sequence is initiated in the memory mode select and timing circuitry when the data is on the input lines. The input character is gated with other outputs from the memory mode select and timing circuitry and is presented to the input of the eight-byte buffer. The character is then clocked into memory. This process can continue until more than four bytes are stored in memory.

If signal CWR from the write deskew counter comes true, data is read from memory and is transferred to tape. CWR initiates a timing sequence in the memory mode select and timing circuitry, and the read memory address counter is

clocked and counts up by one. It is used to select the address of the byte to be read from memory.

The process of alternately storing and reading data is repeated for the entire record and is dependent upon the timing sequence of the memory mode select and timing circuitry.

3-55 Read and Write Memory Address Counters. Refer to paragraph 3-42 for a functional description of the read and write memory address counters.

3-56 Memory Character Counter. Refer to paragraph 3-43 for a functional description of the memory character counter.

3-57 Eight-Byte Buffer Memory. Refer to paragraph 3-44 for a functional description of the eight-byte buffer memory.

3-58 Write Register. The write register stores data from the eight-byte buffer memory and generates the LRC character in a write operation.

In a write operation, the input characters from the eight-byte buffer memory are presented to the set inputs of the flip-flops in the write register over nine individual data lines from the data bus.

The memory mode select and timing circuitry generates a timing sequence that clocks the input character into the register, and the write deskew counter generates a signal that resets the flip-flops in the register. The flip-flop outputs of the write register are connected to the clock inputs of flip-flops in the write data register in the station.

3-59 LRC Character Generation. Each flip-flop in the write register is clocked and is set at the time the LRC character is to be recorded. Any flip-flop in the write data register that was reset at that time is now set, giving an even channel parity to the LRC character.

3-60 CRC Register. The CRC register generates the CRC character that is written on tape from the input data supplied by the IOP.

The input data from the eight-byte buffer memory is presented to the set inputs of the flip-flops in the CRC register over nine individual data lines from the data bus. As the input characters are clocked into the register, they are shifted in a distinct pattern. After the last character in the record has been stored and shifted in the CRC-register, it is loaded selectively complemented into the write register, and it clocks the flip-flops in the write data register in the station. The CRC character is then recorded in the fourth character space after the last character in the record.

3-61 Read After Write. During a write operation, data recorded on tape is immediately read back for checking in a read-after-write operation. This operation is very similar to a normal read operation. The operation of the counters and the registers in the read circuitry is basically the same as explained for the read operation, with the exception of the error pattern and the CRC registers. The CRC register is not used in a read-after-write operation, since it is being used in the write process to generate the CRC character that is to be recorded.

The flip-flop outputs of the read register are presented to the set inputs of the error pattern register over nine individual data lines. The flip-flops in the error pattern register now toggle with each input level change. After the LRC character has been recorded and read back, each flip-flop in the error pattern register is reset and signal ERZ is true if all characters in the record are correct. If they are not correct, ERZ is false, indicating an error.

3-62 Write Order Termination. When the write order is finished, the IOP signals count done, and the controller advances to state 03F01U (order finalization).

SECTION IV
PRINCIPLES OF OPERATION

4-1 INTRODUCTION

This section contains a detailed description of the controller operations, using logic equations and flow, timing, block, and logic diagrams to clarify the description. Definitions of the terms used in the equations in this section can be found in the glossary in section V.

4-2 CONTROLLER STATES

The controller goes through eight states in performing its functions. The states are determined by the condition of of the state counter which consists of flip-flops FF1, FF2, FU1, FU2, and the associated logic. See figure 4-1 for a flow diagram of the controller states. It should be noted that once an equation is given for a specific logic function, it will be repeated only if the terms change in a subsequent function.

4-3 00F00U (IDLE OR READY)

The controller enters state 00F00U for any of the following reasons:

a. Power is initially applied.

b. I/O reset is generated by the CPU.

c. The CPU performs a halt instruction.

d. The IOP indicates halt by a terminal order (unusual end).

e. Channel end takes place without command chaining.

f. The RESET switch on the station has been activated (unusual end, if busy).

g. An unusual end is reported.

h. Channel end and device end do not occur together.

During this state the controller is in an idle or ready condition. It can generate new interrupts, or can accept an SIO if no interrupts are pending. Upon accepting an SIO, the controller advances to the next state. See figure 4-2 for a flow diagram of the SIO operation, and figure 4-3 for a timing diagram of state 00F00U.

The controller enters state 00F00U when flip-flops FF1, FF2, FU1, and FU2 are dc reset and when NFF1, NFF2, NFU1, and NFU2 go true.

00F = NFF1 NFF2

00U = NFU1 NFU2

E/FF1 = RSTA

RSTA = DACFDD HLTD + RSTS

DACFDD = DACFD FSR + . . .

DACFD = (FD1 DA5R + NFD1 NDA5R)
(FD2 DA6R + NFD2 NDA6R)
(FD3 DA7R + NFD3 NDA7R)

FSR = Function strobe receiver output (signal FS from IOP)

HLTD = NFSR AT0

AT0 = DCA FSD HIOR + . . .

RSTS = (RSTR + NINI) NMAN

RSTR = I/O reset receiver output (signal RST from IOP)

NINI = Switch contact signal INI inverted (diode clamp gate for power failure or initialize)

NMAN = NMANC (selector toggle switch output for PET panel)

E/FF2 = RSTA

E/FU1 = RSTS

E/FU2 = RSTS

When flip-flops FF1 and FF2 are reset, the controller raises CRD (controller ready) if there are no interrupts pending.

CRD = NTSH NMAN 00F + . . .

NTSH = NDCA
+ NSIOR NHIOR NTIOR

The controller is now ready to accept an SIO from the IOP, and when it arrives, receiver output SIOR (signal SIO from IOP) goes true.

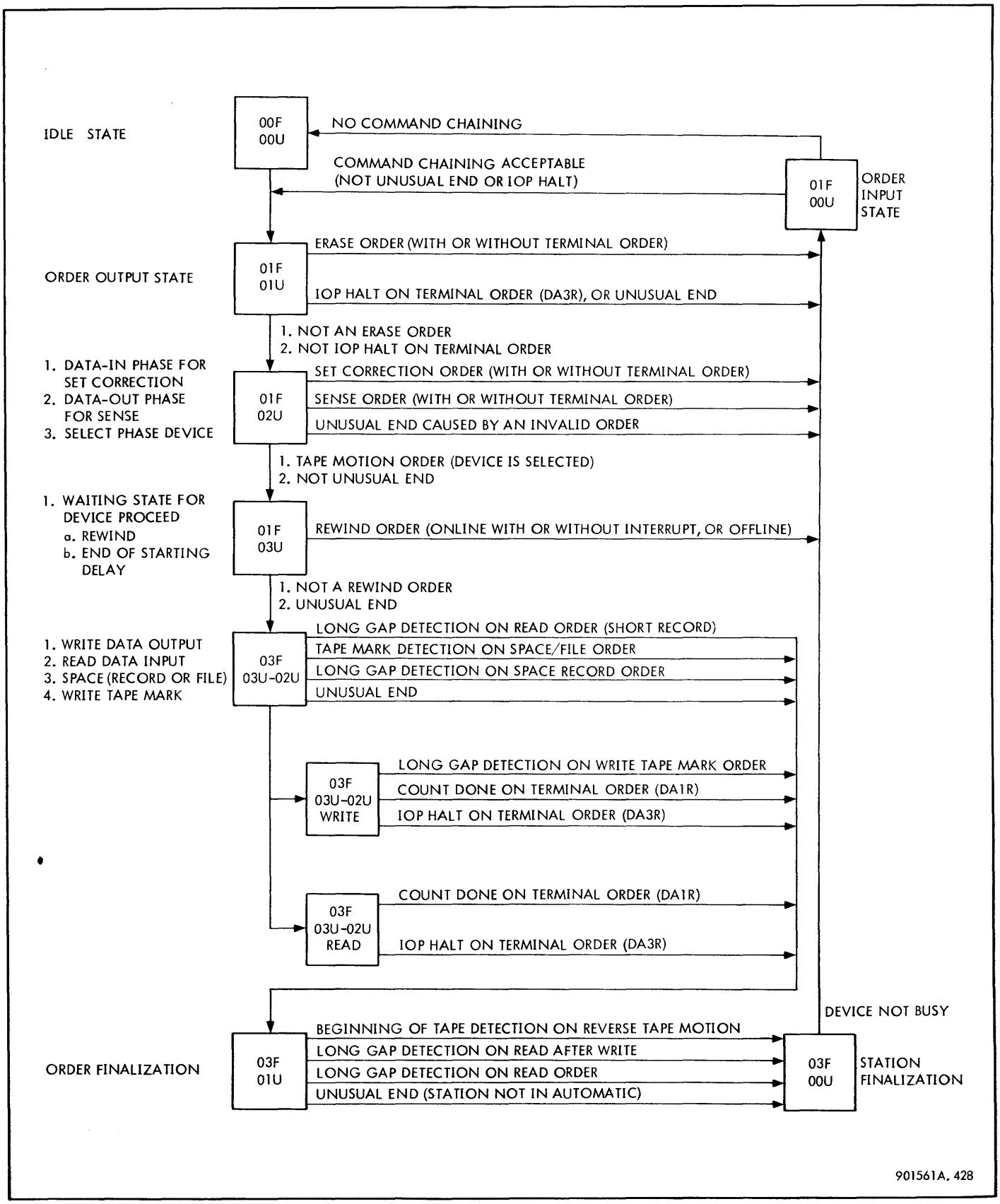


Figure 4-1. Magnetic Tape Controller States, Flow Diagram

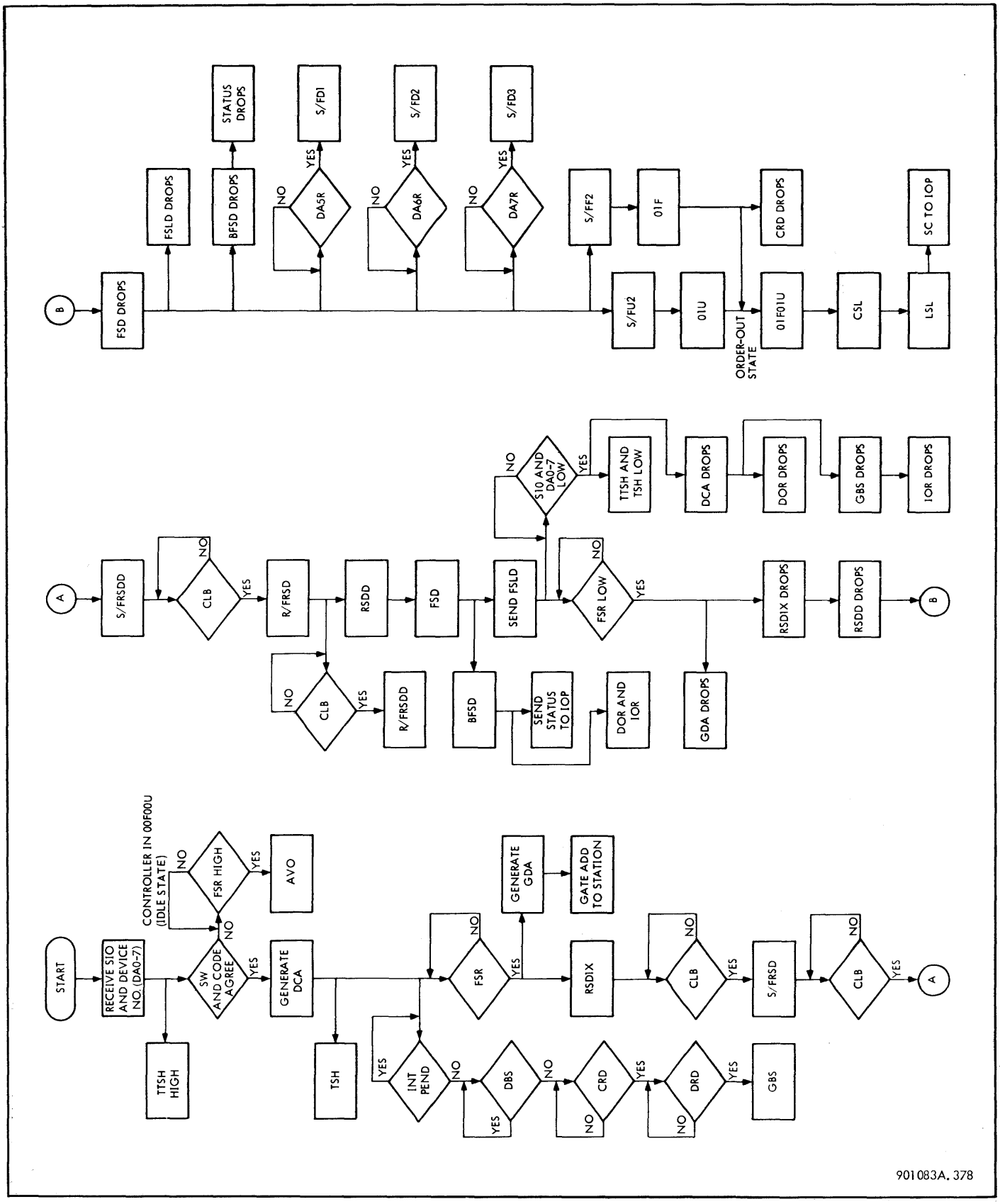
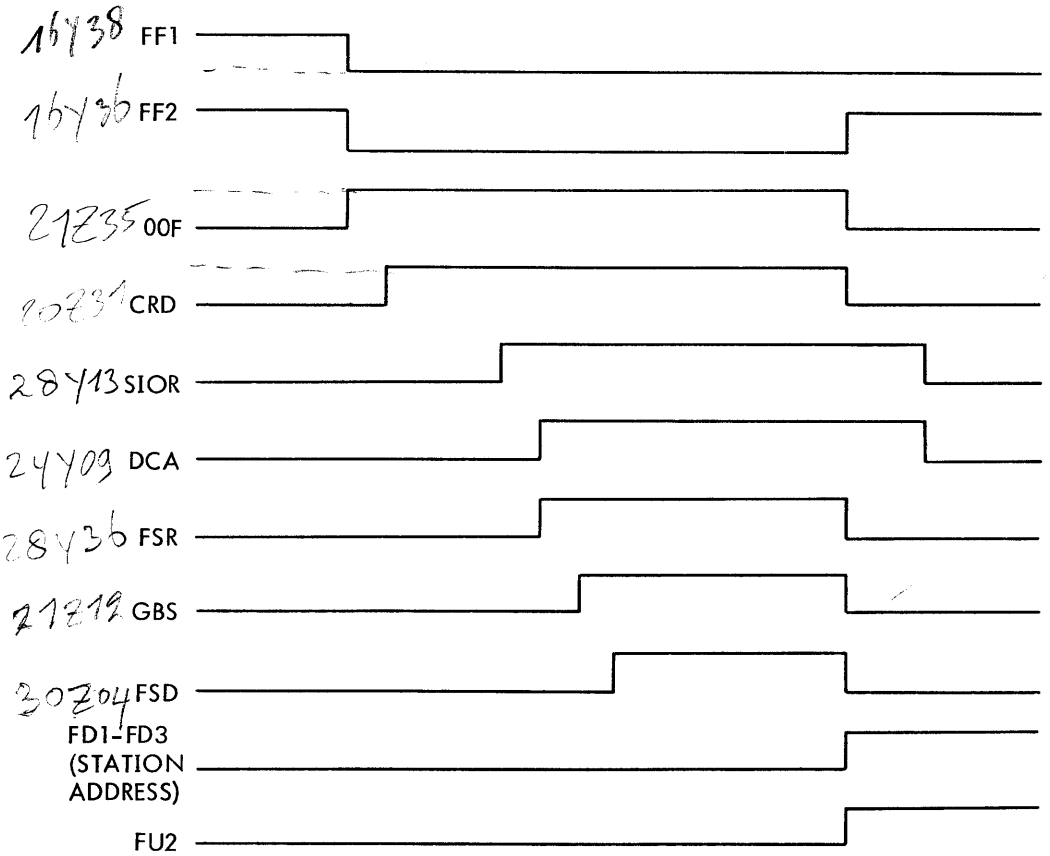


Figure 4-2. SIO Operation, Flow Diagram



NOTE: THE STATES OF FD1-FD3 ARE DETERMINED BY
THE ADDRESS ON THEIR SET INPUTS

901561A. 402

Figure 4-3. 00F00U State (Idle or Ready), Timing Diagram

Controller switches SWA0 through SWA7 and IOP data lines DA0 through DA7 establish the controller and the station numbers used in device selection. In tape systems with more than one station, switches SWA0 through SWA3 and lines DA0 through DA3 determine controller address, and DA5 through DA7 determine the station address. SWA0 is always set to 1 and DA0 is always true. When the controller number on the DA lines matches the switch settings, DCA goes true and, in turn, causes GBS to go true.

DCA = N (SWA0 NDA0R + NSWA0 DA0RC + SWA1 NDA1R + NSWA1 DA1RC + SWA2 NDA2R + NSWA2 DA2RC + SWA3 NDA3R + NSWA3 DA3RC)
N (FSC)

GBS = NDBS NINTPEND CRD DCA SIOR DRD

DRD = Operational control NFCTS NFS1 (NTSH + DRD)

The DA lines designating the station are connected to the set inputs of flip-flops FD1, FD2, and FD3, which are clocked by BAND05.

E/FD1-FD3 = RSTA

S/FD1 = DA5R

C/FD1 = BAND05

BAND05 = GBS FSD

FSD = RSDD (GDA + . . .)

RSDD = FRSDD NFRSD + . . .

GDA = DCA TTSH FSR

TTSH = SIOR + . . .

S/FD2 = DA6R

C/FD2 = BAND05

S/FD3 = DA7R

C/FD3 = BAND05

The IOP raises FS, and FSR goes true. The controller then raises FSD, making clock input BAND05 true. When the controller drops FSD, flip-flops FD1, FD2, and FD3, are clocked and set, if their set inputs are true at this time. The outputs of FD1, FD2, and FD3 go to decoder DFD0 through DFD7, and the decoder output raises lines DV0D through DV7D. These terms are compared with the station address selection switch positions during the device select state 01F02U.

At the time that GBS comes true, the set inputs to flip-flops FF2 and FU2 also comes true. The raising of FSD causes their clock inputs to go true.

S/FF2 = SFF2

SFF2 = GBS NFF2 + . . .

C/FF2 = CFF2

CFF2 = GBSOPEST

GBSOPEST = GBS FSD + . . .

S/FU2 = GBSODST + . . .

GBSODST = GBS 00F + . . .

C/FU2 = CFU2

CFU2 = GBSOPEST + . . .

When FSD drops, it also clocks and sets FF2 and FU2, advancing the controller to state 01F01U.

4-4 01F01U (ORDER OUTPUT)

The controller advances to state 01F01U for the following reasons:

a. An SIO has been accepted from the IOP.

b. The controller has reported channel end with command chaining.

01F01U = 01F 01U

01F = FF2 NFF1

01U = FU2 NFU1

See figure 4-4 for a flow diagram of state 01F01U and figure 4-5 for a timing diagram of state 01F01U.

On entering state 01F01U, flip-flops FPE, FUN, and FCN are dc reset.

E/FPE = BOR03

BOR03 = BAND06 + . . .

BAND06 = 01F01U NESR

E/FUN = BOR03

E/FCN = BOR03

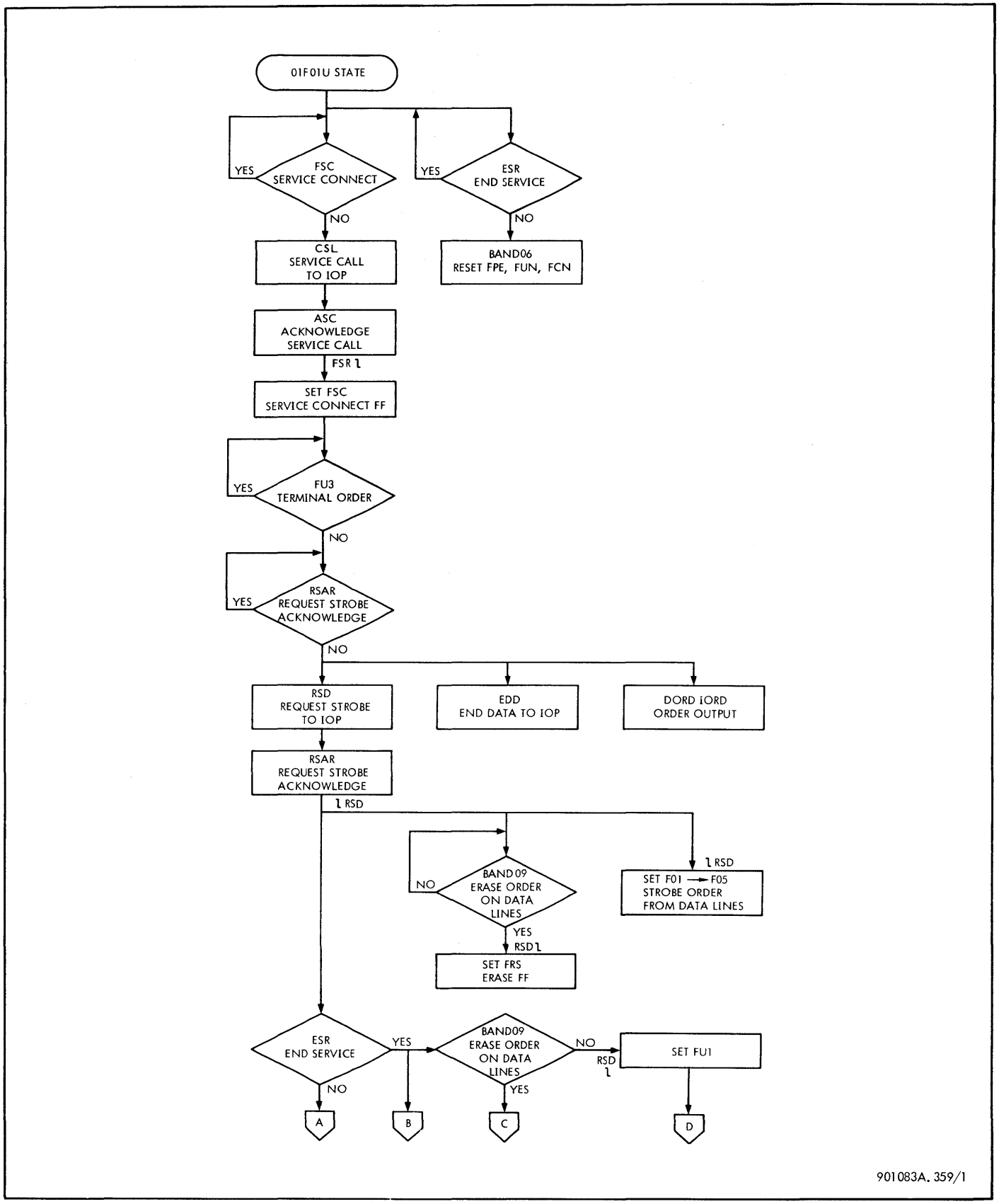


Figure 4-4. 01F01U State (Order Output), Flow Diagram (Sheet 1 of 2)

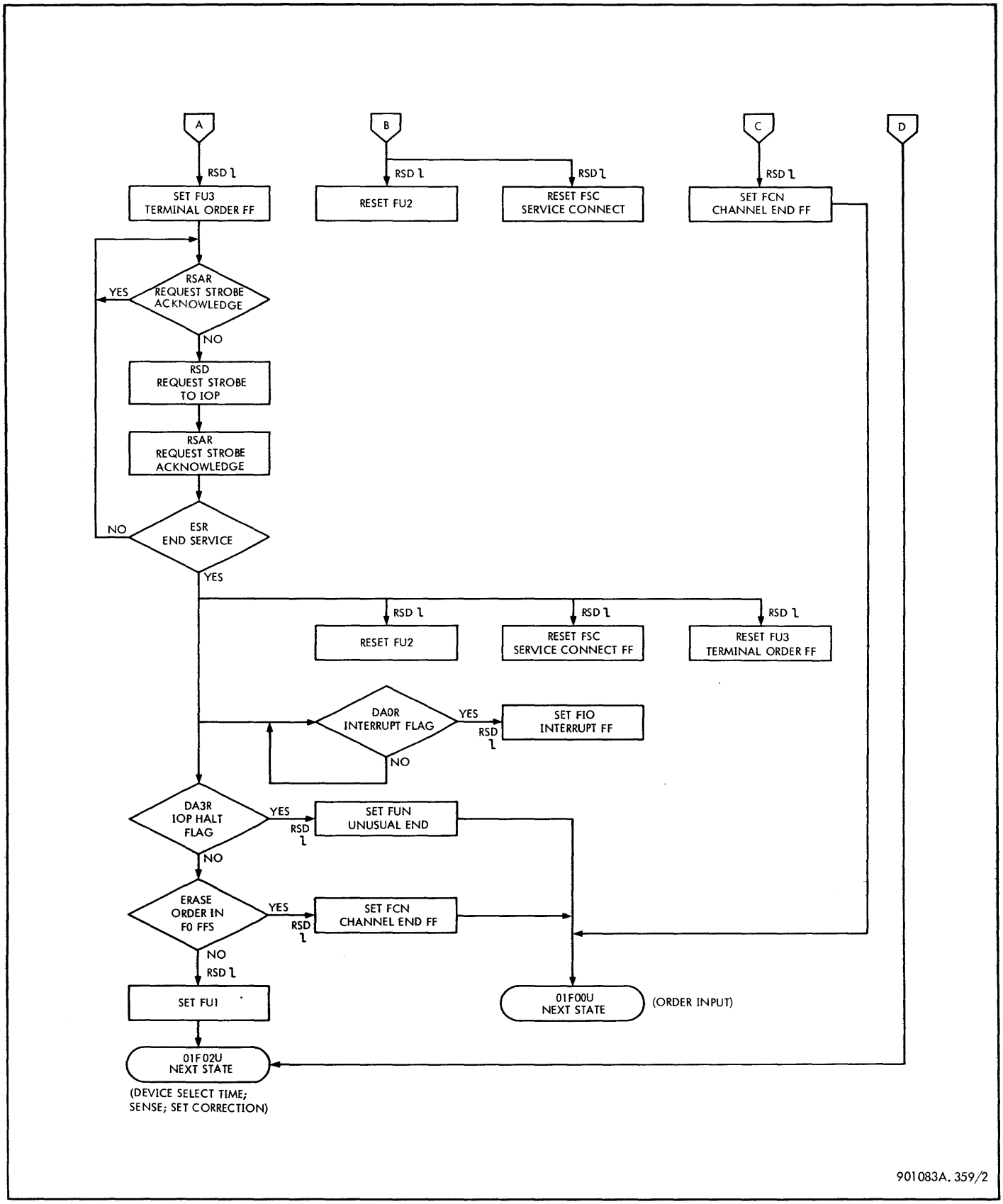
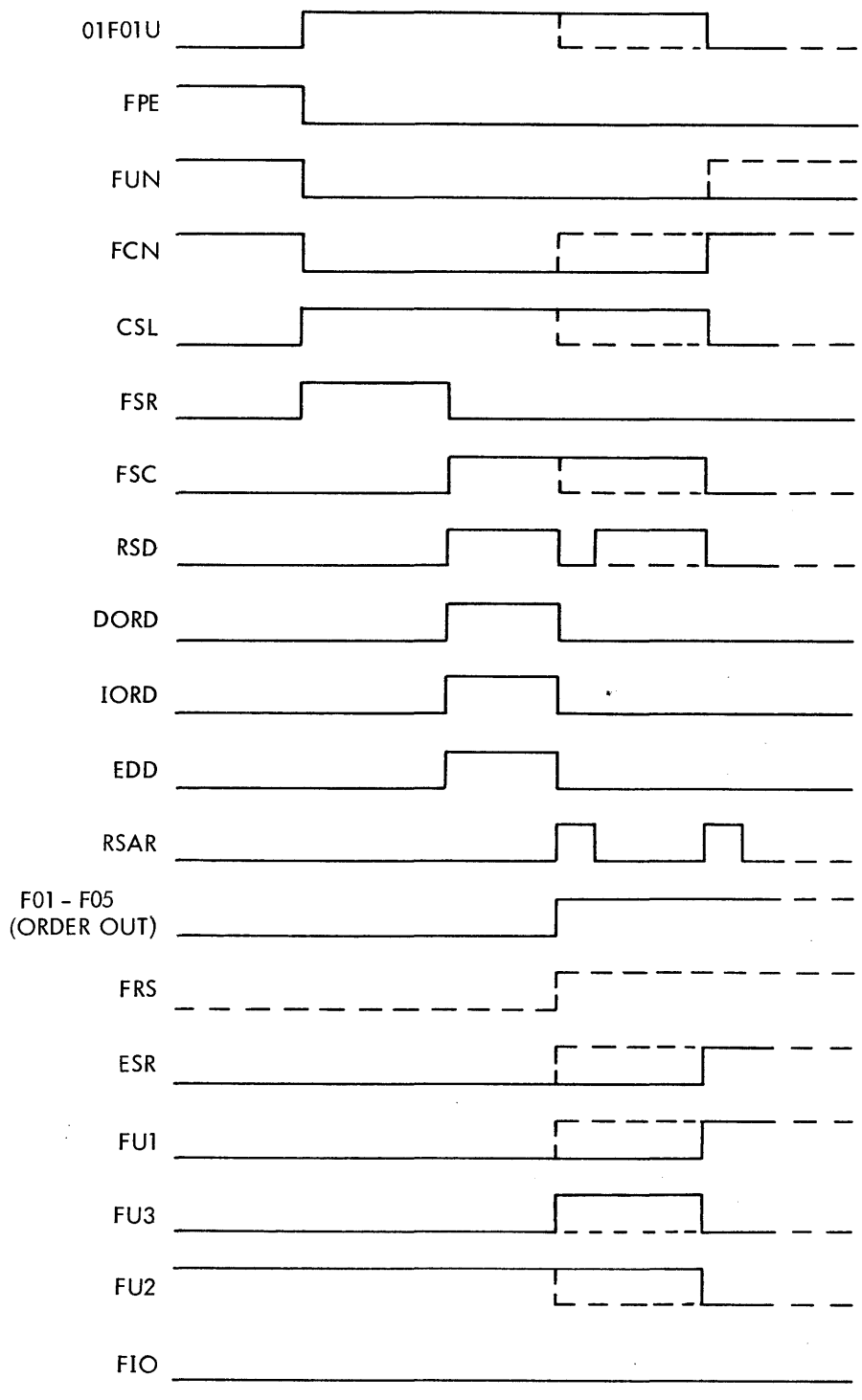


Figure 4-4. 01F01U State (Order Output), Flow Diagram (Sheet 2 of 2)



NOTE:

1. THE STATES OF F01 - F05 ARE DETERMINED BY THE ORDER ON THEIR SET INPUTS
2. DASHED LINE — — — INDICATES ERASE ORDER OR SIGMA 2 SEQUENCE

901013A. 401

Figure 4-5. 01F01U State (Order Output), Timing Diagram

The controller generates a service call request to the IOP by raising CSL.

CSL = CSLX CSLI

CSLX = 01F01U + . . .

CSLI = Service request inhibit signal (100 ns delay of NFSC)

The service call is latched by LSL and is sent to the IOP as SC.

LSL = LSL INI NFSC ASCR NRSTR + CSL INC NFSC NASCR

The IOP raises acknowledge service call (ASC), FS and available input (AVI). These signals generate the set and the clock inputs for service connect flip-flop FSC, which has initially been dc reset.

E/FSC = RSTR + . . .

S/FSC = ASCB

ASCB = AVIR ASCR ASCM FSR + . . .

AVIR = Available input receiver output

ASCR = Acknowledge service call receiver output (signal ASC from IOP)

ASCM = LSL NHPSL + . . .

NHPSL = NHPSL ASCR + . . .

INI = Initialize controller signal

C/FSC = FSR NFSC + . . .

The set and the clock inputs to FSC are now true, and FSC is clocked and set when FSR goes false. At this time, the request strobe driven (RSD), order request (DORD), order output request (IORD), and end data (EDD) lines go true.

RSD = RSDX FSC NMAN + . . .

RSDX = NRSAR 01F01U + . . .

DORD = BAND08 + . . .

BAND08 = 01F01U NFU3 RSD

IORD = BAND08 + . . .

EDD = BAND08 + . . .

The controller uses the DORD and the IORD lines to specify to the IOP the type of communication which is to take place. The DORD and the IORD line designations are as follows:

	DORD	IORD
Order out	1	1
Order in	1	0
Data out	0	1
Data in	0	0

DORD and IORD are both true in this state, which is an order out operation.

The IOP presents the order information to the controller on lines DA0 through DA7, and it appears on the set outputs of order flip-flops F01 through F05. The status of the data lines and the flip-flops for each order is shown in table 4-1.

Table 4-1. Data Line and Order Flip-Flop Information

STATUS OF DATA LINES								DESCRIPTION	STATUS OF ORDER FLIP-FLOPS				
DA0	DA1	DA2	DA3	DA4	DA5	DA6	DA7		F01	F02	F03	F04	F05
				0	1	0	0	Sense	0	0	1	0	0
				1	1	0	0	Read backward	0	1	1	0	0
				0		0	1	Write	0	0	0	0	1
				0		1	0	Read forward	0	0	0	1	0
	0	0	0			1	1	Set correction (to be used with next read)	1		0	0	0
	0	0	1	0		1	1	Rewind and interrupt	1		0	0	1
	0	1	0	0		1	1	Rewind offline	1		0	1	
	0	1	1	0		1	1	Rewind	1	0	0	1	1
	1	C	0	D*		1	1	Space record	1	D*	1	0	0
	1	0	1	D†		1	1	Space file	1	D†	1	0	1
	1	1	0			1	1	Set erase (to be used with next write)	1	0	1	1	0
	1	1	1	0		1	1	Write tape mark	1	0	1	1	1

*Space record forward if D = 0; space record backward if D = 1

†Space file forward if D = 0; space file backward if D = 1

17Y24 17Y27 17Y38 17Y23 17Y17

The IOP places the order on lines DA0 through DA7, and it is applied to the set inputs of flip-flops F01 through F05, which are initially dc reset.

E/F01-F05 = RSTS

S/F01 = CTL

CTL = DA6R DA7R

S/F02 = DA4R

S/F03 = SF03

SF03 = CTL DA1R + NCTL DA5R

S/F04 = SF04

SF04 = CTL DA2R + NCTL DA6R

S/F05 = SF05

SF05 = CTL DA3R + NCTL DA7R

The clock inputs to the flip-flops are true at this time.

C/F01-F05 = BAND08

The IOP then raises the request strobe acknowledge (RSA) and RSAR goes true. RSD now goes false, clocking flip-flops F01 through F05 which set to the order that is on their set inputs.

RSAR = RSARC NMAN + . . .

RSARC = Request strobe acknowledge receiver output (signal RSA from IOP)

The controller now performs its logic functions and advances to either 01F00U (order input) or 01F02U (device selection), depending upon the order in flip-flops F01 through F05.

4-5 Erase Order

If the data lines contain an erase order when the IOP raises ESR (end service), when RSD goes false it sets FCN and FRS and resets FU2 and FSC, advancing the controller from 01F01U to state 01F00U.

ESR = ESRC NMAN + . . .

ESRC = End service receiver output (signal ES from IOP)

S/FCN = SFCN + . . .

SFCN = NFU3 BAND09 BAND11 + . . .

BAND09 = CTL DA1R DA2R NDA3R (erase order)

BAND11 = 01F01U ESR

C/FCN = CFCNY RSD1 + . . .

CFCNY = 01F01U + . . .

S/FRS = SFRS

SFRS = SFRSX BAND10 + . . .

SFRSX = BAND09 + . . .

BAND10 = 01F01U NFU3

C/FRS = CFRS

CFRS = CFRSX RSD

CFRSX = NFRS BAND10

C/FU2 = CFU2

CFU2 = CFU2X RSD + . . .

CFU2X = 01F01U

R/FSC = ESRC FSC

C/FSC = RSD FSC + . . .

If ESR is raised but if the data lines do not contain an erase order, then flip-flops FU2 and FSC are clocked and reset, and flip-flop FU1 is clocked and is set when RSD goes false, advancing the controller to the other possible state, 01F02U.

S/FU1 = SFU1X

SFU1X = BAND11 BAND12 F0134NF05I

BAND12 = NDA3R FU3

F0134NF051 = NF01 + NF03 + NF04 + F05

C/FU1 = CFU1

CFU1 = CFU1X RSD + . . .

CFU1X = 01F01U + . . .

4-6 Terminal Order

With Sigma 5 and 7 only, a terminal order is always required during the order out state. The IOP does not raise ESR, but instead flip-flop FU3 (terminal order) is clocked and set when RSD goes false and flip-flops FU2 and FSC remain in the set condition.

S/FU3 = SFU3

SFU3 = EDR NESR FSCC + . . .

EDR = EDRC NMAN + . . .

EDRC = End data receiver output (signal ED from IOP)

FSCC = FSC NMAN + . . .

C/FU3 = CFU3

CFU3 = FSCC RSD + . . .

The controller again raises RSD, the IOP sends the terminal order on the DAx lines, and then raises ESR and RSAR. When RSAR goes true, RSD goes false and resets flip-flops FU2, FU3, and FSC.

If the IOP gave the controller a halt command (DA3R true) in the terminal order, flip-flop FUN (unusual end) sets when RSD goes false, advancing the controller to state 01F00U.

S/FUN = SFUN + ...

SFUN = SFUNX + ...

SFUNX = FU3 NDST ESR DA3R

C/FUN = FU3 NFUN NDST RSD1 + ...

If the controller does not contain an erase order, and if no IOP halt is received, flip-flop FU1 sets when RSD goes false, advancing the controller to the other possible state, 01F02U.

4-7 01F02U (DEVICE SELECT TIME)

The controller advances to state 01F02U and enters one of three possible phases:

a. Data output phase for set correction

b. Data input phase for sense

c. Device select phase

01F02U = 01F 02U

02U = FU1 NFU2

See figure 4-6 for a flow diagram of state 01F02U.

4-8 Sense

The sense order transfers the track-in-error byte to the IOP during a read operation and initiates the set correction process. A description of the sense order is given in paragraph 2-24. See figure 4-7 for a timing diagram of this phase.

If the controller contains a sense order when entering state 01F02U, SCO is false and SEP is true.

SEP = 01F02U SEPD

SEPD = NF01 NF02 F03 NF04 NF05

When SEP comes true, it also enables RW4 which gates the track-in-error byte from the write register to the DA lines.

RW4 = (CCRC NCRCC) SEP CMPR

The controller raises CSL, and the IOP raises ASC, FSR, and AVI. The set and the clock inputs to flip-flop FSC come true, and FSC is clocked and set when FSR goes false. At the same time, RSD, EDD, NDORD NIORD (data input), and ABD (enable data read), come true.

CSL = CSLX CSLI

CSLX = SEP + ...

ABD = SEP BAND16 + ...

BAND16 = NFU3 RSD

The IOP raises ESR and RSAR, and RSD goes false. When RSD goes false, it clocks and sets flip-flop FCN and resets flip-flops FU1 and FSC.

S/FCN = SEP ESR + ...

C/FCN = CFCNY RSD

CFCNY = SEP + ...

C/FU1 = CFU1

CFU1 = CFU1X RSD + ...

CFU1X = SEP + ...

The controller is advanced to state 01F00U with a sense order and without a terminal order.

4-9 Terminal Order (Sense)

If a terminal order is required, the IOP does not raise ESR. Instead, flip-flop FU3 is clocked and set when RSD goes false, and FCN remains in the reset state. FSC and FU1 remain in the set state.

The controller again raises RSD, and the IOP raises RSAR and ESR. When RSAR goes true, RSD goes false, clocking and setting FCN, and resetting FU1, FU3, and FSC. The controller is advanced to state 01F00U with a sense and a terminal order.

If the IOP gives the controller an interrupt in the terminal order, flip-flop FIO clocks and sets when RSD goes false.

S/FIO = DA0R SFIO

SFIO = FU3 NDST

NDST = SEPD + ...

If the IOP gives the controller a halt command in the terminal order, flip-flop FUN clocks and sets when RSD goes false.

4-10 Set Correction

The set correction order transfers the track-in-error byte from the IOP to the controller error pattern (EP) register during this phase. A description of the set correction order is given in paragraph 2-24. See figure 4-8 for a timing diagram of this phase.

SCO is true if the controller contains a set correction order when entering state 01F02U.

SCO = 01F 02U SCOD

SCOD = F01 NF03 NF04 NF05

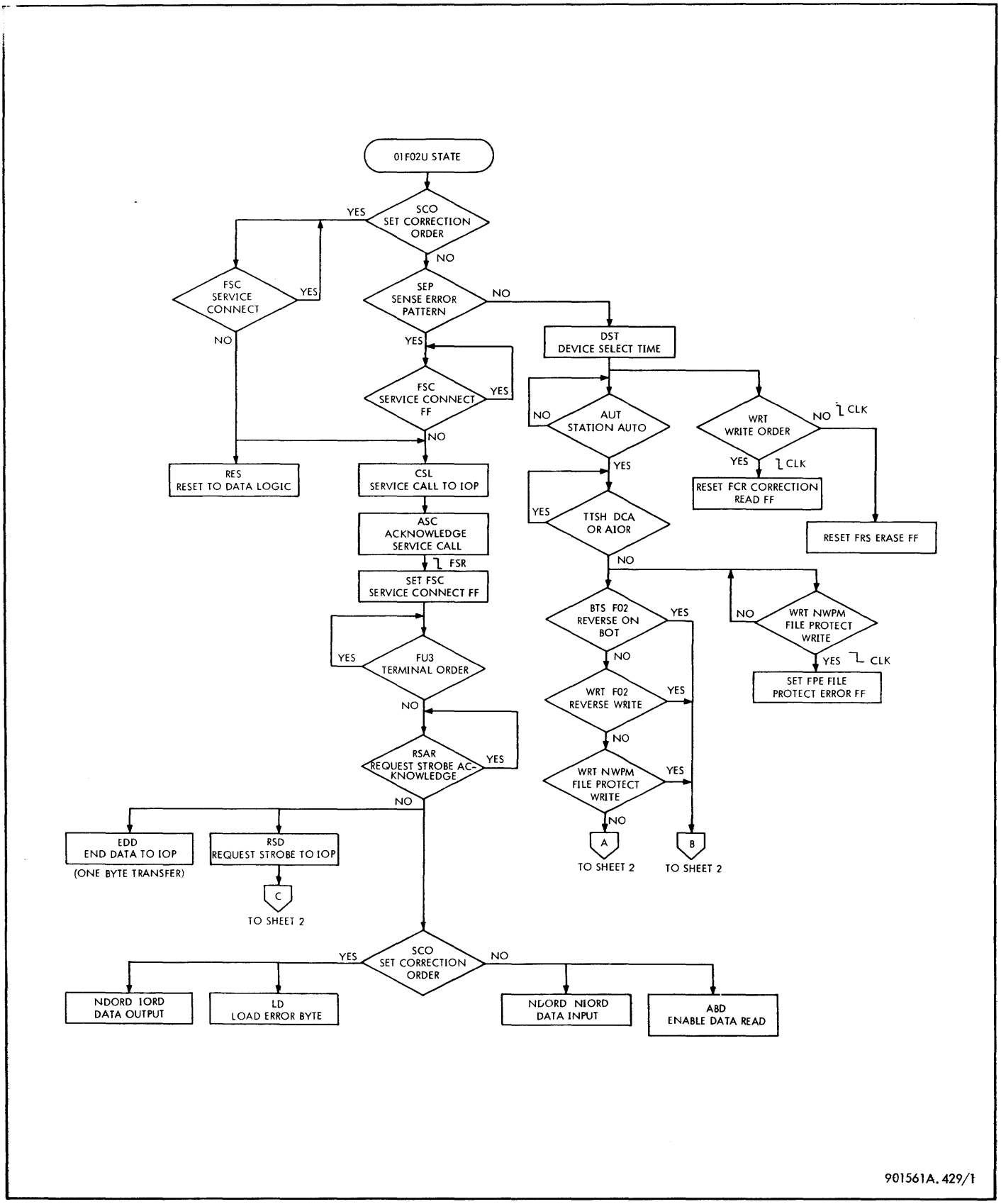


Figure 4-6. 01F02U State (Device Select Time), Flow Diagram (Sheet 1 of 2)

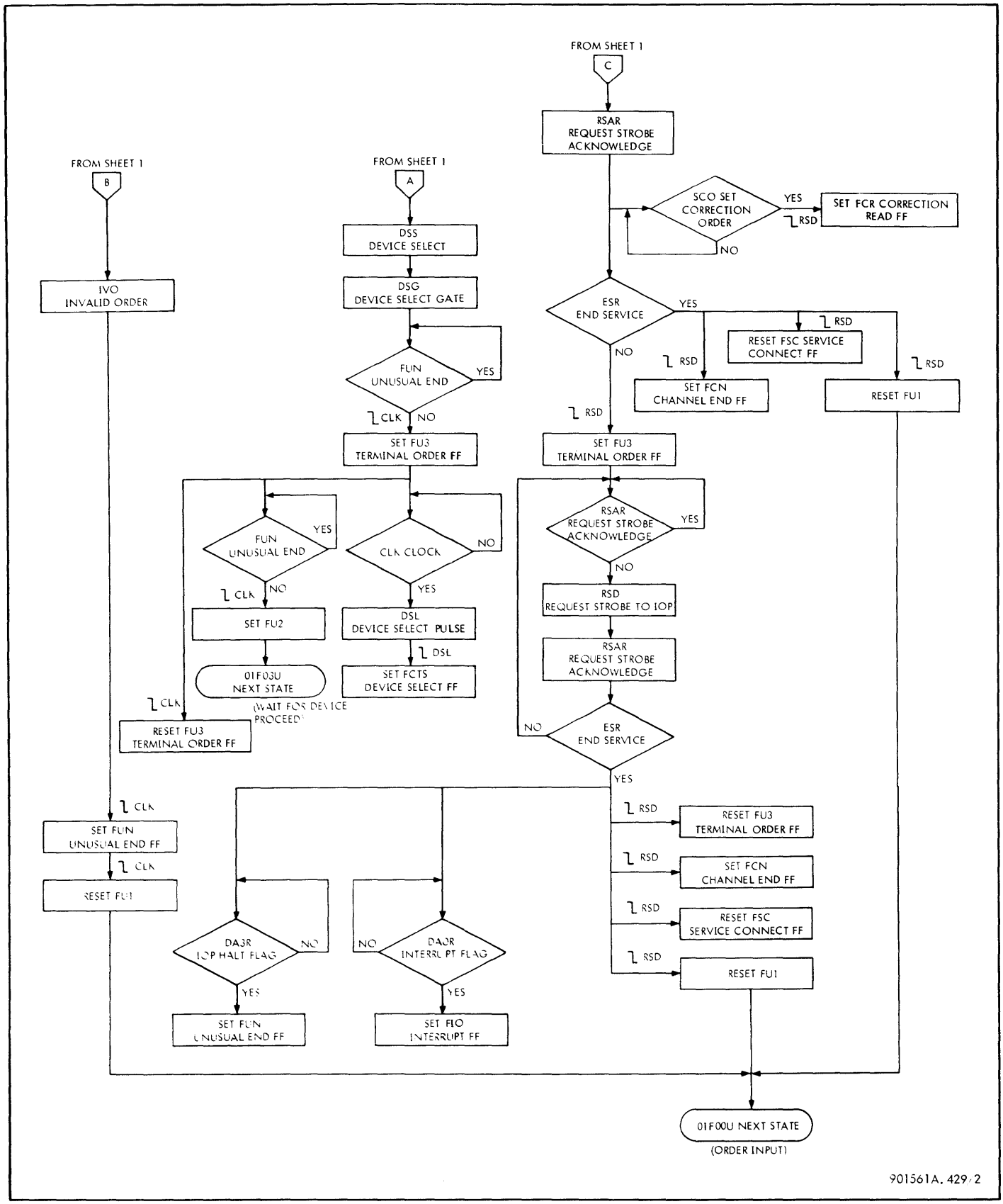
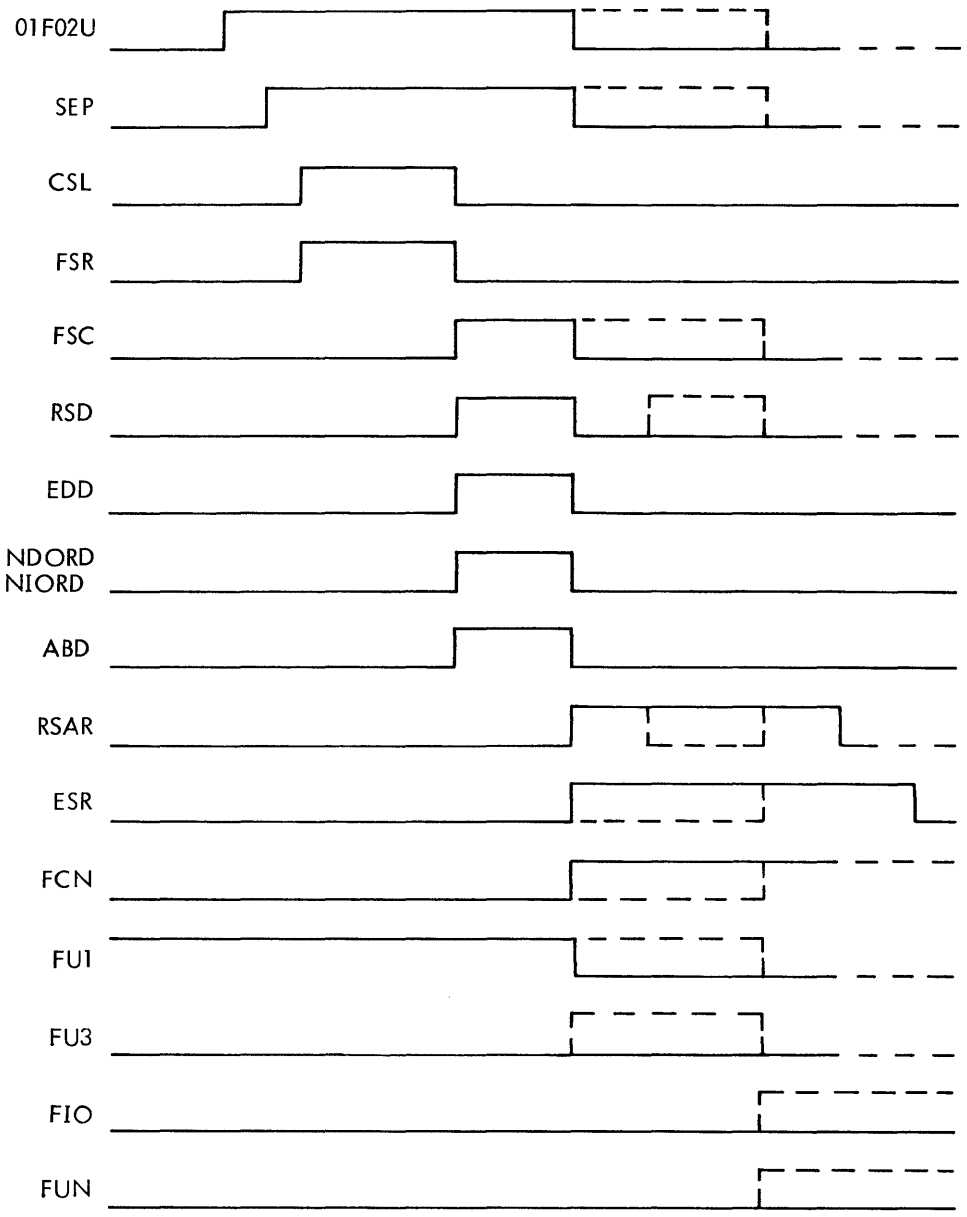


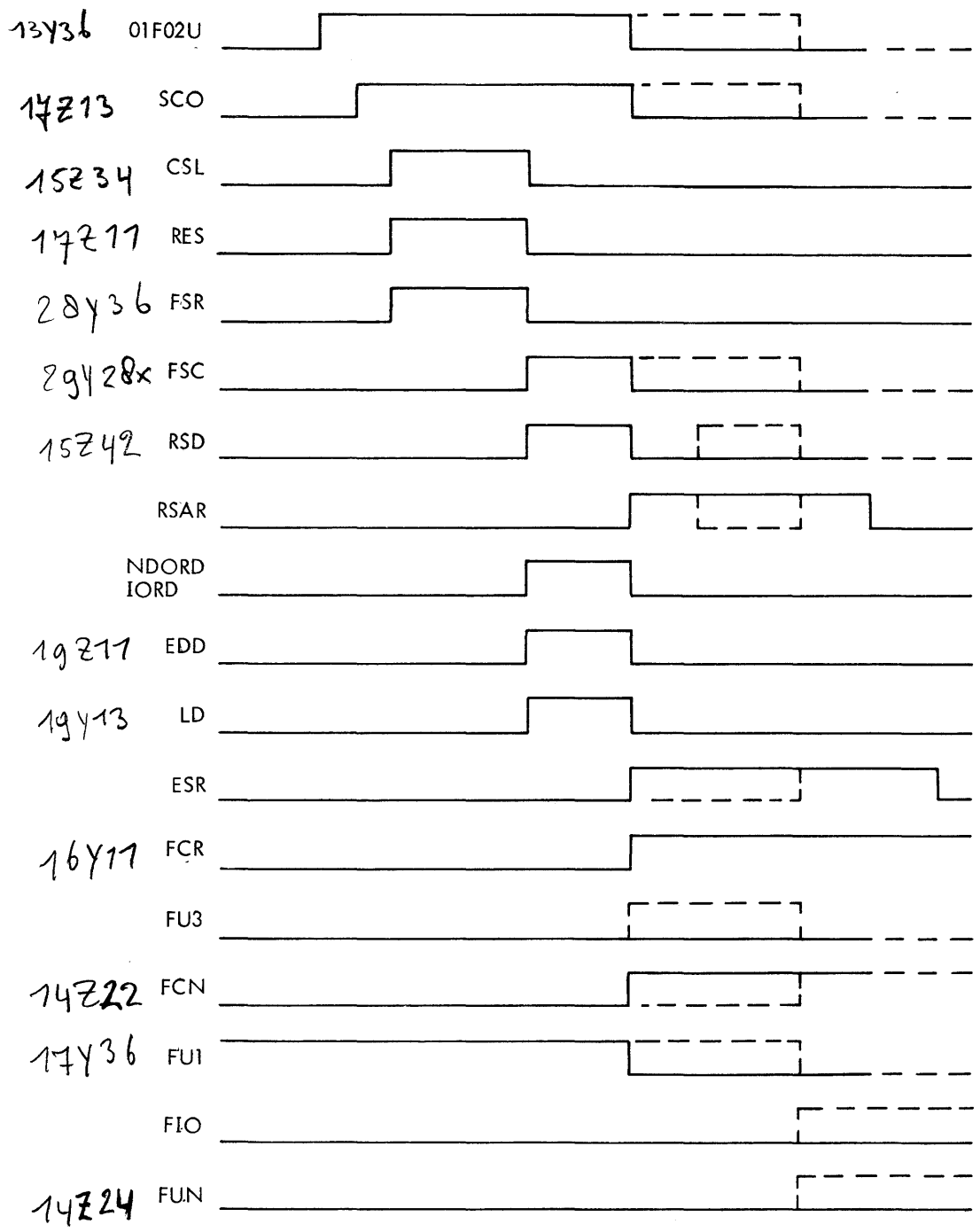
Figure 4-6. 01F02U State (Device Select Time), Flow Diagram (Sheet 2 of 2)



NOTE:
DASHED LINE (- - -) INDICATES ALTERNATE STATE

901561A. 404

Figure 4-7. 01F02U State (Sense Phase), Timing Diagram



NOTE:
DASHED LINE (- — -)INDICATES ALTERNATE STATE

901561A.403

Figure 4-8. 01F02U State (Set Correction Phase), Timing Diagram

RES (data logic reset) comes true, and the controller raises CSL.

RES = SCO NFSCC CSLI + ...

NFSCC = NFSC NMAN + ...

CSL = CSLX CSLI

CSLX = SCO + ...

The IOP raises ASC, FSR, and AVI. The set and clock inputs for flip-flop FSC are now true, and FSC is clocked and set when FSR goes false. At the same time, RSD, NDORD IORD (data output), EDD, and LD (load error byte) come true.

LD = SCO NFU3 RSD1

LD is the input gating term to the E-register to receive the track in error byte.

The IOP raises ESR and RSAR, and RSD goes false. When RSD goes false, it clocks and sets flip-flops FCR (correction read) and FCN, and resets flip-flops FU1 and FSC. The controller is advanced to state 01F00U with a set correction order and without a terminal order.

S/FCR = SFCR

SFCR = BAND18 + ...

BAND18 = SCO NFU3

C/FCR = CFCR

CFCR = CFCRX RSD

CFCRX = BAND18 + ...

S/FCN = SFCN

SFCN = SCO ESR + ...

C/FCN = CFCNY RSD1

CFCNY = SCO + ...

C/FU1 = CFU1

CFU1 = CFU1X RSD + ...

CFU1X = SCO + ...

When FCR comes true, the track-in-error byte is transferred into the EP register at the same RSD time.

Note

The following equations are applicable to stages E1 through E9 of the EP register.

S/Ex = NEx DCx EPRL + ...

DCx = DAxR LD SELECT + ...

LD SELECT = NWRITE1

EPRL = FCR + NREAD

READ = 03F NF01 NWRT

C/Ex = EP REG CLOCK

EP REG CLOCK = NSHIFT LD NLRCT

E/Ex = EPR RESET

EPR RESET = SCO NFSCC CSLI NFCR + 01F03U NDPR

4-11 Terminal Order (Set Correction)

If a terminal order is required, the IOP does not raise ESR. Instead, flip-flops FCR and FU3 are clocked and set when RSD goes false. FCN remains in the reset state, and FSC and FU1 remain in the set state.

The controller again raises RSD, and the IOP raises RSAR and ESR. When RSAR goes true, RSD goes false, clocking and setting FCN, and resetting FU1, FU3, and FSC. The controller is advanced to state 01F00U with a set correction and a terminal order.

If the IOP gives the controller an interrupt in a terminal order, flip-flop FIO clocks and sets when RSD goes false.

S/FIO = DA0R SFIO

SFIO = FU3 NDST

NDST = SCOD + ...

If the IOP gives the controller a halt command in the terminal order, flip-flop FUN clocks and sets when RSD goes false.

4-12 Device Select Phase

In the device select phase, the controller separates valid and invalid orders and connects the device for operation. See figure 4-9 for a timing diagram of this phase.

DST (device select time) comes true if SEP and SCO are false when the controller enters state 01F02U.

DST = 01F 02U NSEPD NSCOP

If the controller contains a write order when entering state 01F02U, WRT comes true, and CLK goes false, clocking and and resetting flip-flop FCR.

WRT = NFPET1 F05NF014 + ...

NFPET1 = Flip-flop output for PET panel

F05NF014 = F05 NF01 NF04

C/FCR = CFCR

CFCR = CFCRY CLK + ...

CFCRY = DST WRT

CLK = (WDC 45C)

If the controller does not contain a write order, flip-flop FRS is clocked and is reset when CLK goes false.

C/FRS = CFRS

CFRS = DSTNWRT CLK + ...

The controller checks the station at this time for automatic mode, which the station indicates by raising AUT. If TTSH,

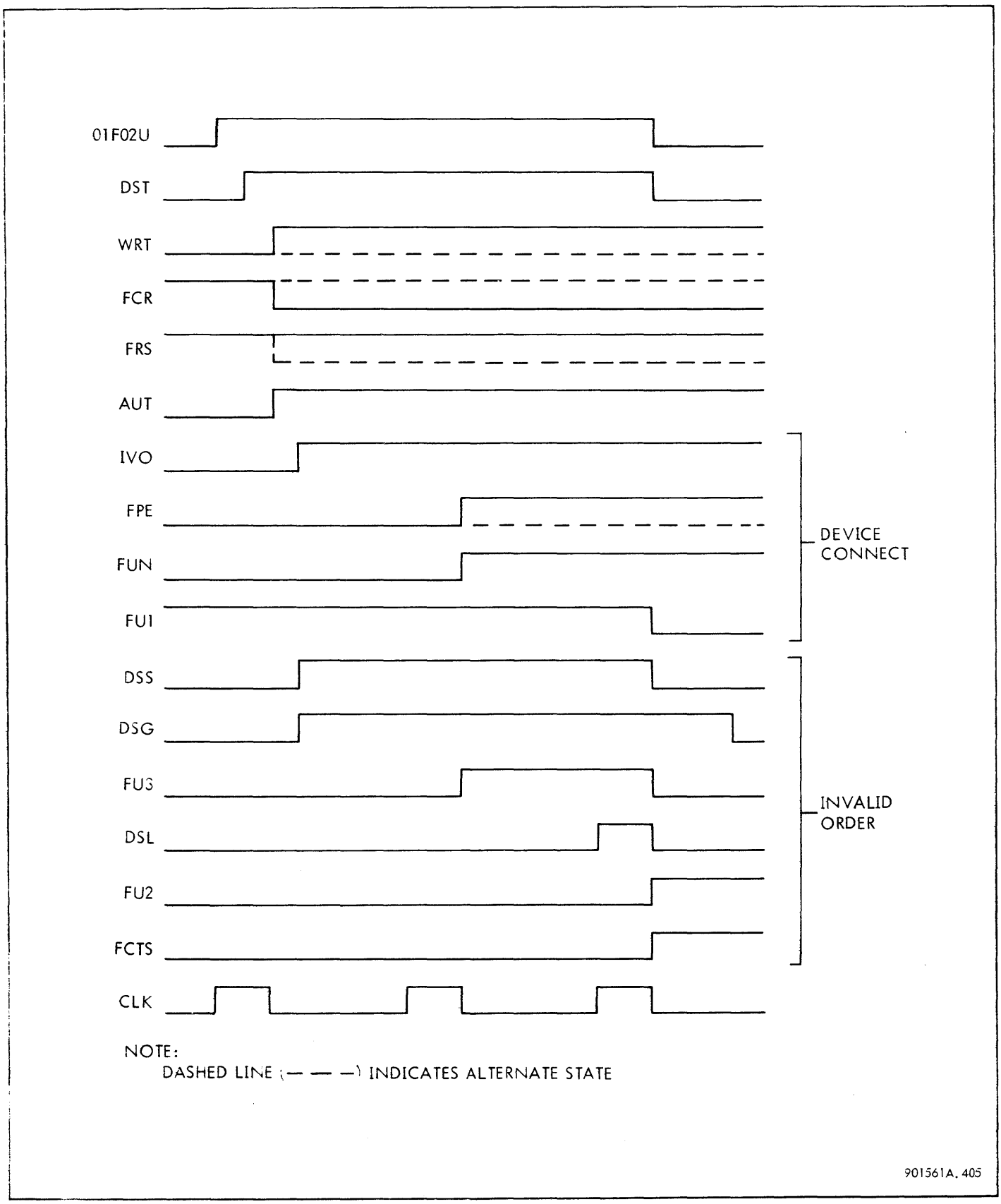


Figure 4-9. 01F02U State (Device Select Phase), Timing Diagram

DCA or AIOR is not raised at this time, the controller then proceeds to check for invalid orders.

IVO comes true if an invalid order is detected.

IVO = BTS F02 (reverse on BOT) + WRT F02 (reverse write) + NWPM WRT (file protect write)

BTS = Beginning of tape (signal BTSC from station)

NWPM = N(WPM)

WPM = Addressed device file protected (signal WPMC from station)

If the invalid order is a file protect write, flip-flop FPE (file protect error) is clocked and set when CLK goes false.

S/FPE = NWPM WRT + . . .

C/FPE = CLK BAND17

Flip-flop FUN is clocked and set when CLK goes false.

S/FUN = SFUN + . . .

SFUN = CFUNX + . . .

CFUNX = BAND17 IVO + . . .

BAND17 = DST NOR01 AUT

NOR01 = NAIOR (NDCA + NTTSH)

C/FUN = CFUNX CLK

The clock input to flip-flop FU1 now comes true, and the next time that CLK goes false, it clocks and resets FU1. The controller is advanced to order input state 01F00U on an invalid order.

C/FU1 = CFU1

CFU1 = CFU1Z CLK + . . .

CFU1Z = DST FUN

If the controller does not contain an invalid order, DSS (device select) comes true.

DSS = NIVO BAND17

The station raises the device select gate (DSG) and, since there is not an unusual end condition, flip-flop FU3 clocks and sets when CLK goes false.

S/FU3 = SFU3

SFU3 = BAND20 DSG NFU3

BAND20 = DST NFUN

C/FU3 = CFU3

CFU3 = DST CLK

The controller responds to the station signal by raising the device select pulse (DSL) the next time that CLK comes true.

DSL = CLK FU3 DST

When CLK again goes false, flip-flop FU3 is clocked and reset. DSL goes false and flip-flop FU2 is clocked and set. Flip-flop FCTS (device connect) in the station is clocked and set when DSL goes false, and the setting of FU2 advances the controller to state 01F03U (wait for device proceed) on a device select order.

S/FU2 = SFU2

SFU2 = GBSODST + . . .

GBSODST = DST + . . .

C/FU2 = CFU2

CFU2 = CFU2Y CLK + . . .

CFU2Y = BAND20 FU3

After flip-flop FCTS is set, the station initiates tape motion.

4-13 01F03U (WAIT FOR DEVICE PROCEED)

The controller advances to state 01F03U and initiates either a rewind order or an advance to the next state for a read, write, or space operation.

01F03U = 01F 03U

03U = FU1 FU2

See figure 4-10 for a flow diagram, and figure 4-11 for a timing diagram of state 01F03U.

4-14 Rewind

The controller can initiate either of two rewind orders at this time.

a. Rewind online with interrupt

b. Rewind offline

4-15 REWIND ONLINE WITH INTERRUPT. When the rewind order is initiated, the controller raises WN102.

WN102 = WN1 + . . .

WN1 = F01 NF03 F05

In response to the rewind order, the station raises DPR (device proceed), resets flip-flop FCTS, and initiates reverse tape motion.

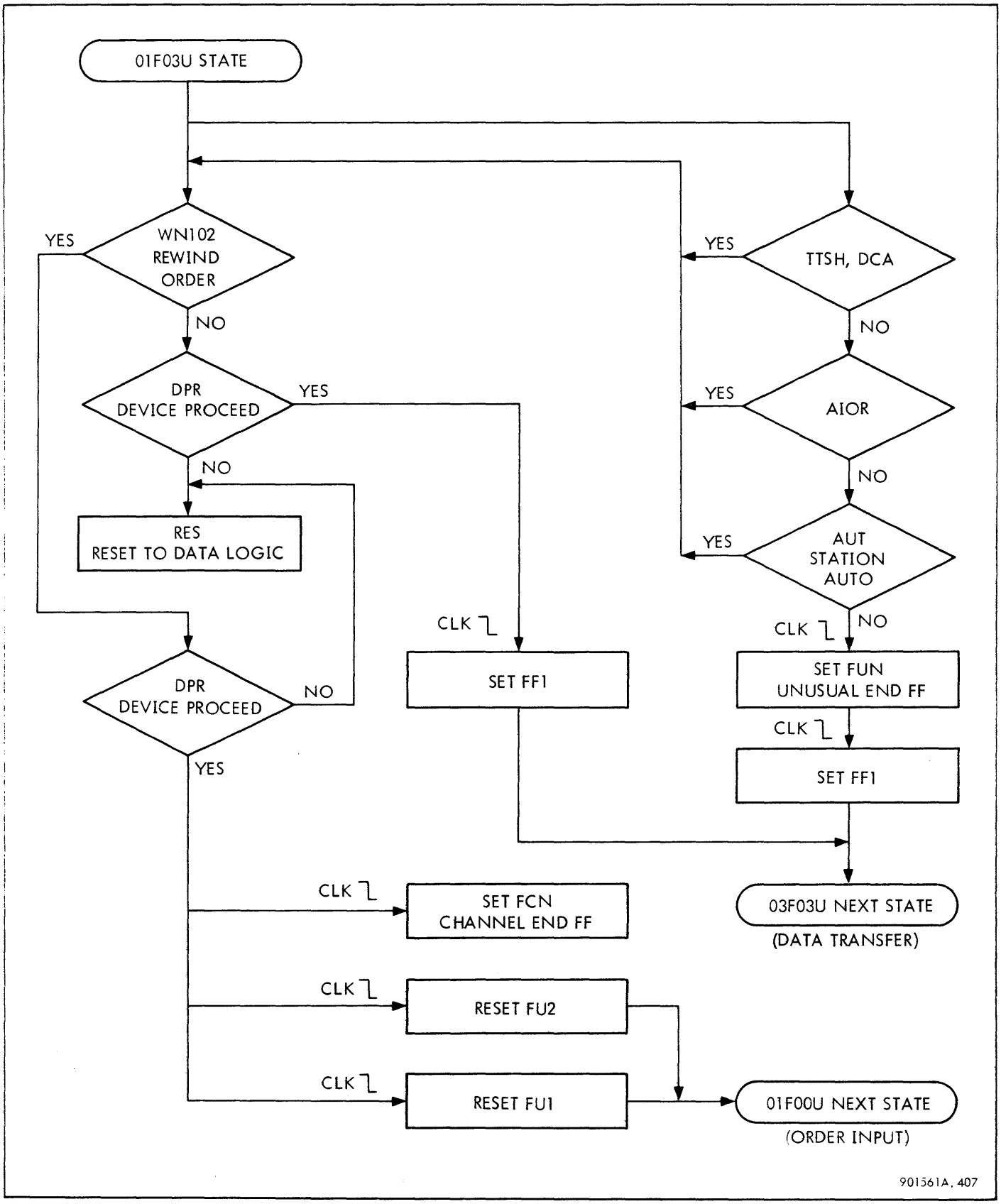


Figure 4-10. 01F03U State, Flow Diagram

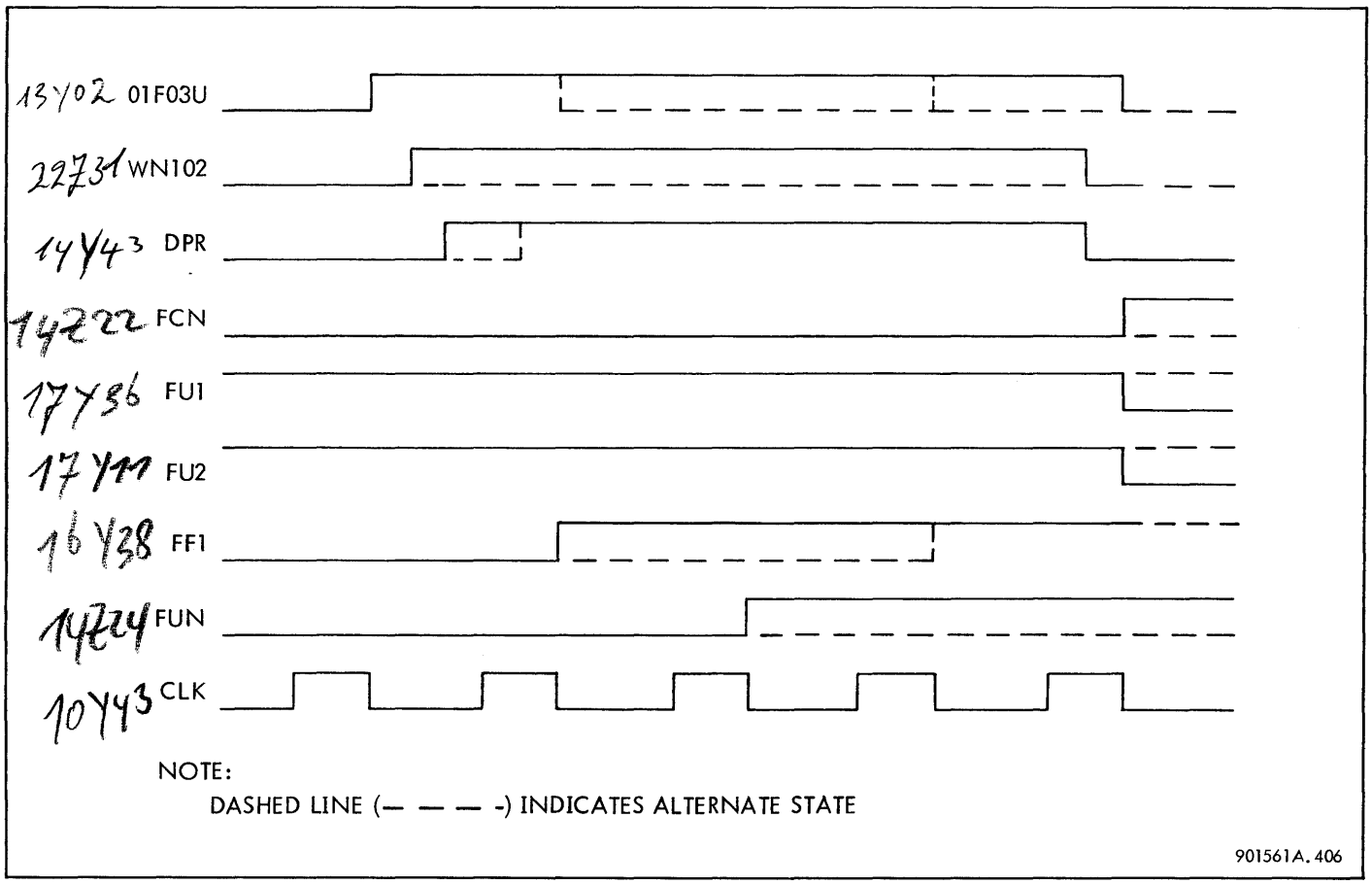


Figure 4-11. 01F03U State (Wait for Device Proceed), Timing Diagram

When DPR comes true in the controller, it enables the set and the clock inputs of flip-flop FCN and the clock inputs of flip-flops FU1 and FU2.

S/FCN = SFCN

SFCN = CFCNX + . . .

CFCNX = BAND19 + . . .

BAND19 = 01F 03U DPR WN102

C/FCN = CFCNX CLK

C/FU1 = CFU1

CFU1 = CFU1Z CLK + . . .

CFU1Z = BAND19 + . . .

C/FU2 = CFU2

CFU2 = CFU2Y CLK + . . .

CFU2Y = BAND19 + . . .

When CLK goes false, it clocks and sets FCN and resets FU1 and FU2. The controller is advanced to state 01F00U (order input) on a rewind order. When the station detects the beginning of tape marker (BOT), it ends the rewind operation and transmits an interrupt to the controller. The controller acknowledges the interrupt and is made aware that the station is again ready for service.

4-16 REWIND OFFLINE. When the rewind order is initiated, the controller raises WN102.

WN102 = WN2 + . . .

WN2 = F01 NF03 F04

The execution of the rewind offline order in the controller is the same as for the rewind online with the interrupt order. The difference is in the station, which switches to the manual mode at the start of rewind.

4-17 Pre-Data Transfer Phase

If a rewind order is not initiated and if flip-flop FCTS is set, the station enters a delay state to ensure the attainment of proper tape speed and to establish the correct gap between records. From this delay state, the station enters a ready state

and raises DPR in the controller to indicate it is ready to begin a data transfer operation.

When DPR comes true it enables the set input of flip-flop FF1.

S/FF1 = NFF1 SFF1

SFF1 = DPRNWN + . . .

DPRNWN = DPR NWN102

C/FF1 = CLK + . . .

When CLK goes false, it clocks and sets FF1, and the controller is advanced to state 03F03U (data transfer).

The controller can also be advanced to state 03F03U in an alternate manner. If signals TTSH, DCA, AIOR, and AUT are not true, flip-flop FUN is clocked and set when CLK goes false.

S/FUN = SFUN

SFUN = CFUNX + . . .

CFUNX = 01F03U BAND29 + . . .

BAND29 = NOR01 NAUT

C/FUN = CFUNX CLK + . . .

The set input to flip-flop FF1 is enabled when FUN is set, and the next time that CLK goes false, it clocks and sets FF1. The controller is advanced to state 03F03U.

S/FF1 = NFF1 SFF1

SFF1 = FUN + . . .

4-18 03F03U (DATA TRANSFER)

The controller advances to state 03F03U and enters one of three possible phases: read, write, space file or record.

During the service cycles, the controller advances to state 03F02U and returns to state 03F03U between data byte transfers from the buffer (fast) memory (FM). This operation ensures that a byte transfer cannot occur until the present FM cycle is completed.

4-19 03F03U-02U (DATA TRANSFER)

As explained in paragraph 4-18, the data transfer operations alternate between states 03F03U and 03F02U. State 03F03U-02U is the combination of the two.

03F03U = 03F 03U

03F = FF1 FF2

03F02U = 03F 02U

02U = FU1 NFU2

See figure 4-12 for a flow diagram of state 03F03U-02U.

4-20 Read Phase

The controller enters the read phase when WRT and F01 are false and a read order is present. Paragraphs 4-21 through 4-82 contain a detailed explanation of the read data flow through the controller, and figure 4-13 is a flow diagram of state 03F03U-02U read phase.

4-21 READ CIRCUITRY AND DATA PATH. The controller receives read input data from the station electronics over nine individual data lines. The data is in bytes, and each byte consists of eight data bits and one parity bit. The nine bits are transferred in parallel from the station to the controller over the nine data lines.

The station electronics contains the read head, the read amplifiers, and the buffer amplifiers. The output of each buffer amplifier connects to the input of a peak detector counter in the controller. The controller contains nine individual read channels and each read channel times, shifts, compares and finally submits the information read to the IOP. See figure 4-14 for a block diagram of the read data flow.

4-22 CLOCK OPERATION. The clock circuit consists of a crystal oscillator, flip-flops to divide the oscillator output in half and in half again, and of associated logic circuitry.

The clock circuit generates an accurate and stable reference signal that is used as a master clock throughout the various counter stages in the controller.

The clock circuit oscillator uses a 3.840 MHz crystal to control accurately the frequency and the stability of the master clock signal. The oscillator output (OSCLO) is connected through a buffer amplifier (CLB) to the clock input of flip-flops CD1 and CD2 which divide the oscillator output in half and in half again. See figure 4-15 for a logic diagram of the clock operation.

S/CD1 = NCD1

R/CD1 = CD1

C/CD1 = CLB

CLB = OSCLO

OSCLO = Output of crystal oscillator

S/CD2 = NCD1 NCD2

R/CD2 = NCD1

C/CD2 = CLB

CLOCK = OSCLO 150 SEL
+ CD1 75 SEL
+ CD2 37.5 SEL

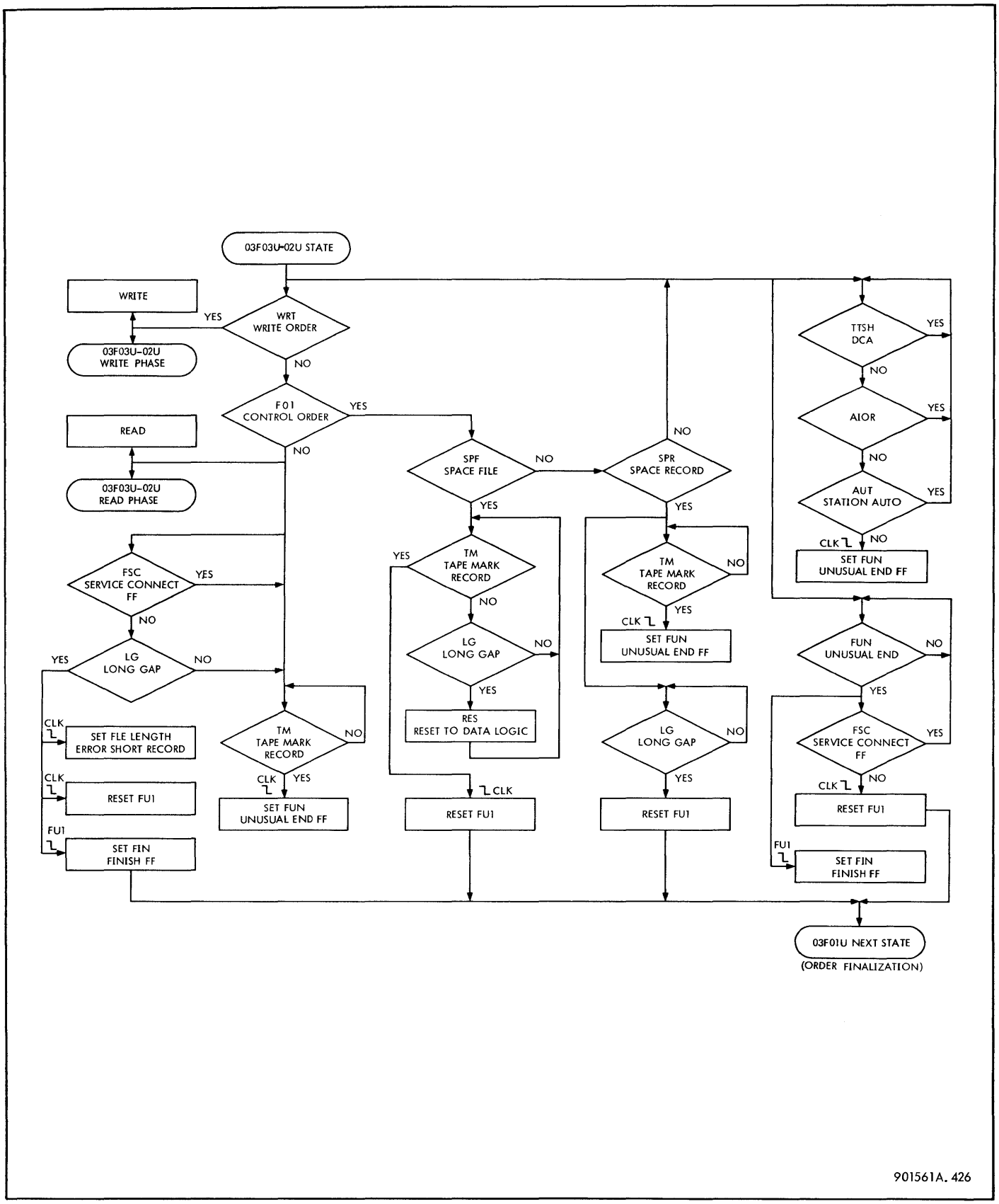


Figure 4-12. 03F03U-02U State (Data Transfer), Flow Diagram

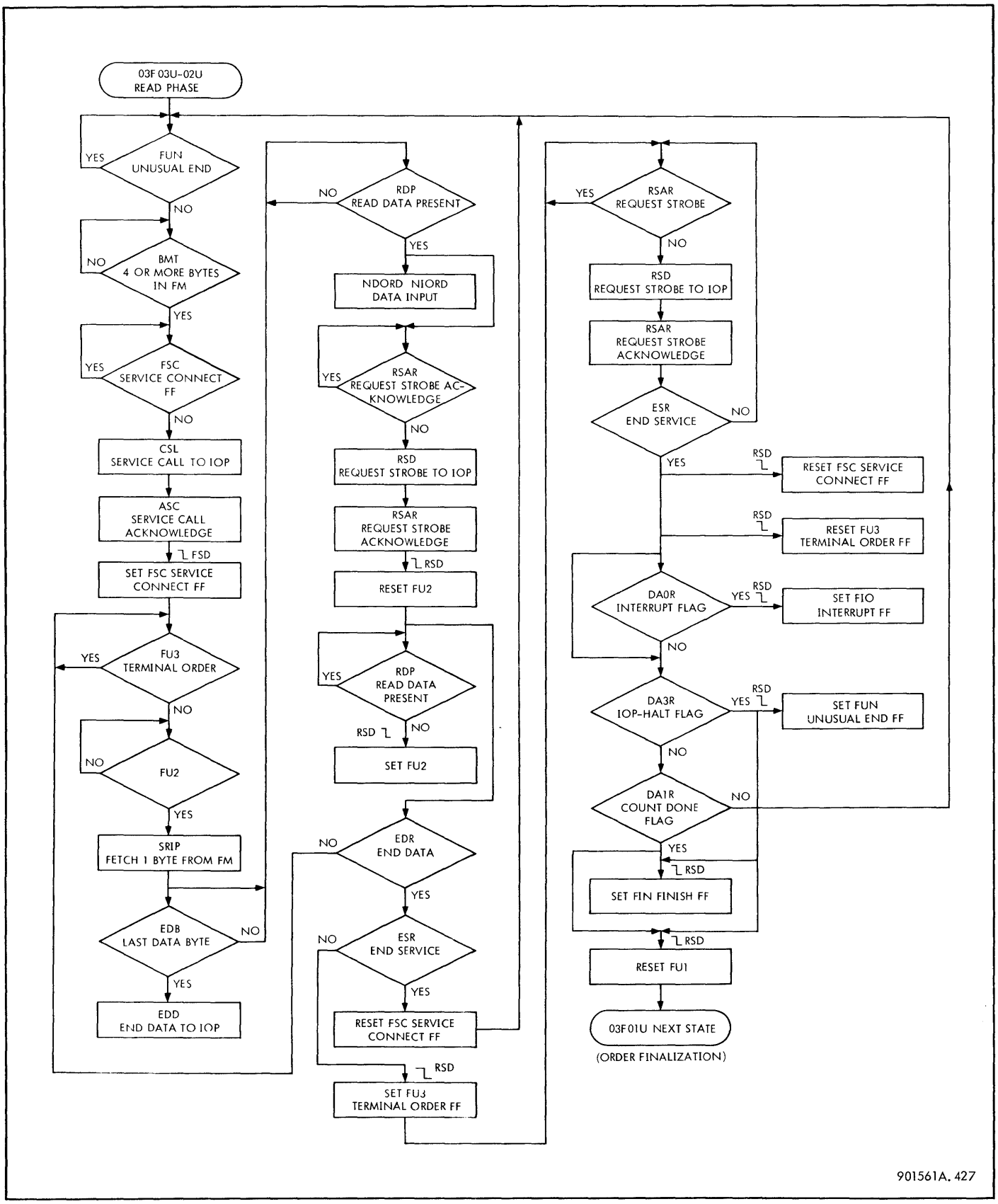


Figure 4-13. 03F03U-02U State (Read), Flow Diagram

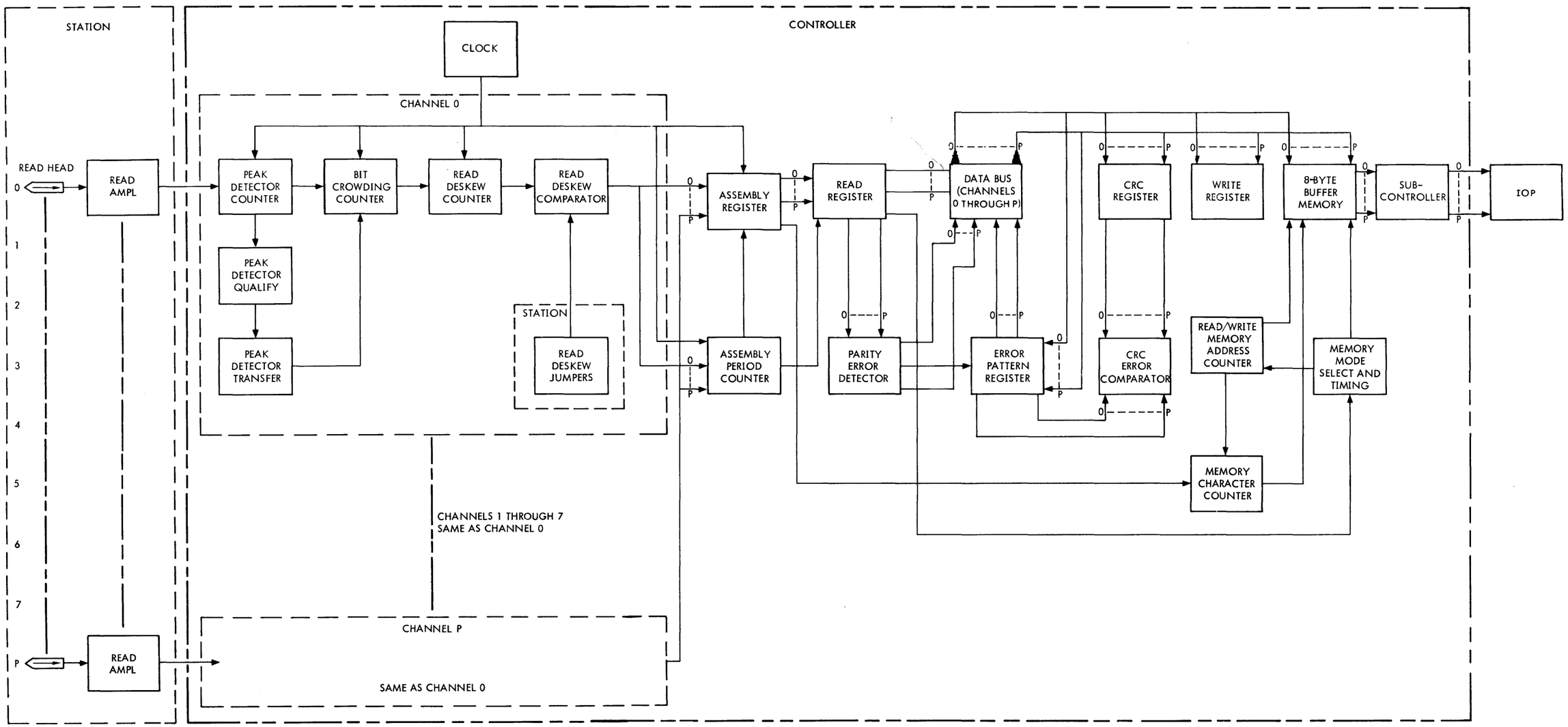


Figure 4-14. Read Data Flow, Block Diagram

901561A.430

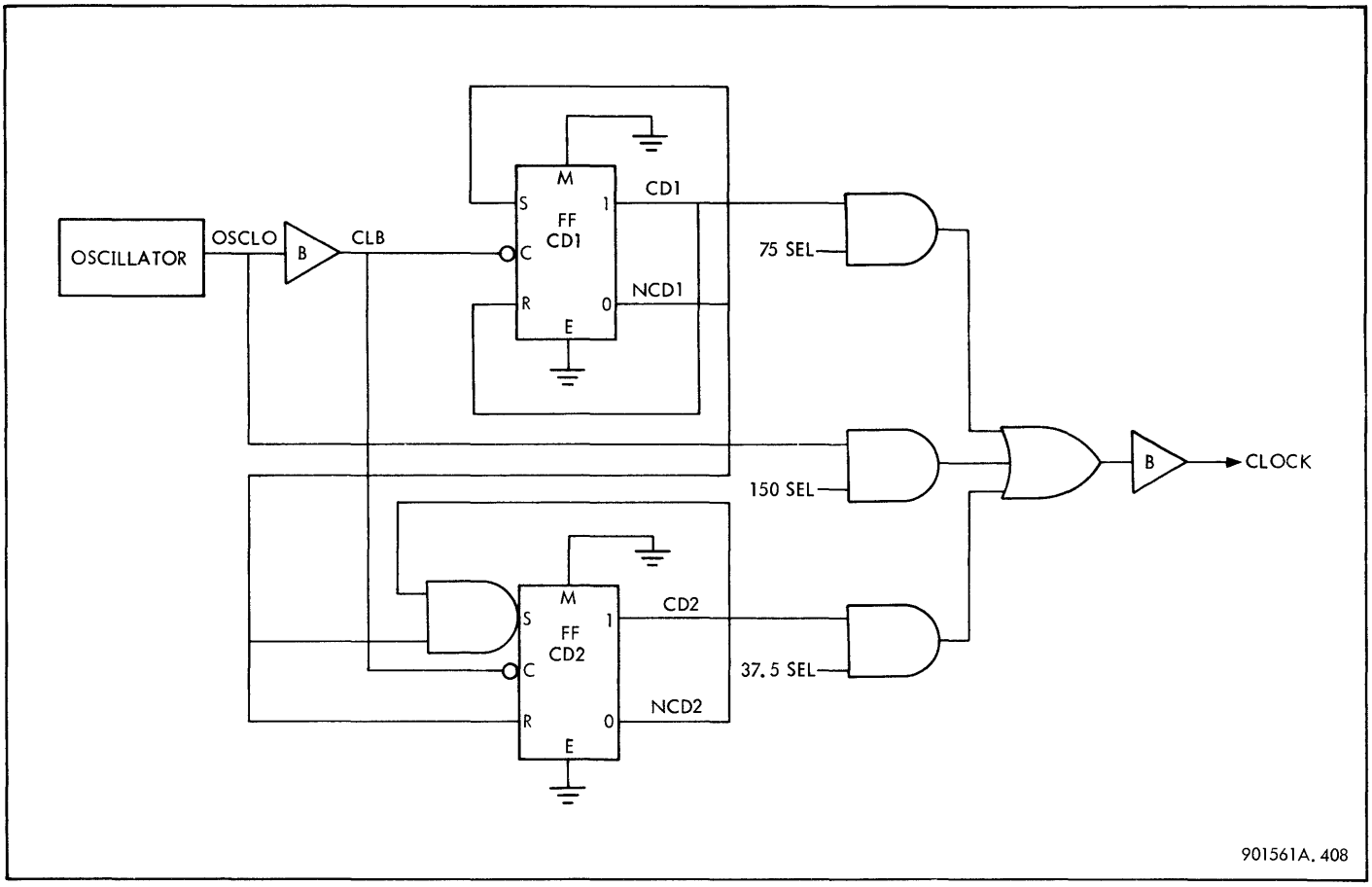


Figure 4-15. Clock Circuit, Simplified Logic Diagram

The frequency of the CLOCK signal is determined by the type of station used with the controller. Enable signal 150 SEL from the station is always true in a 150 ips (Model 7323) station, and signal lines 75 SEL and 37.5 SEL are hardwired to ground to disable them. CLOCK is a 3.840 MHz (4 MHz) squarewave.

In a 75 ips (Model 7322) station enable signal 75 SEL is true, and 150 SEL and 37.5 SEL are hard-wired to ground to disable them. CLOCK is a 1.920 MHz (2 MHz) squarewave.

Although the controller can generate a CLOCK signal for a 37.5 ips station, it is not used at this time with such a station, and 37.5 SEL is always disabled.

4-23 PEAK DETECTOR COUNTER OPERATION. The peak detector counter is a six-stage binary counter and receives its input from the station electronics. When input signal RDAMP from the station electronics goes true, indicating a one has been read on the tape, the counter begins its count cycle. It counts as long as RDAMP is true or until it reaches a count of 64. At that time, all stages are reset. (Then it starts the count again if the input is still true.) See figure 4-16 for a simplified logic diagram of the peak detector counter.

The counter is modified by terms PDQ and PDT. PDQ (peak detector qualify) is a qualifying term that allows the counter to complete its count operation only when the input pulse width exceeds a certain predetermined limit. This rejects any short duration noise pulses that could be mistaken for valid data bits. PDT (peak detector transfer) is the term that transfers the data from the counter to the following stage.

The counter is reset to zero by REPKDET before a data bit is accepted.

$$\text{REPKDET} = \text{NRDAMP NPDQ}$$

The above equation and those that follow apply equally to any of the nine read channels.

The flip-flops used in this counter, and in other controller circuits, are of the SDS T-series and "set overrides reset" type. This means that the set input always overrides the reset input when both are true simultaneously. The reset inputs of all the flip-flops in the peak detector counter are wired true on the circuit card, and the reset output of each stage is wired back to its set input. This creates a simple toggle flip-flop for each stage of the counter.

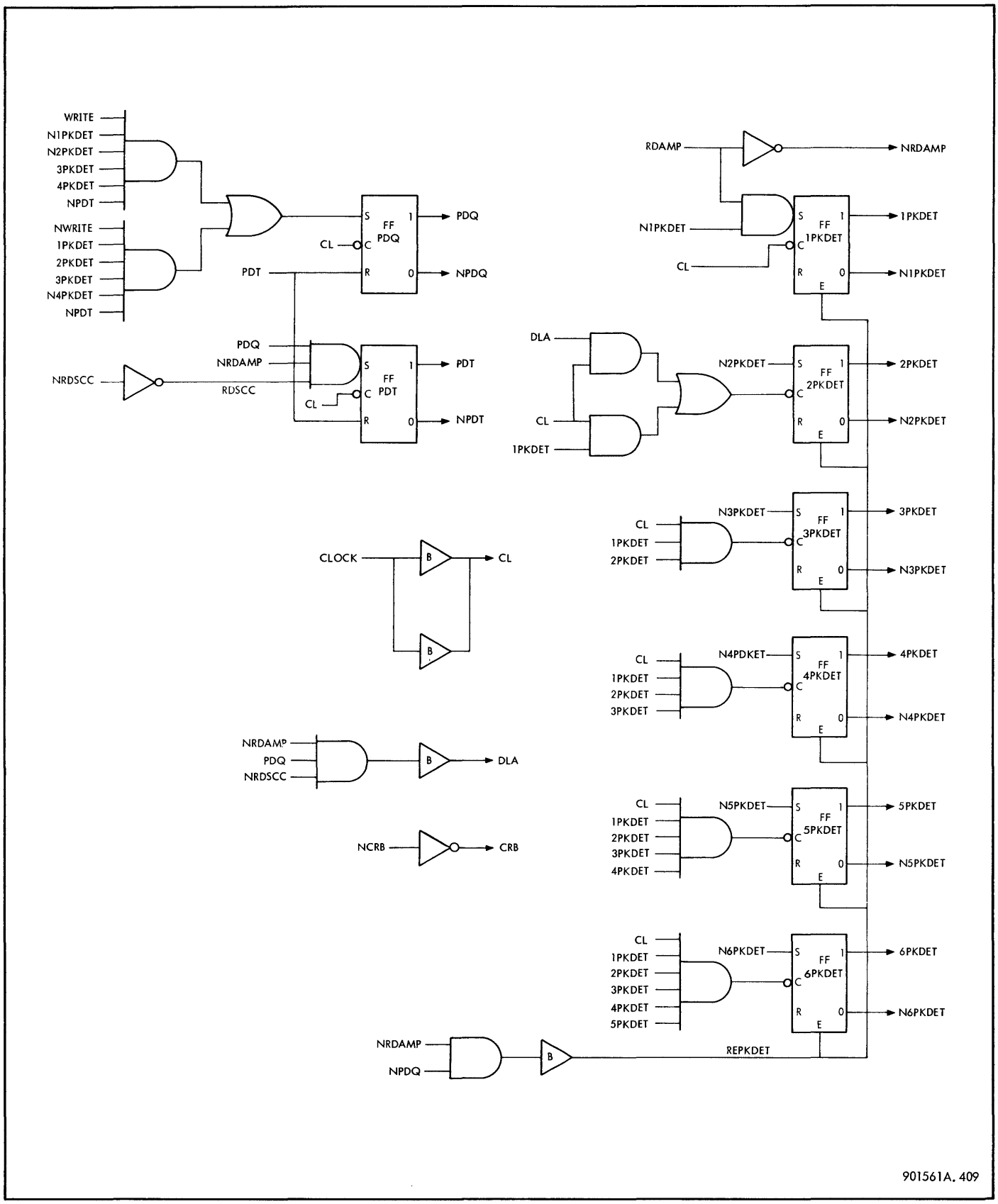


Figure 4-16. Peak Detector Counter, Simplified Logic Diagram

CLOCK is used to trigger the flip-flops and is connected to the CLOCK input of each of the nine peak detector counters in the controller. The CLOCK input is connected to two buffer amplifiers in parallel in each counter. The output of the amplifiers (CL) is AND-gated at the clock inputs of each flip-flop with qualifying terms from the other flip-flops. When CL goes false, the flip-flops in the counter are clocked.

When the input signal from the station electronics goes true (RDAMP), it causes the set input for the first stage in the counter (1PKDET) to go true.

S/1PKDET = RDAMP N1PKDET (LSB)

C/1PKDET = CL

The first stage of the counter clocks and sets the next time CL goes false. The set output of the first stage is AND-gated with CL at the clock input of the second stage. The second clock signal then resets the first stage, since its set input is now held false by N1PKDET and sets the second stage. If RDAMP were true long enough, the counter would advance in a binary manner until it reached a full count of 64; then if RDAMP were still true, it would reset and start counting again.

Assume that, instead of a normal data bit, a noise spike of short duration has been detected. The counter starts counting when the noise spike goes true. Now assume that the counter has reached an arbitrary count of six and that the noise spike goes false. A count of six is not sufficient to cause flip-flop PDQ (a pulse width detector) to set; therefore, its reset output (NPDQ) is still true.

S/PDQ = NWRITE 1PKDET 2PKDET 3PKDET N4PKDET NPDT + . . .

C/PDQ = CL

(For a read process, the equation indicates, that the set input of PDQ requires a count of seven in the register in order to be true. Then PDQ sets on the eighth clock signal.) PDQ must be set for the counter to continue its count cycle. NPDQ remains true since the noise spike was of insufficient duration to set PDQ. The dc-reset input for all stages in the counter is:

REPKDET = NRDAMP NPDQ

NPDQ is true, and when the noise spike goes false, NRDAMP goes true and resets all flip-flops in the counter.

Assume that a normal data bit has caused RDAMP to go true. The counter begins its count when the first clock signal following the true state of RDAMP goes false (see figure 4-17). The first clock signal sets 1PKDET; the second clock resets 1PKDET and sets 2PKDET and so on. On the eighth clock signal PDQ sets. The counter continues to count until RDAMP goes false. Assume that RDAMP goes false between the 15th and 16th clock signals. The count in the register at the time that RDAMP goes false is 001111 (15). The set input for PDT is now true.

S/PDT = PDQ NRDAMP RDSCC

RDSCC = N1RDSC N2RDSC N3RDSC N4RDSC N5RDSC

C/PDT = CL

RDSCC is true when the bit crowding counter is reset.

At the next clock signal, PDT and 5PKDET set, and 1PKDET through 4PKDET reset.

S/5PKDET = N5PKDET

C/5PKDET = CL 1PKDET 2PKDET 3PKDET 4PKDET

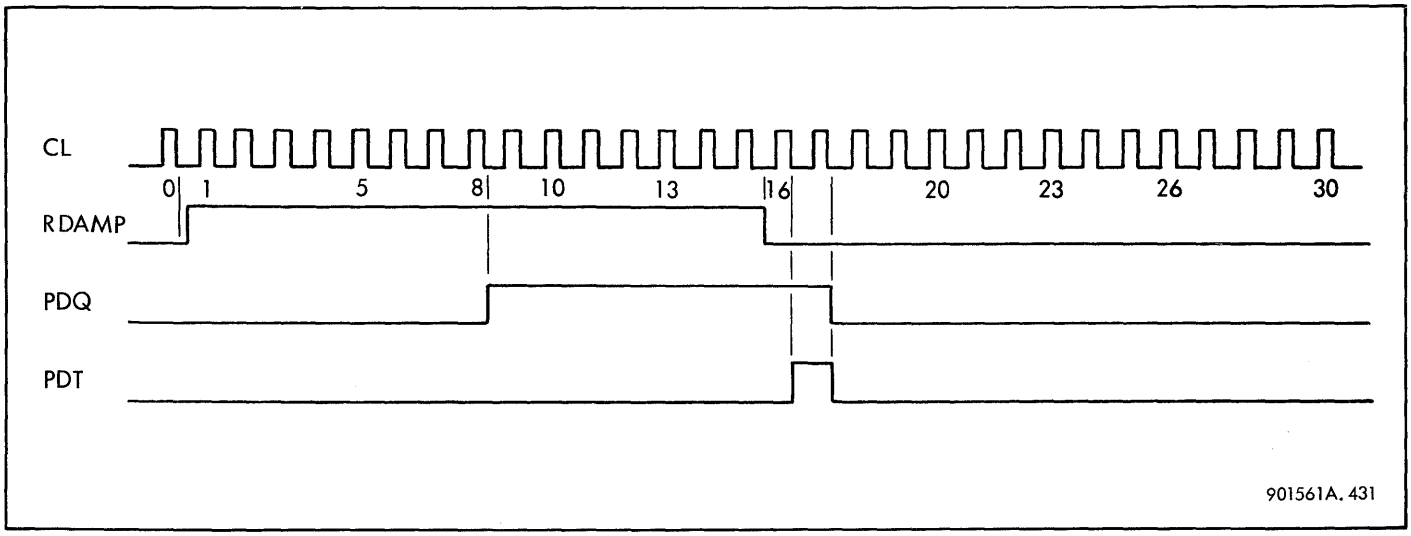


Figure 4-17. PDQ and PDT (Read), Timing Diagram

This gives a count of 01000 (16) in the register. Output PDT is connected to the reset inputs of PDQ and PDT.

R/PDQ = PDT

R/PDT = PDT

When PDT goes true on the 16th clock signal, it is gated directly into the dc-set inputs of the following bit crowding counter. One of its stages is set, and RDSCC is no longer true, allowing the set input of PDT to go false. The next clock signal (17th) then resets both PDT and PDQ. NPDQ is AND-gated with NRDAMP to generate dc-reset signal REPKDET. When NPDQ goes true, all stages in the counter are dc-reset.

4-24 BIT CROWDING COUNTER OPERATION. The bit crowding counter is a five-stage binary counter that at PDT time begins its count at a value equal to one-half that of the peak detector counter value. In this manner, the bit crowding counter effectively starts at the peak of input signal RDAMP. Timewise, the peak is located in the center of signal RDAMP.

The input signals for the bit crowding counter are obtained directly from the peak detector counter. Each flip-flop set output in the peak detector counter (with the exception of the least significant bit, 1PKDET) is AND-gated with PDT at the dc-set input of a particular flip-flop in the bit crowding counter. By deleting the LSB of the peak detector counter, the count is effectively divided in half.

The flip-flops in the bit crowding counter are wired in toggle fashion, with their reset outputs connected to their set inputs. The first stage, 1RDSC, has an additional control signal (NRDSCC) connected to its set input.

Signal CLOCK provides the clock signal (CL) for the flip-flops in the counter. CL is the same clock signal used in the peak detector counter and is AND-gated at the clock inputs of each flip-flop with NPDT and qualifying terms from the other flip-flops. All flip-flops in the counter are initially reset to zero. See figure 4-18 for a simplified logic diagram of the bit crowding counter.

The bit crowding counter timing diagram, figure 4-19, shows that PDT comes true at the 16th clock signal. PDT is AND-gated at the dc-set input of each flip-flop in the counter with an output from the peak detector counter.

M/1RDSC = PDT 2PKDET

M/2RDSC = PDT 3PKDET

M/3RDSC = PDT 4PKDET

M/4RDSC = PDT 5PKDET

M/5RDSC = PDT 6PKDET (MSB)

The output of the peak detector counter at the 16th clock is 010000 (16). When PDT comes true, flip-flop 4RDSC is set (see table 4-2). This gives a count of 01000 (8) in the bit crowding counter and starts its count at half that of the peak detector counter.

When output 4RDSC goes true signal NRDSCC is no longer false, since a flip-flop in the counter is now set.

NRDSCC = 4RDSC + . . .

When NRDSCC comes true, it causes the set input of flip-flop 1RDSC to go true.

S/1RDSC = NRDSCC N1RDSC

Clock signal CL is AND-gated with NPDT at the clock input of 1RDSC.

D/1RDSC = CL NPDT

When NPDT comes true, the next clock signal sets 1RDSC. The second clock signal then resets 1RDSC and sets 2RDSC and so forth.

S/2RDSC = N2RDSC

C/2RDSC = CL 1RDSC NPDT

The count continues in a normal binary manner until the 32nd count. At this point, all flip-flops in the counter are reset. When all the flip-flops are reset, NRDSCC goes false. Since NRDSCC is the control term for the set input of 1RDSC, the first stage in the counter cannot set, and the counter remains reset.

4-25 READ DESKEW COUNTER OPERATION. The read deskew counter, a three-stage binary counter, starts its count operation when the count in the bit crowding counter reaches a certain number, and controls the read deskew comparator circuits by the count it develops in its own register.

Table 4-2. Bit Crowding Counter Operation

PEAK DETECTOR COUNTER		BIT CROWDING COUNTER	
State	Term	Term	State
0	6PKDET → 5RDSC		0 (MSB)
1	5PKDET → 4RDSC		1
0	4PKDET → 3RDSC		0
0	3PKDET → 2RDSC		0
0	2PKDET → 1RDSC		0
0	1PKDET		

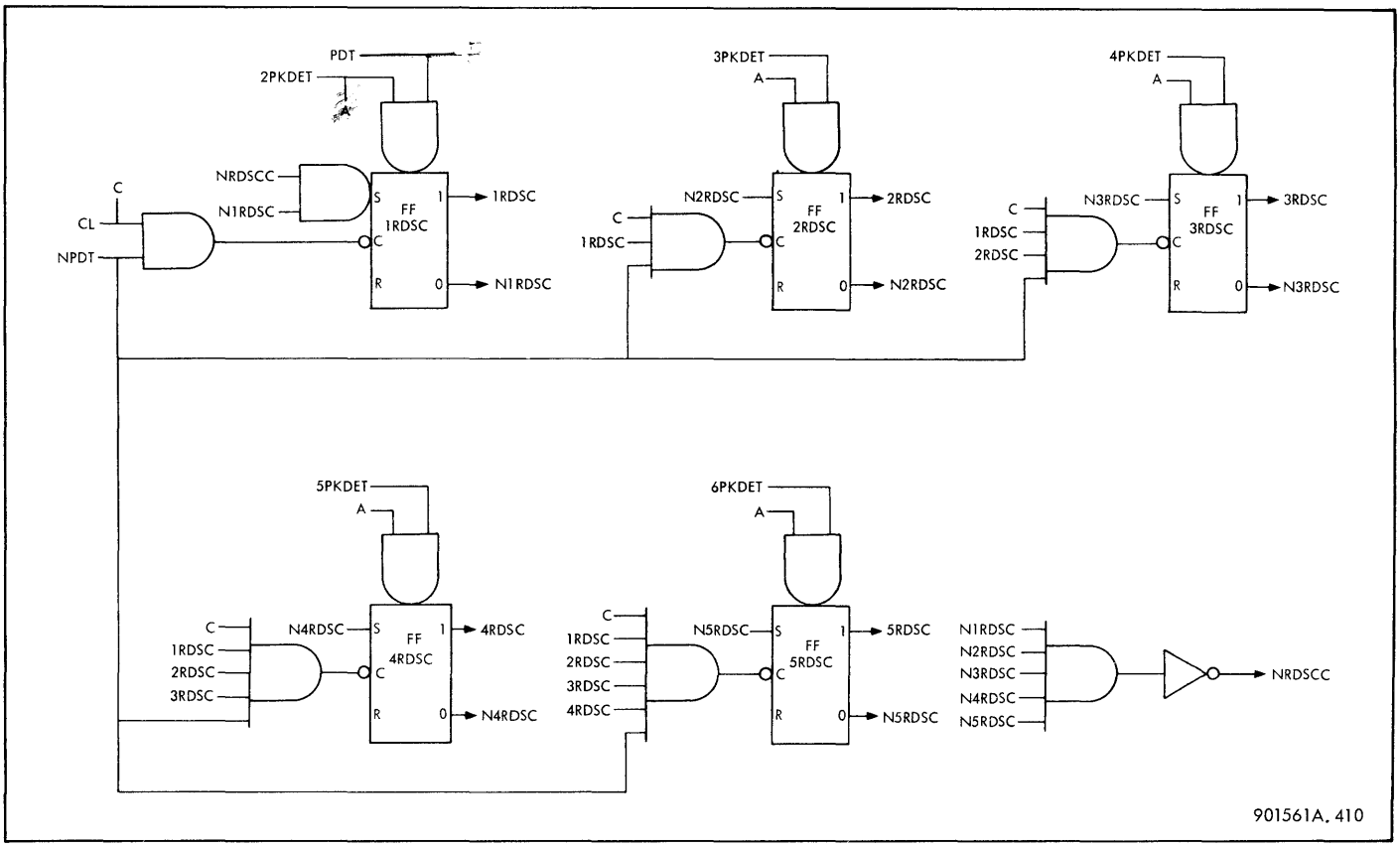


Figure 4-18. Bit Crowding Counter, Simplified Logic Diagram

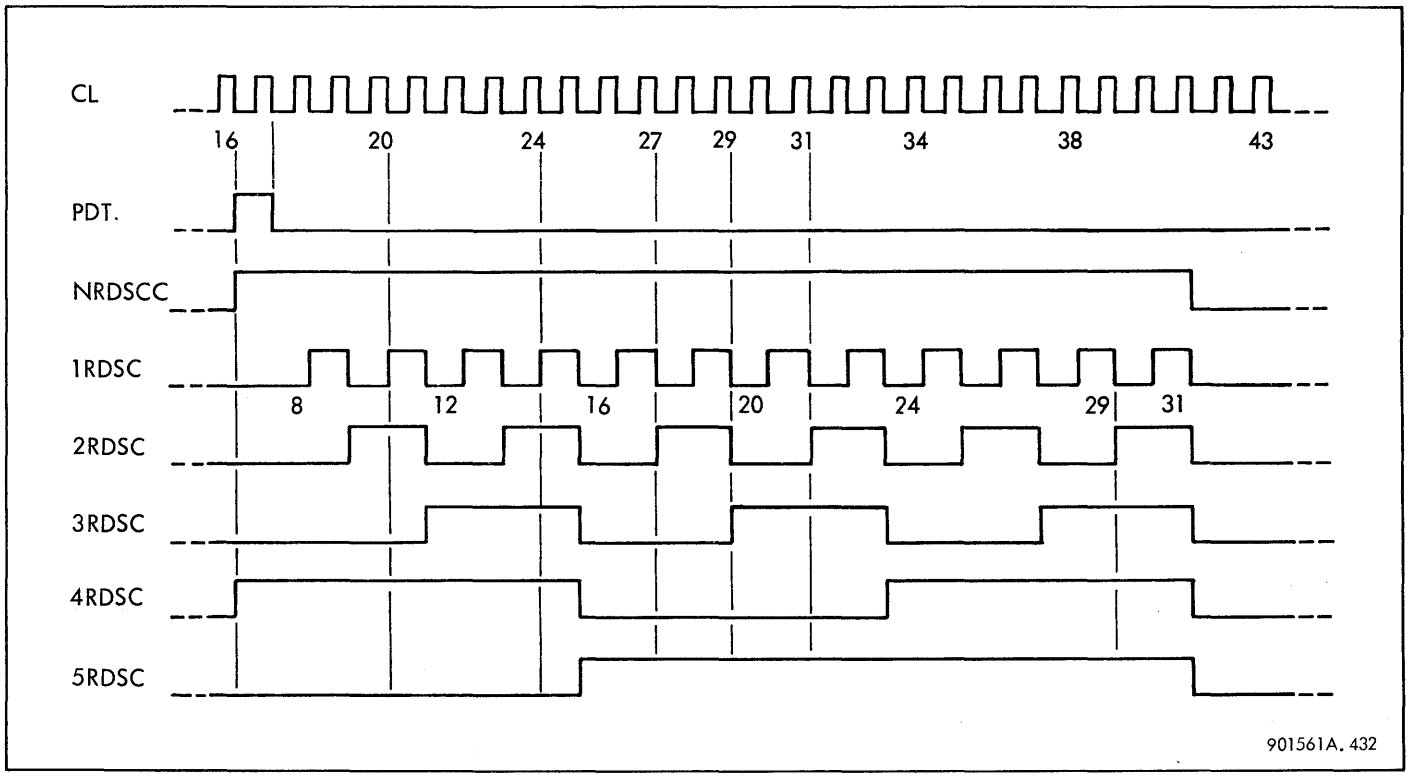


Figure 4-19. Bit Crowding Counter, Timing Diagram

The input signal for the read deskew counter is taken directly from the bit crowding counter. Certain flip-flop outputs from the bit crowding counter are used if the previous bit in the read register was a one. Other flip-flop outputs are used if the previous bit was a zero. A control signal is generated from these inputs to start the read deskew counter operation.

The flip-flops in the counter are wired in a toggle configuration, with their reset outputs connected back to their set inputs. The first stage in the counter, 1BCC, has additional control signals gated into its set input. The flip-flops in the counter are initially reset to zero.

Signal CLOCK provides the clock signal (CL) for the flip-flops in the counter. CL is the same clock signal used in the peak detector counter and is connected directly to the clock input of the first stage in the counter. CL is AND-gated with qualifying terms at the second and third stages.

If the preceding bit in the track (stored in the read register) is a zero, the output of the read register (line RR) is false. RR is inverted to generate NCORR, which is true at this time.

NCORR = NRR

NCORR is AND-gated with flip-flop outputs from the bit crowding counter and is inverted to generate signal NCRB. NCRB is also inverted and CRB is the control signal used to start the read deskew counter.

CRB = NCORR 1RDSC 2RDSC 3RDSC 4RDSC 5RDSC + ...

The read deskew counter starts its count operation when the count in the bit crowding counter reaches 31.

If the preceding bit in the read register is a one, the output of the read register (line RR) is true. NCORR is then false. NCORR is inverted to generate signal CORR, which is true at this time. CORR is AND-gated with flip-flop outputs from the bit crowding counter and is inverted to generate signal NCRB. NCRB is inverted to generate signal CRB.

CORR = RR

CRB = CORR 1RDSC N2RDSC 3RDSC 4RDSC 5RDSC + ...

The read deskew counter starts its count operation when the count in the bit crowding counter reaches 29. See figure 4-20 for a simplified logic diagram of the deskew counter.

Assume that the preceding bit in the read register was a one. CORR is made true, and CRB comes true when the count in the bit crowding counter is 29. CRB is AND-gated with other control signals at the set input of the first stage in the counter.

S/1BCC = N1BCC CRB + ...

C/1BCC = CL

The next time that CL goes false, flip-flop 1BCC is clocked and is set. The next clock signal resets 1BCC and sets 2BCC. The counter continues to count in a binary manner until it reaches a count of 111 (7); then the next clock signal resets the counter. Since the bit crowding counter has already reset previous to this count, CRB is false, and the set input of 1BCC is held false. The counter then remains reset.

4-26 READ DESKEW COMPARATOR OPERATION. The read deskew comparator selects the period required to correct read skew error in a read channel. Since the skew errors may differ, depending on whether the record is being read in a forward or a reverse direction, provisions have been incorporated to correct both forward and reverse skew.

Part of the comparator circuitry is physically located in the station electronics, and the remainder is located in the magnetic tape controller.

The read deskew comparator receives its input directly from the three flip-flops in the read deskew counter and from preset comparison circuitry in the station electronics, which contains the deskew jumpers. The output of the comparator is an exclusive OR function of the combined deskew jumpers and the deskew counter outputs. When the preset count, established by the jumpers, and the count in the deskew counter are complementary, the comparator output is true. See figure 4-21 for a simplified logic diagram of the read deskew comparator.

Assume that the tape unit is reading in a forward direction and that the center forward jumper in the station electronics (figure 4-21) has been removed. This gives a preset count of 010 in the station circuitry. In order to have a comparison (that is, RBD true), the read deskew counter must have a complementary count of 101.

RBD = (1BCC NDJAC + ...)
(N2BCC DJBC + ...)
(3BCC NDJCC + ...)

NVRSS and RVSS are the forward and the reverse control signals, respectively, for the preset comparison circuitry. When reading forward, NVRSS is true and RVSS is false; when reading backward, RVSS is true and NVRSS is false.

By removing the center jumper, DJBS is made true.

DJBS = RVSS NRVSS

(Removing the jumper is the same as making the input true.) The two other gates, DJAS and DJCS are false, since

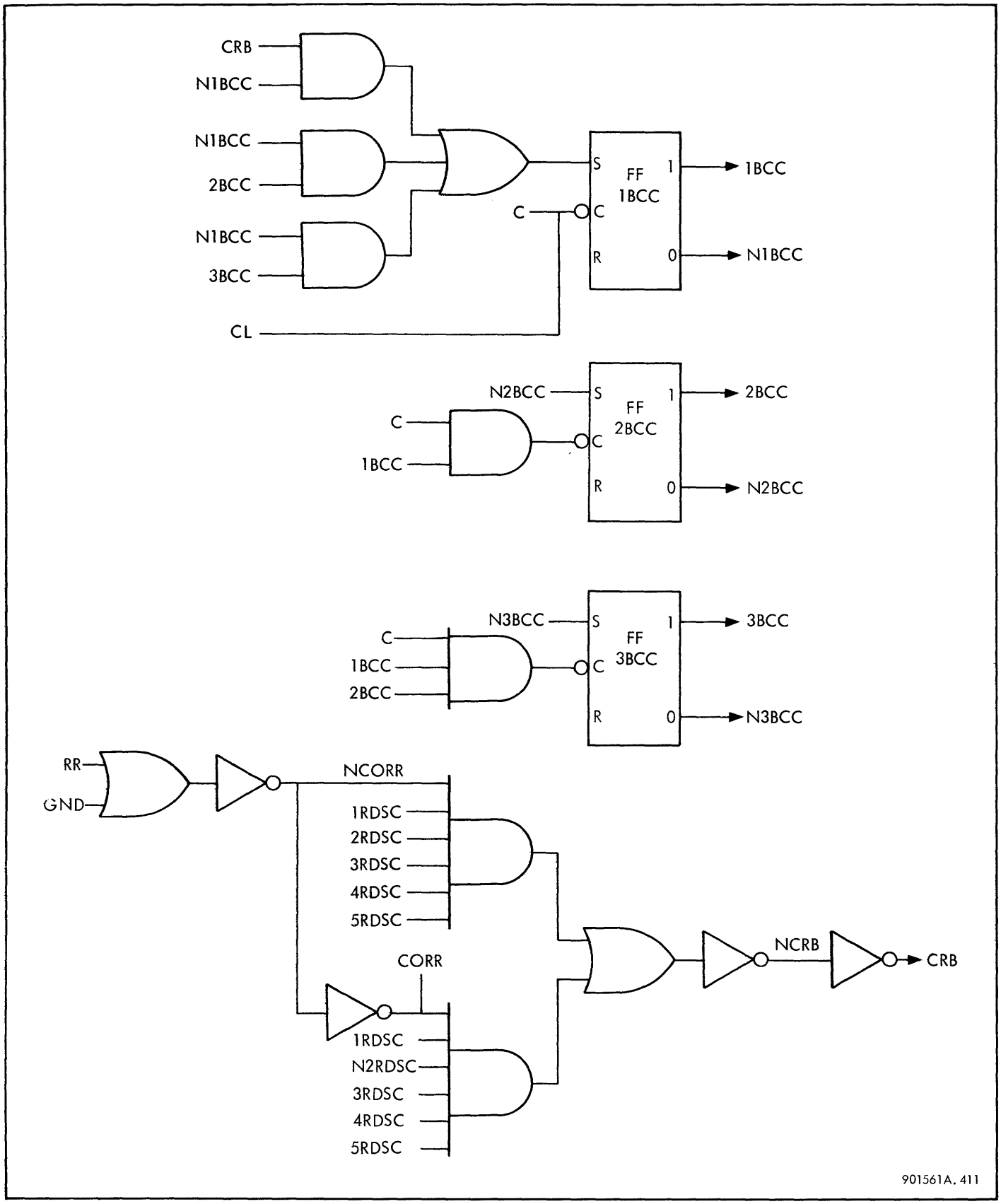


Figure 4-20. Deskew Counter, Simplified Logic Diagram

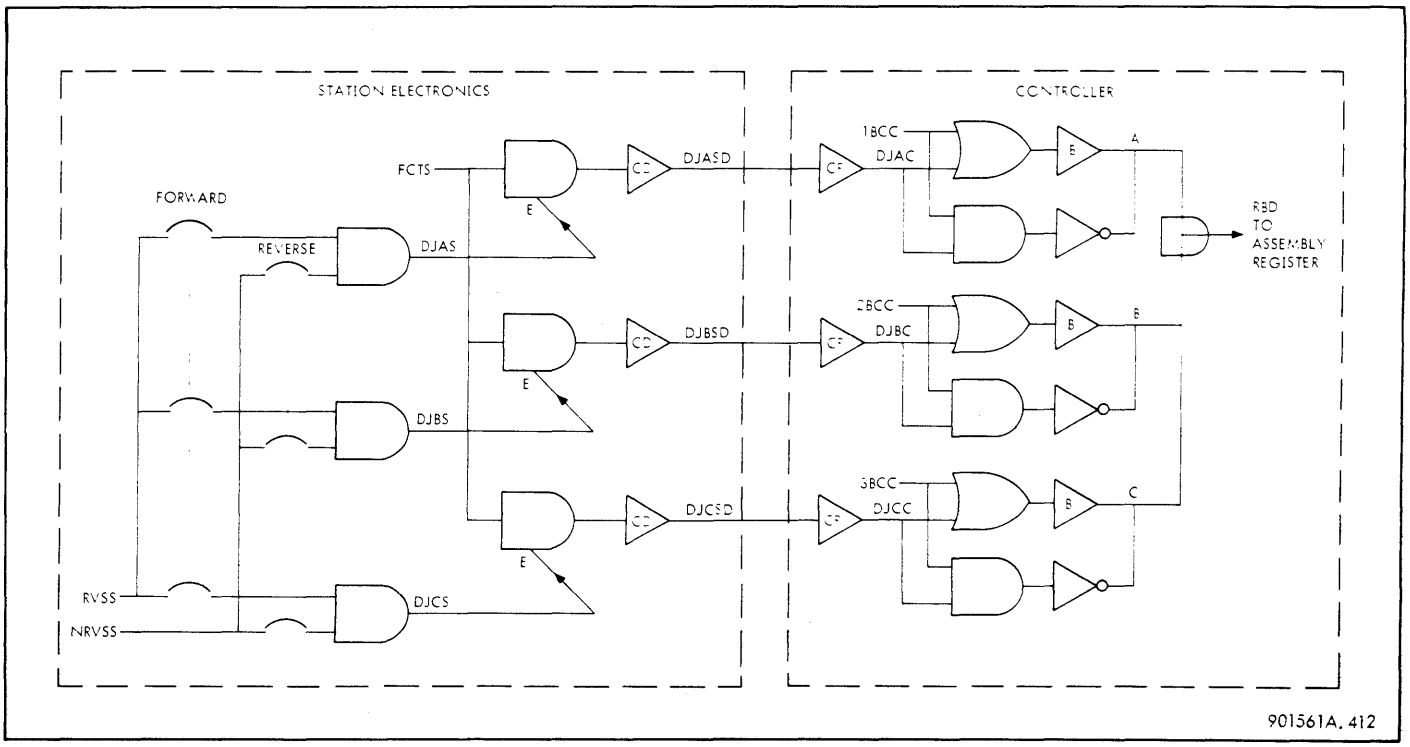


Figure 4-21. Read Deskew Comparator, Simplified Logic Diagram

RVSS is false. The output of the station electronics is as follows:

DJASD = FCTS DJAS

DJAS = RVSS NRVSS

DJBSD = FCTS DJBS

DJCSD = FCTS DJCS

DJCS = RVSS NRVSS

FCTS = The set output of device connect flip-flop FCTS, which is true when the station is connected.

If the count in the deskew counter is the complement of the preset count, points A, B, and C are true, and, consequently, RBD is true. (Signal RBD is used as an input to the assembly register.) If 1BCC is equal to DJAC, point A and RBD are false, indicating no comparison. A false output from any one of the three buffer-inverter combinations drops the entire line and causes RBD to be false. Therefore, it is necessary for the three outputs to be true for RBD to be true.

The outputs of the nine deskew comparators also provide initiate count signals to the assembly period counter. Comparator outputs RBD0, 1, 2, and 7 are inverted and are connected in a dot-OR configuration to form signal NRBDC1.

Outputs RBD3, 4, 5, 6, and P are inverted and are connected in a dot-OR configuration to form signal NRBDC2.

NRBDC1 = NRBD0
NRBD1
NRBD2
NRBD7

NRBDC2 = NRBD3
NRBD4
NRBD5
NRBD6
NRBDP

Signals NRBDC1 and NRBDC2 are AND-gated and are inverted to form signal RBDC.

RBDC = N(NRBDC1 NRBDC2)

Signal RBDC which goes true when any comparator output is true initiates the count in the assembly period counter with NRBDC1 and NRBDC2. See figure 4-22 for a simplified logic diagram of this comparator function.

The operation of the read deskew comparator is exactly the same for a read backward operation as for a read forward. Since the control signals are reversed and because the deskew

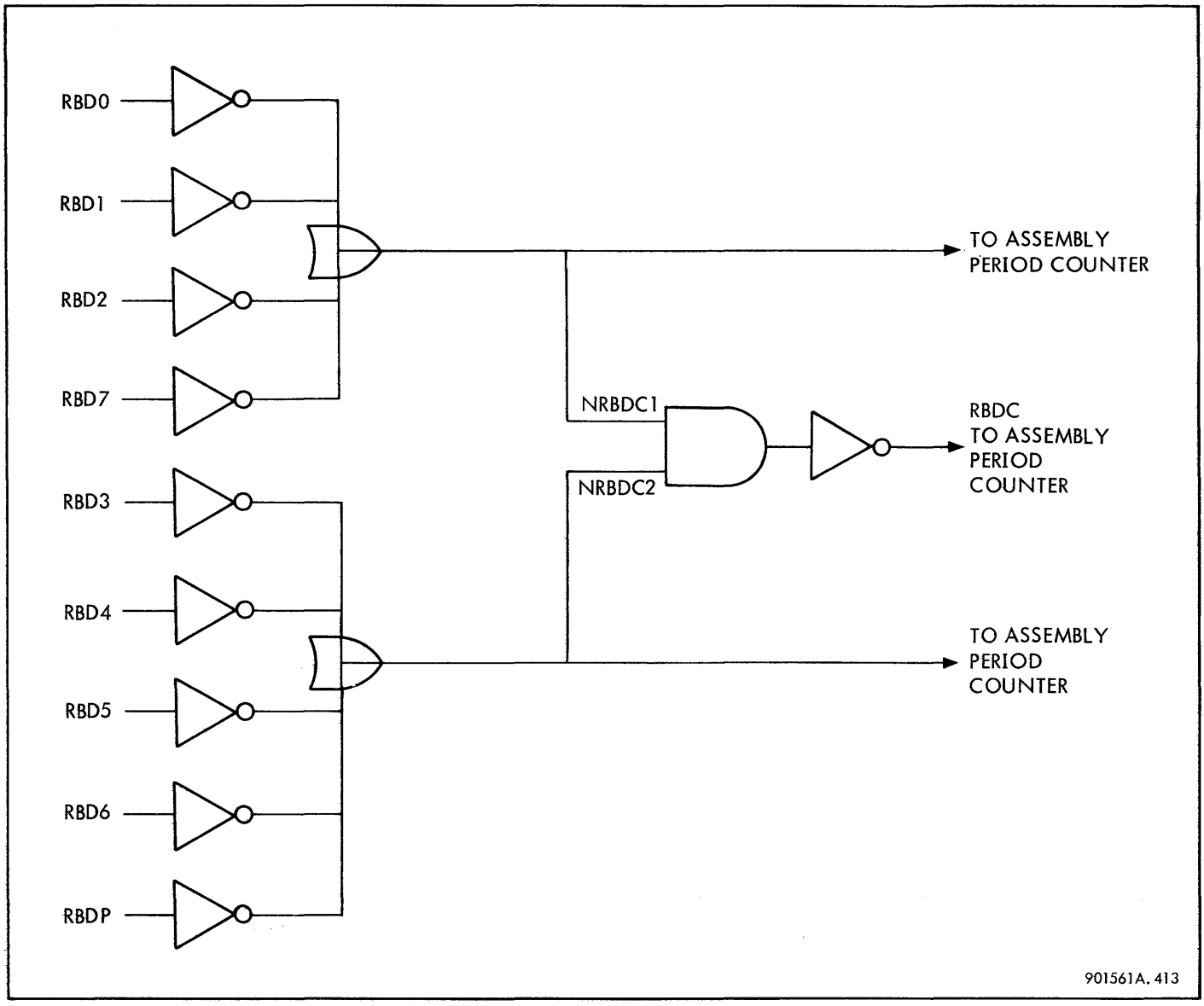


Figure 4-22. RBDC Generation, Simplified Logic Diagram

time might be different in a read backward operation, another set of jumpers is used for this operation.

4-27 ASSEMBLY PERIOD COUNTER. The assembly period counter is primarily an eight-stage binary counter that specifies the period during which bits in a character are accepted from the read deskew comparator by the assembly and read registers.

The nine read channel outputs from the read deskew comparators are connected as signals NRBDC1, NRBDC2, and RBDC, to the input of the assembly period counter. When the first bit from any one of the nine read channels arrives, the assembly period counter starts its timing sequence. As the bits in the character arrive, they are also presented to the assembly register inputs and are clocked into the register.

When the assembly period counter reaches a predetermined count, it generates a signal that causes the read register to accept all bits of the character that are in the assembly register at that time.

Immediately following this data transfer, the assembly period counter generates another signal that causes the assembly register to reset. The register is then ready to accept the next character. After the last character in a word is read and is stored in the read register, the assembly period counter generates timing and control signals SG (short gap) and LG (long gap).

In the assembly period counter, there are two control flip-flops, eight toggle flip-flops in the binary counter itself, logic circuitry that generates the end of assembly period

signals, and the circuitry that generates the short and long gap signals. The control flip-flops and the first six stages of the counter are initially reset; the last two stages of the counter are initially set.

The clock signal (CLA) for the flip-flops is derived from signal CLOCK. CLA is connected directly to the clock input of the control flip-flops and the first stage in the counter and is AND-gated with qualifying terms at the second through sixth stages of the counter. A clock signal is not used for the seventh and eighth stages of the counter. The seventh stage is clocked directly by the set output of the sixth stage, and the eighth stage is clocked directly by the reset output of the seventh stage.

The outputs from the various stages of the counter are gated to generate control signals for the end of assembly period signals and for the short gap and long gap signals. See figure 4-23 for a simplified logic diagram and figure 4-24 for a timing diagram of the assembly period counter.

4-28 Assembly Period Counter Operation, Read Forward. Control flip-flop APCC1 is initially dc-reset.

E/APCC1 = RES3

RES3 = RES (reset to data logic)

RES = NTM•LG•SPF + 01F03U NDPR

Control flip-flop APCC2 is initially clocked and reset.

C/APCC2 = CLA

R/APCC2 = APCC2

When flip-flop APCC2 is reset, it generates the initial dc-reset signal (REAPC) for the first six stages of the counter.

REAPC = N(APCC2) NAPCC2

E/APC1-APC6 = REAPC

Flip-flop APCC2 also provides the initial dc-set input for the last two stages of the counter.

M/APC7-APC8 = NAPCC2

Counter input RBDC is normally false and goes true on the first one data bit that appears on any one of the nine input data lines. When RBDC goes true, signifying the arrival of a character, it enables the set input of the first control flip-flop, which is clocked and set the next time CLA goes false. The flip-flop remains set until EAP (end of assembly period) goes true.

S/APCC1 = RBDC NAPCC1

C/APCC1 = CLA

Input RBDC goes false soon after APCC1 is set. The set output of APCC1 enables the set the next time CLA goes false.

S/APCC2 = APCC1 NRES + . . .

NRES = Inverted output of RES1 (false at this time)

C/APCC2 = CLA

The set output of APCC2 enables the set input of the first stage of the binary counter, and APC1 is clocked and set the next time that CLA goes false.

S/APC1 = APCC2 NAPC1

C/APC1 = CLA

The counter now operates as a normal binary counter, and when a predetermined number is reached, EAP comes true.

EAP = NWRITE APC2 APC3 APC4 NAPC5 NAPC6 APC7 APC8

EAP1, the read register clock input, is raised the next time that CLA comes true.

EAP1 = EAP CLA NAPC1

When CLA goes false, APCC1 is clocked and reset, EAP1 goes false, and EAP2 comes true.

R/APCC1 = EAP

EAP2 = EAP APC1

When EAP1 goes false, it clocks and sets into the read register flip-flops the input data stored in the assembly register at that time.

EAP2 is the reset input for the assembly register flip-flops. The next time that the assembly register clocks go false, the flip-flops are reset, and the register is ready to accept the next character. In the assembly period counter, the resetting of APCC1 does not cause APCC2 to reset at this time because its set input is latched true.

S/APCC2 = NRES NRBDC1 NRBDC2 APCC2 NAPCZ NCZSG

NAPCZ = APC1 + . . .

NCZSG = APC8 + . . .

On the first bit of the next character, RBDC goes true, and NRBDC1 and NRBDC2 go false. The next time that CLA goes false, it clocks and sets APCC1 and resets APCC2. When APCC2 resets, initialize signals REAPC and NAPCC2 come true. The first six stages of the counter are dc-reset,

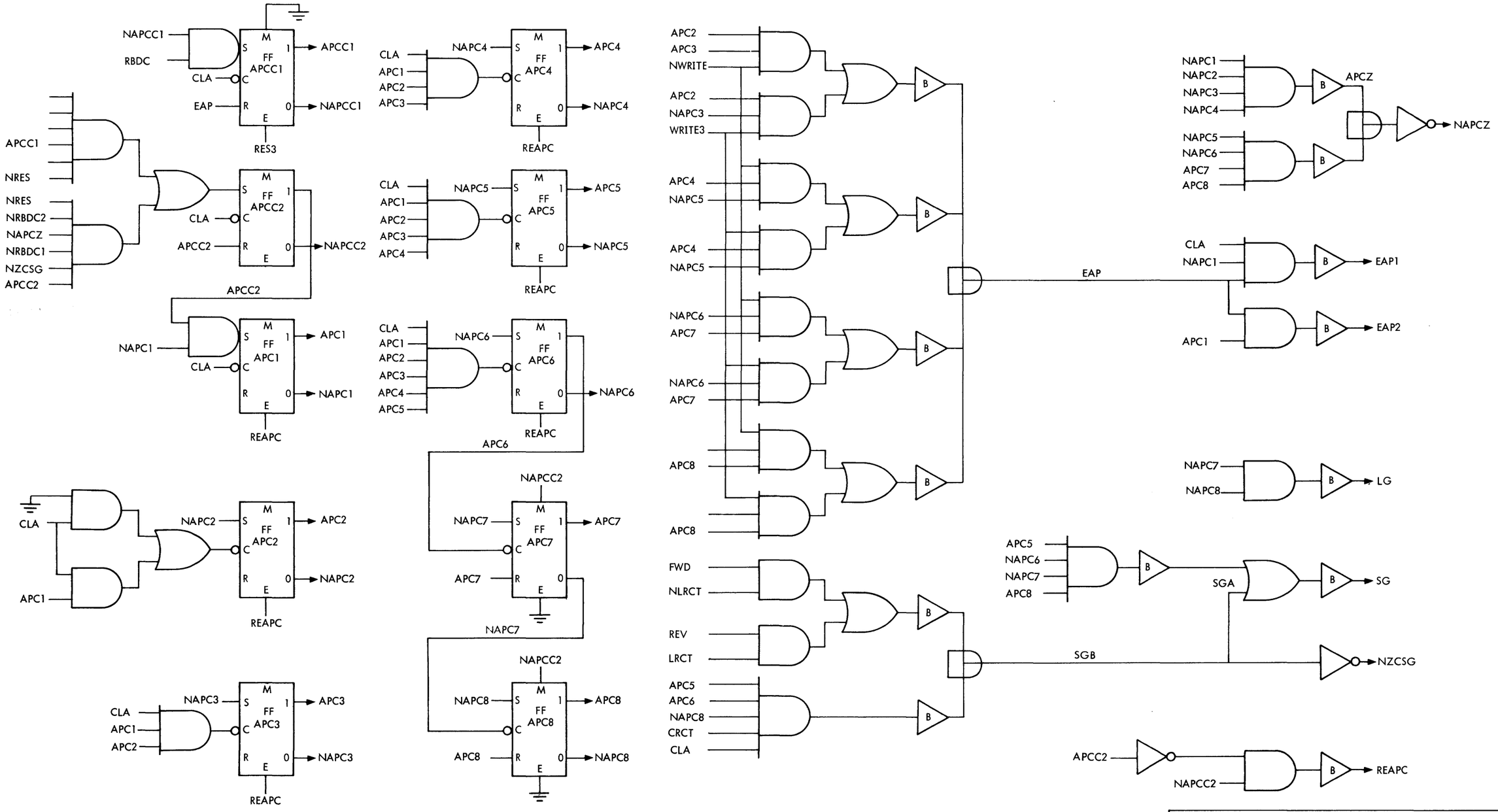


Figure 4-23. Assembly Period Counter, Simplified Logic Diagram

901561A.417

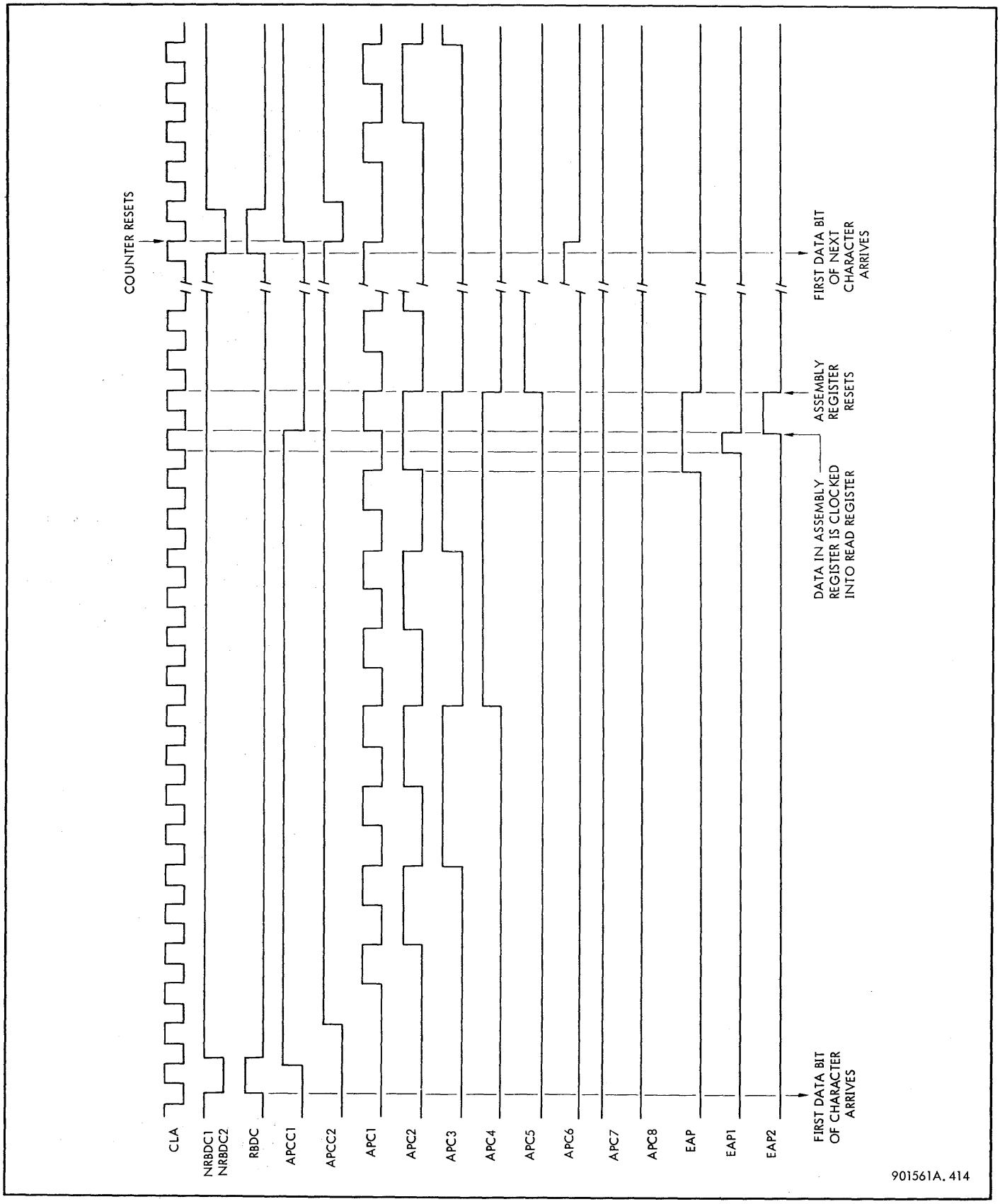


Figure 4-24. Assembly Period Counter (Read Forward), Timing Diagram

and the last two are dc-set, as previously described. On the fall of the next clock signal, the counter repeats the cycle just described, resets and repeats the cycle again at the first bit of each succeeding character.

After the last character in a record has been stored in the read register, the assembly period counter does not reset as previously described because the counter depends on the first one bit in a new character to initiate the reset process. If none is forthcoming, the counter continues to count. At a predetermined count, signal SGA comes true.

SGA = APC5 NAPC6 NAPC7 APC8

Signal SG (short gap) now comes true and is used as a control signal for the CRC and the LRC detect flip-flops which indicate when these characters should be read or validated.

SG = SGA + . . .

See figure 4-25 for a timing diagram of the CRC and LRC sequences.

SG remains true for one count and, when it goes false, clocks and sets flip-flop CRCT.

S/CRCT = NCRCT

C/CRCT = SG

The counter continues its count operation until the CRC character is read from the tape. When the CRC character is detected by RBDC, the counter resets and starts another count operation. It counts as before, and SG again comes true. SG remains true for one count, as before, and then goes false. Flip-flop CRCT is clocked and reset, and flip-flop LRCT is clocked and set.

S/LRCT = NLRCT

C/LRCT = FWD CRCT SG

The counter continues to count until the LRC character is detected. When this occurs, the counter resets and starts the count operation again. SG comes true for the third time; however, this time no other character appears, and signal LG (long gap) comes true.

LG = NAPC7 NAPC8

When the counter has completed its entire count cycle, LG goes false and signal APCZ comes true.

APCZ = NAPC1 NAPC2 NAPC3 NAPC4 NAPC5 NAPC6 APC7 APC8

When APCZ comes true, signal NAPCZ goes false, unlatching the set input of control flip-flop APCC2. APCC2 then resets the next time CLA goes false. The counter is reset and remains initialized until the next record.

4-29 Assembly Period Counter Operation, Read Forward With CRC Character All Zeros. The operation of the counter when the CRC character contains all zeros differs from the operation described in paragraph 4-33 only in the method used to reset the counter when the CRC character appears. See figure 4-26 for a timing diagram of the CRC = 0 operation. If the CRC character contains all zeros, RBDC does not come true when CRC is read. NRBDC1 and NRBDC2 remain true, keeping the set input of flip-flop APCC2 latched. Therefore, the counter does not reset at CRC time, as required, unless it provides a signal to unlatch and to reset APCC2. The counter generates SGB for this purpose.

At a count equal to three character spaces from the last character read, and corresponding to CRC time, SGB comes true.

SGB = FWD NLRCT + . . . (APC5 APC6 NAPC8 CRCT CLA)

When SGB comes true, SG also comes true, and NZCSG goes false.

SG = SGB + . . .

NZCSG = SGB inverted

NCZSG is a set input to APCC2, and, when it goes false, the set input is unlatched and APCC2 resets on the next clock time. The counter is reset and remains reset until the LRC character is detected. At this time, the counter starts its count operation again and operates as previously explained. SG and LG come true following the LRC character, and the counter is initialized when LG goes false.

4-30 Assembly Period Counter Operation, Tape Mark Record Detect. The counter operation during the tape mark record detection is similar to the CRC = 0 operation. See figure 4-27 for a timing diagram of the tape mark operation.

When the tape mark is detected between information files or after the last file on the tape, the counter is reset and another count operation is then started. The count continues until SGB comes true. As previously described, signal NZCSG goes false and the counter is reset and remains reset until the final LRC character is detected. When the counter resets, SGB goes false and flip-flop LTMD is clocked and set. LTMD indicates that it is time to read the LRC character that follows the tape mark.

S/LTMD = TMD

TMD = NRCP + . . . (TMD + . . .)

C/LTMD = SGB

When the LRC character is detected, the count operation starts again and SG comes true. The tape mark detect flip-flop, TM, is dc-set at this time.

M/TM = SG LTMD TMD + . . .

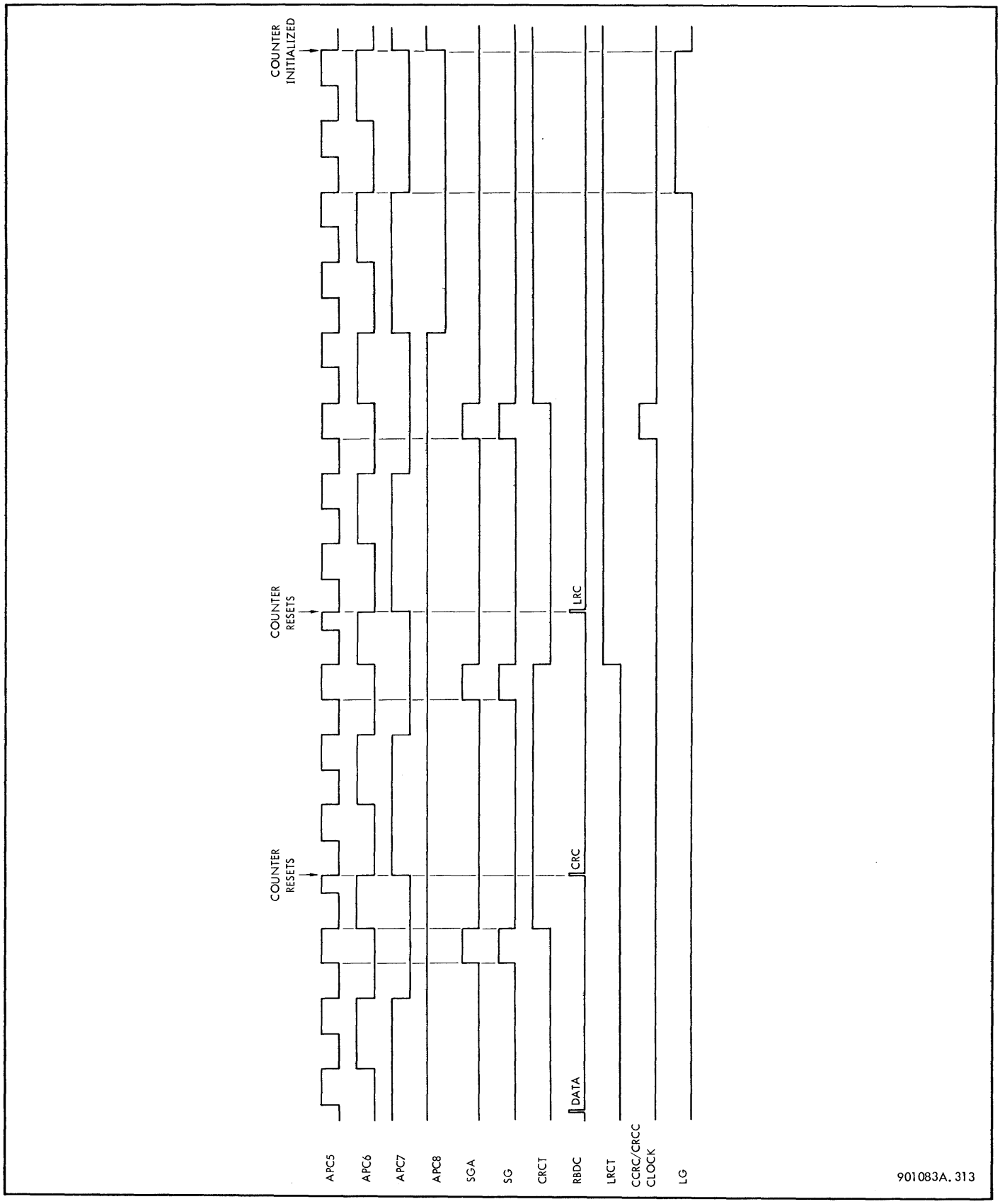


Figure 4-25. Assembly Period Counter (Read Forward, CRC $\neq$ 0), Timing Diagram

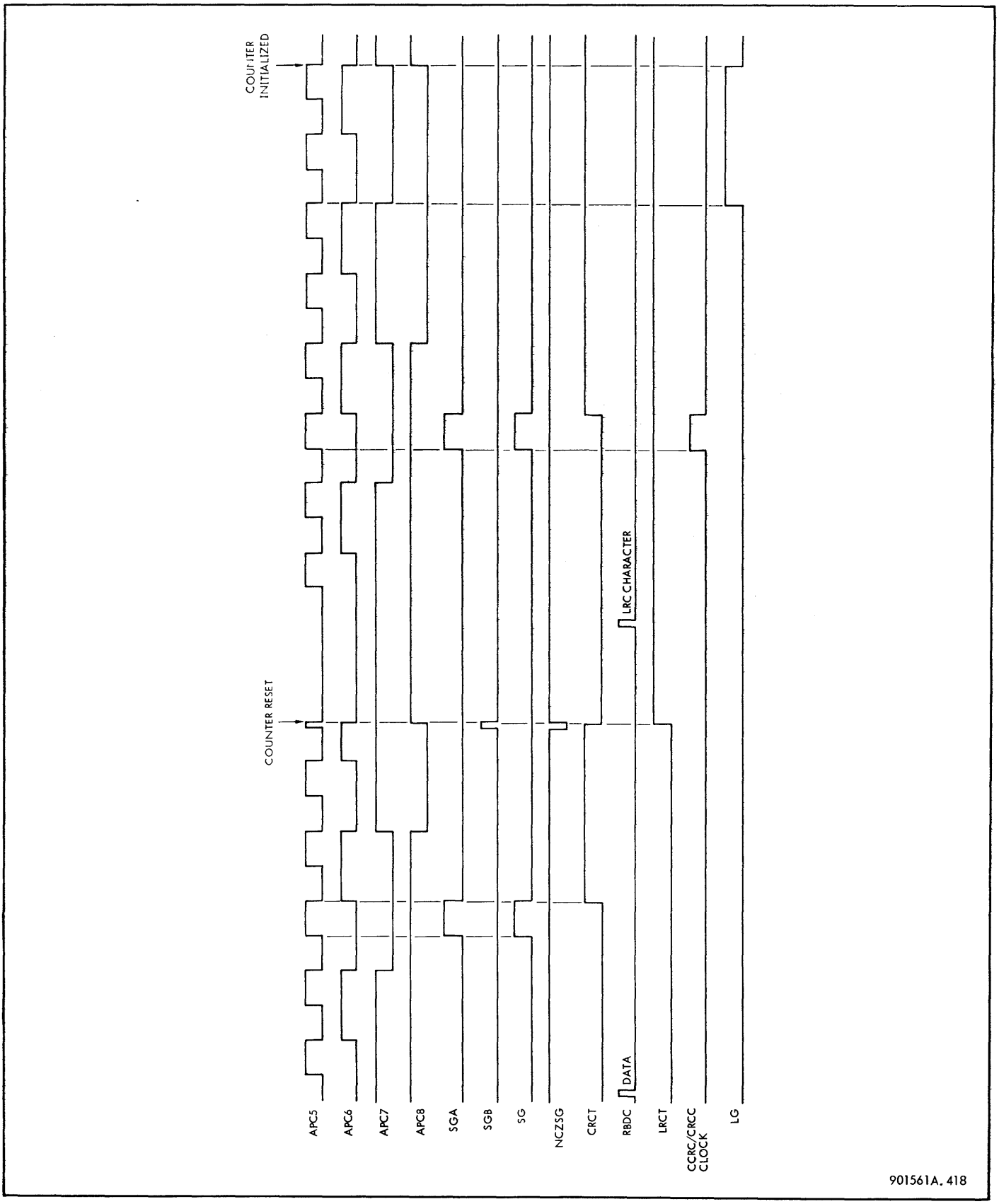


Figure 4-26. Assembly Period Counter (Read Forward, CRC = 0), Timing Diagram

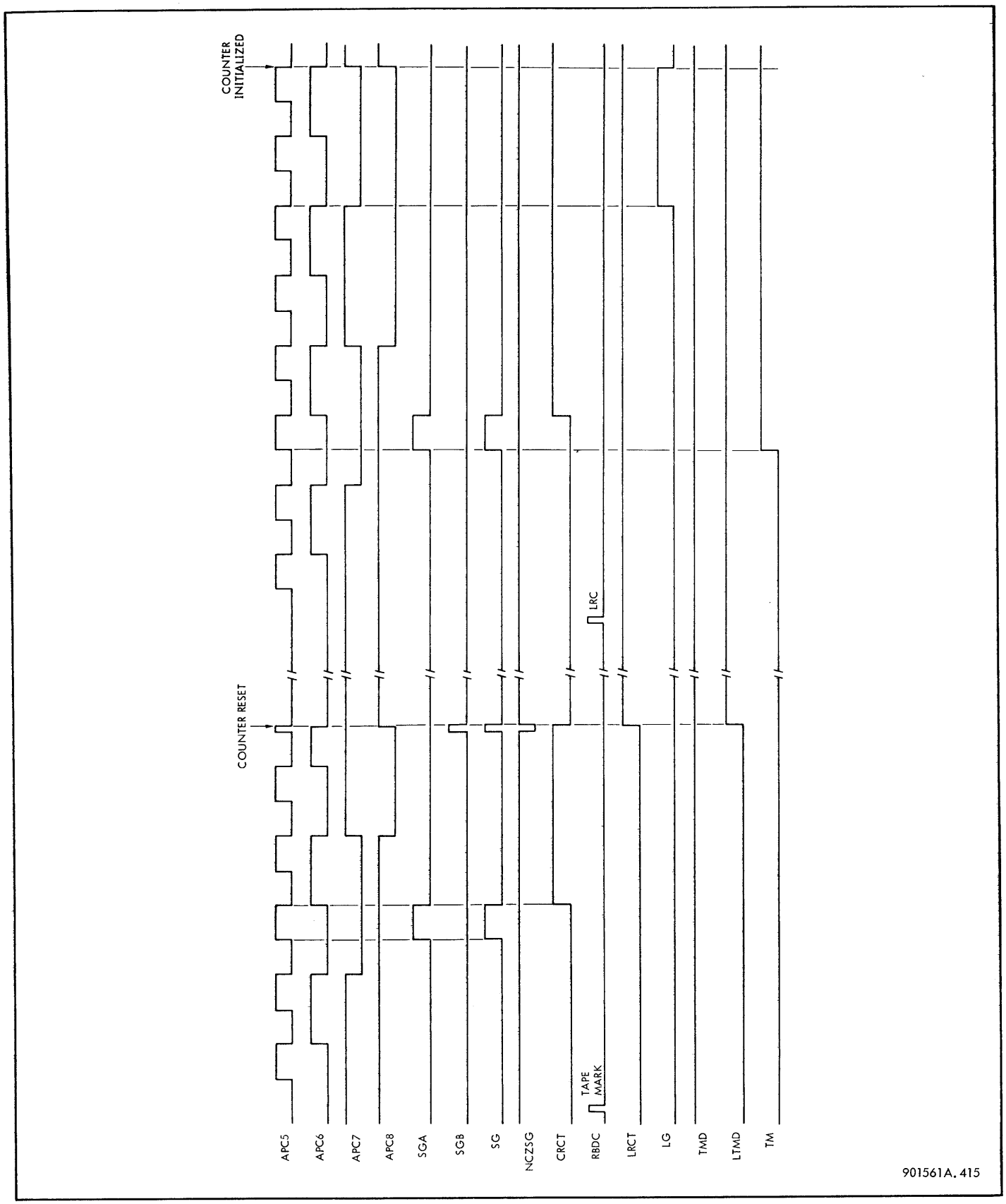


Figure 4-27. Assembly Period Counter (Tape Mark Detect), Timing Diagram

The counter operation continues, and LG comes true. When the count cycle is completed, LG goes false and the counter is initialized.

4-31 Assembly Period Counter Operation, Read Backward. The read backward operation of the assembly period counter is very similar to that of the read forward operation. It differs in that the LRC character is detected first, next the CRC character, then the last character recorded in the record, and so forth. See figure 4-28 for a timing diagram of a typical read backward operation with a normal CRC character.

The LRC character is detected first and starts the counter operation. The counter operates normally, and when the CRC character is detected, the counter is reset. The count operation starts again and continues until the last character recorded in the record is detected. The counter is again reset, and a new count cycle is started. The counter continues to operate in a normal manner as the record is read.

A read backward operation in which the CRC character is all zeros is also the reverse of the read forward operation. See figure 4-29 for a timing diagram of the CRC = 0 operation.

The LRC character is detected first and starts the counter operation. The counter operates normally, and when it is time to detect the CRC character, SGB comes true and NZCSG goes false. The counter is reset at this time and remains reset until the last character recorded in the record is detected. This starts another count cycle, and the counter continues to operate in a normal manner as the record is read.

4-32 ASSEMBLY AND READ REGISTERS. The assembly and the read registers each consist of nine individual flip-flops which store the data bits that make up the character that is read from the tape. See figure 4-30 for a simplified logic diagram of the assembly and the read registers.

Each of the nine read channel outputs from the read deskew comparators is connected to the set input of an assembly register flip-flop. The flip-flops in both assembly and read registers are initially dc-reset, and the data on the input lines is clocked into the assembly register by a clock signal derived from signal CLOCK. The set output of each assembly register flip-flop is connected to the set input of a read register flip-flop. At the end of the character assembly period, signal EAP1 from the assembly period counter clocks and stores in the read register the data that is in the assembly register. Signal EAP2 is the reset input for the assembly register. Signal EAP2 is also provided by the assembly period counter. The assembly register is clocked and reset and remains reset until the next character arrives.

In this system, the characters are recorded on tape using the NRZ method (see paragraph 3-4). Therefore, only one bits are actually recorded, because zero bits cause no change in tape flux. As a character is read, the corresponding input data line goes true for each track containing a one bit. The line stays false for a zero bit since no output is forthcoming from the read amplifier, and, subsequently, its read channel is inoperative. The output of the read deskew comparator for that channel is false, and the set input of the corresponding assembly register flip-flop is also false. When the assembly register is clocked, the flip-flop remains reset and its set output stays false. The set input of the read register flip-flop to which it is connected is also false. If the first character in the record is being read, then when the read register is clocked, any read register flip-flop with a false set input remains reset. The operation is the same for characters that follow because the flip-flop reset inputs are open and are always true. If the set input is still false, the flip-flop remains reset or resets if it had previously been set.

4-33 Assembly and Read Register Operation. The flip-flops in both registers are initially dc-reset.

E/ARx = RES3

E/RRx = RES3

Note

A lower-case x in an equation indicates that the equation can be applied to any of the nine tape channels.

When a character is read from the tape, the one bits of that character appear as true levels on the RBDx lines. The RBDx lines remain false if zero bits are present. During RBDx time CLx goes false and the assembly register flip-flops are clocked.

S/ARx = RBDx

C/ARx = CLx

The time period allotted for the character to be stored in the assembly register is established by the assembly period counter. EAP1 comes true, when the assembly period is finished. When EAP1 goes false, the data that is on the outputs of the assembly register flip-flops is clocked into the read register. If the assembly register flip-flop has been previously set, the read register flip-flop is also set.

S/RRx = ARx

C/RRx = EAP1

When EAP1 goes false, EAP2 comes true. EAP2 is the reset input to the assembly register flip-flops. The next time that CLx goes false, the assembly register is clocked and reset, and remains reset until the next character arrives. See figure 4-24 for the timing of EAP1 and EAP2.

R/ARx = EAP2

To illustrate further the operation of the assembly and the read registers, assume that the binary sequence 1101110 appears on a particular read channel as the characters are being read. Figure 4-31 shows how this number appears on the RBDx line and the timing of events that follow.

At the first one bit, RBDx comes true, and the first time that CLx goes false, ARx (assembly register set output) comes true.

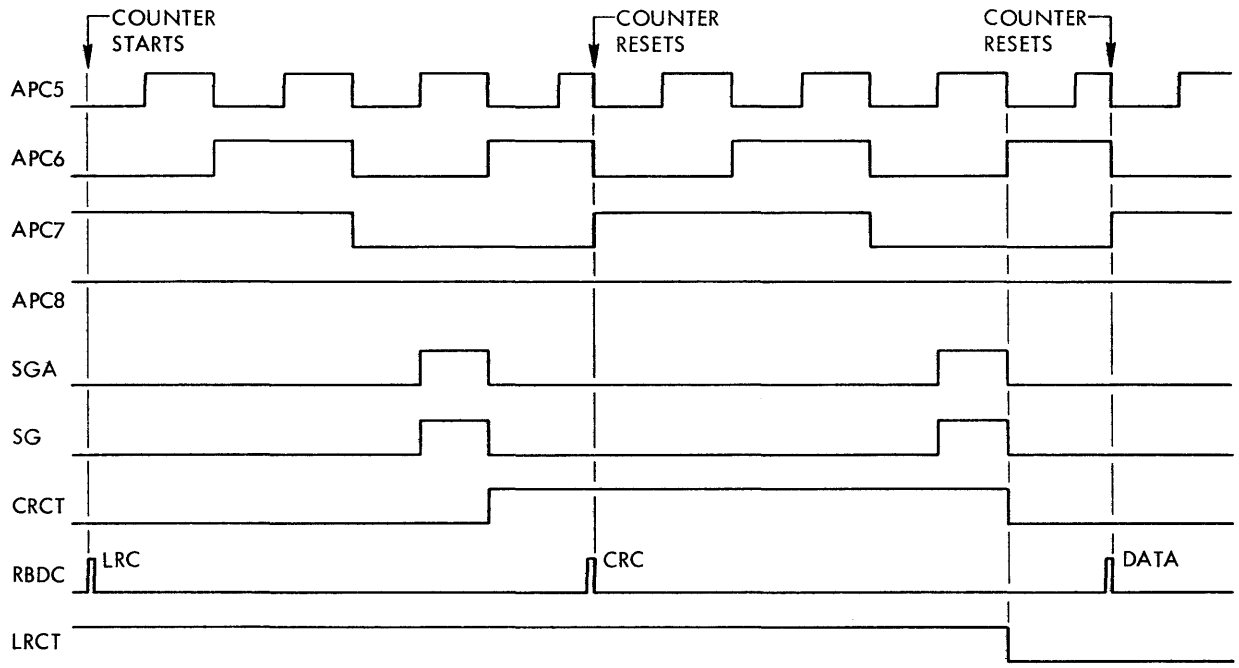


Figure 4-28. Assembly Period Counter (Read Backward, CRC $\neq$ 0), Timing Diagram

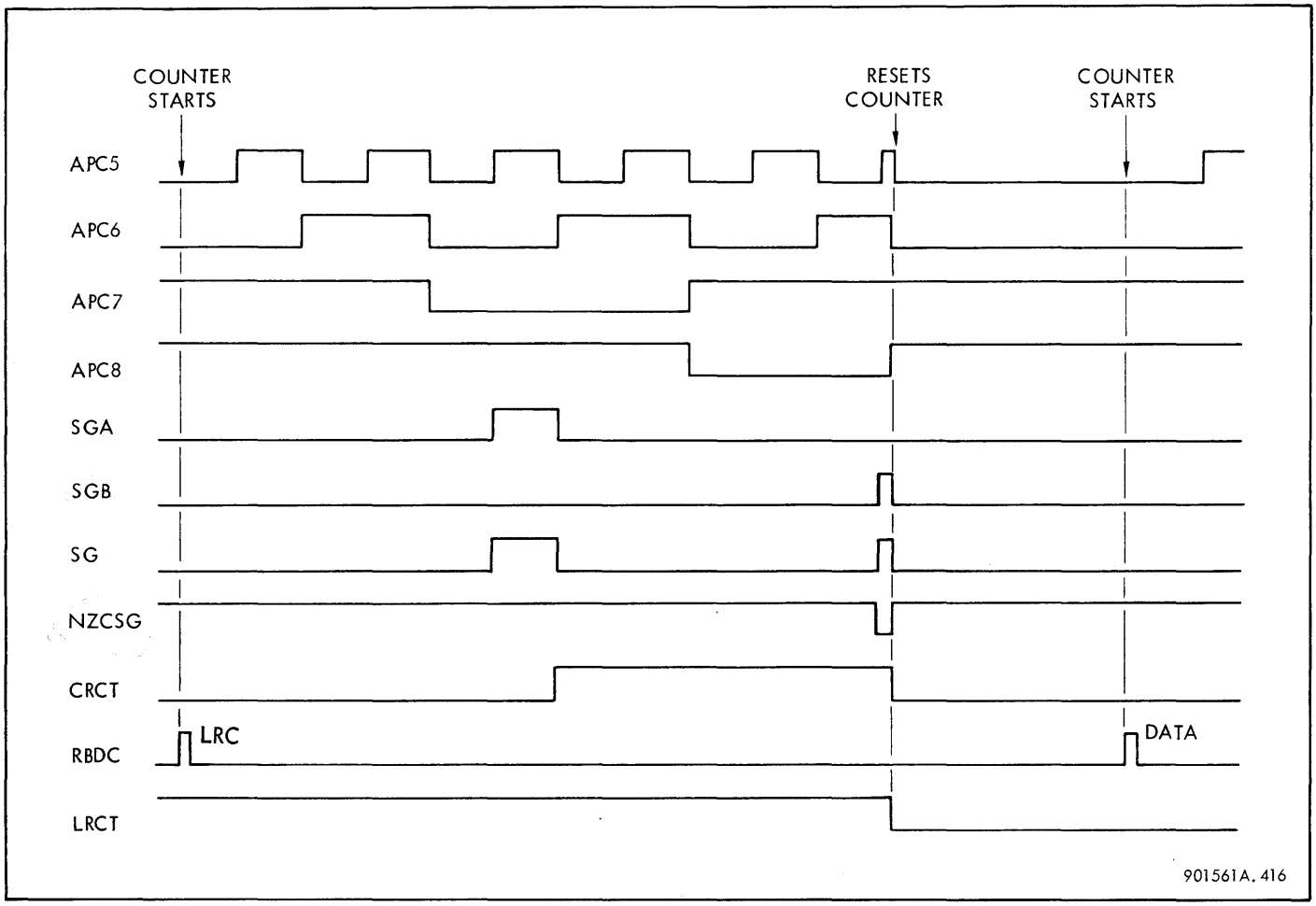


Figure 4-29. Assembly Period Counter (Read Backward, CRC = 0), Timing Diagram

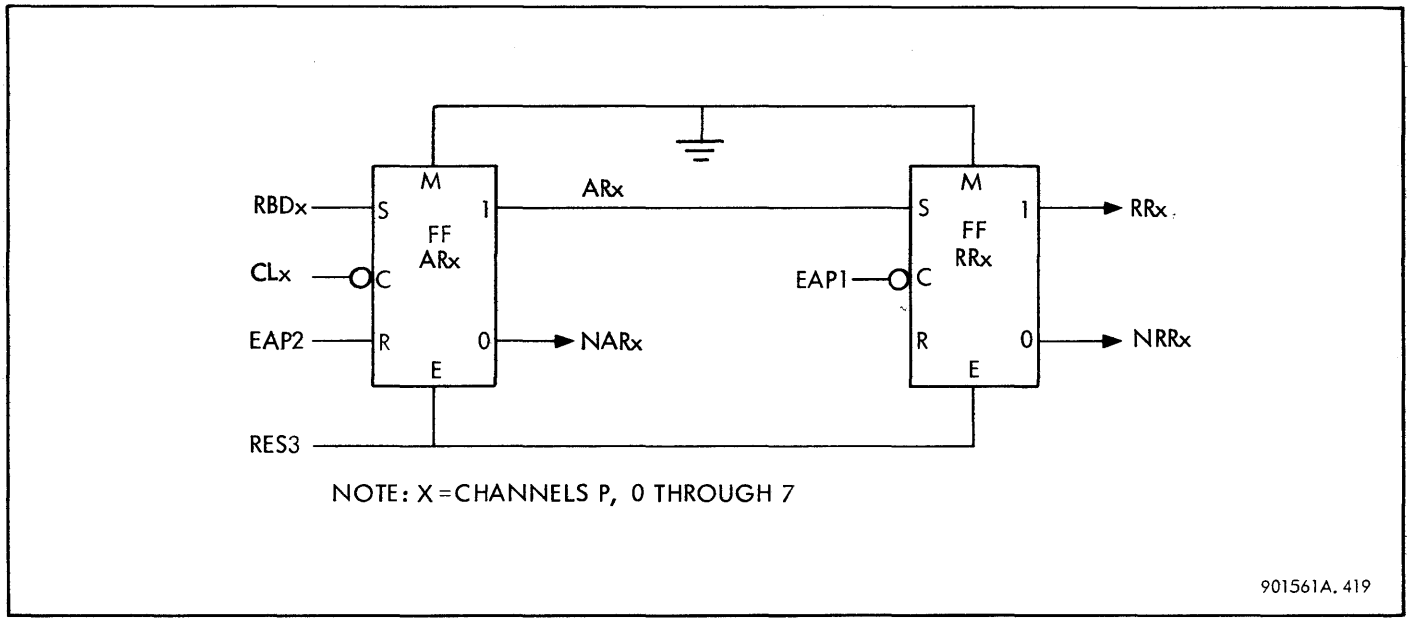


Figure 4-30. Assembly and Read Registers, Simplified Logic Diagram

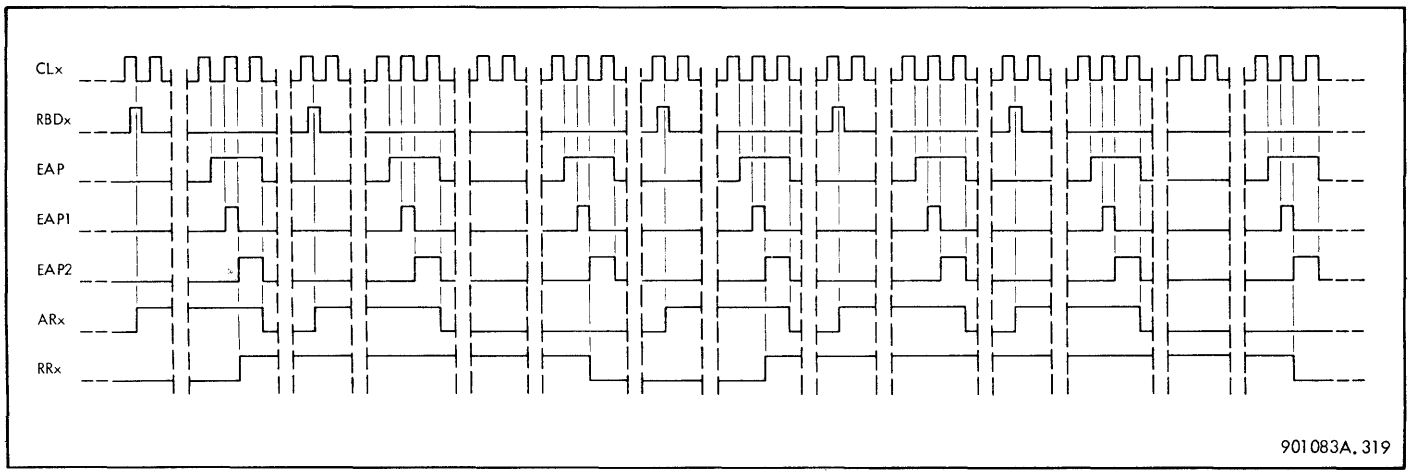


Figure 4-31. Assembly and Read Register Outputs with 1101110 Input, Timing Diagram

At the proper time, EAP1 comes true and then goes false, causing RRx (read register set output) to come true and to store the one bit. EAP2 comes true and at the next clock time, ARx goes false. When the second one bit arrives, RBDx and then ARx again go true. Flip-flop RRx, having been previously set, remains set when EAP1 goes false, since its set input (ARx) is again true. EAP2 comes true, and flip-flop ARx is clocked and reset. On the third character time, the channel contains a zero bit and ARx remains reset. The next time that EAP1 goes false, RRx goes false, and the zero bit is stored. Flip-flop RRx resets at this time because its reset input is open and is always true, and because its set input is now false. The assembly and the read registers continue to operate in the same manner for the remaining bits in the sequence.

4-34 Read Character Present Flip-Flop. The read character present (RCP) flip-flop is initially dc-reset. The assembly register is reset each time that a character is stored in the read register. Flip-flop RCP is also clocked and set when EAP2 goes false. RCP is used in the parity error detector logic.

E/RCP = RES3

S/RCP = NRCP

C/RCP = EAP2 + . . .

In between each character, the RCP flip-flop is reset and is clocked.

R/RCP = RCP

C/RCP = RCPR + . . .

RCPR = READ2 SHIFT

SHIFT = Set output of shift flip-flop

The shift flip-flop is initially dc-reset and it is clocked and set at the start of each memory cycle.

E/SHIFT = RES2

S/SHIFT = NSHIFT

C/SHIFT = SHIFT CLOCK

SHIFT CLOCK = DLAP NPRI SRRWA NWCRC + . . .

At the end of the memory cycle which occurs in between characters, the shift flip-flop is clocked and reset. When it resets, signal SHIFT goes false, RCPR then goes false, and flip-flop RCP is reset.

SHIFT CLOCK = SHIFT DLEP + . . .

4-35 PARITY ERROR DETECTOR. The parity error detector logic checks each byte stored in the read register for possible parity errors. It also provides logic for comparing the number of bytes in a record with the parity of the CRC character. The parity error detector output is connected to each of the data bus lines and in a particular manner, to the error pattern register.

4-36 Parity Error Detector Operation. The output of each read register flip-flop is connected to an input gate of the parity error (PE) detector. The input lines are AND-gated or OR-gated to form three groups of three flip-flops each. An initial parity check is made of each group. See figure 4-32 for a simplified input logic diagram of the error detector.

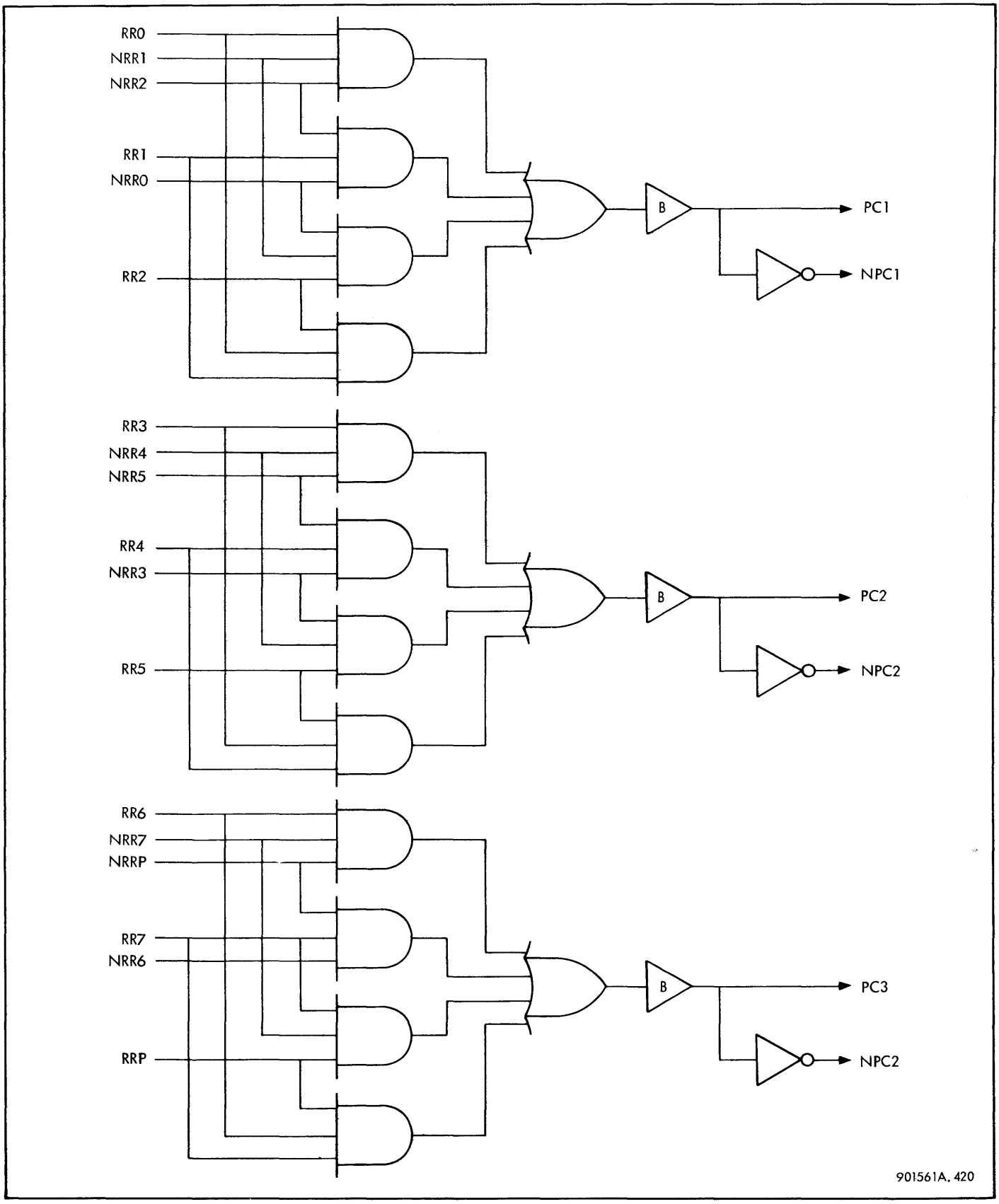


Figure 4-32. Parity Error Detector, Simplified Input Logic Diagram

To be correct, the data byte parity must be odd. If the data byte parity is odd, outputs PC1, PC2 and PC3 are true. Each output is also inverted for further comparisons.

PC1 = RR0 NRR1 NRR2 + NRR0 RR1 NRR2 + NRR0 NRR1 RR2 + RR0 RR1 RR2

PC2 = RR3 NRR4 NRR5 + NRR3 RR4 NRR5 + NRR3 NRR4 RR5 + RR3 RR4 RR5

PC3 = RR6 NRR7 NRRP + NRR6 RR7 NRRP + NRR6 NRR7 RRP + RR6 RR7 RRP

The comparison logic compares PC1, NPC1, PC2, NPC2, PC3, and NPC3 to determine if the complete byte has odd parity. If it does, output PC4 is true.

PC4 = PC1 NPC2 NPC3 + NPC1 PC2 NPC3 + NPC1 NPC2 PC3 + PC1 PC2 PC3

See figure 4-33 for a simplified comparison logic diagram, and figure 4-34 for a simplified output logic diagram of the error detector.

If a parity error is detected, parity error signal PE comes true at RCP time.

PE = [(READ+WRITE) PES + ...] NLRCT + ...

(READ+WRITE) = READ2 + ...

PES = RCP NPC4 NCRCT NLRCT

During a read forward operation, RCP comes true each time that a character is stored in the read register. When RCP goes false, flip-flop ODD is clocked. Flip-flop ODD, initially dc-reset, indicates whether an odd or an even number of bytes have been read. Flip-flop ODD is used also in the CRC character parity check. If ODD is true, the number of data bytes read is odd; if NODD is true, the number of data bytes read is even.

E/ODD = RES4

S/ODD = NODD

C/ODD = (ODD CLOCK)

(ODD CLOCK) = RCP FWD + ...

To further illustrate the parity error detector operation, assume that the first character in a record in a read forward operation is stored in the read register. Assume that RCP is true and that the character parity is odd. PC4 is then true, and NPC4 is false. Looking at the aforementioned equation for PE, it can be seen that at this time, RCP, NCRCT, NLRCT, and READ2 are true. However, since NPC4 is false, PE remains false, indicating that the character in the read register has correct parity. If the character had even parity, NPC4 would be true and PC4 would be false. With all terms in the equation true, PE comes true, indicating that the stored character has a parity error.

After the parity check is made and between characters, RCP goes false. Flip-flop ODD is clocked and set. ODD comes true, indicating that the number of bytes read so far (one) is odd. When the second character is stored in the read register and when the parity check is completed, RCP again goes false and flip-flop ODD is clocked and reset. NODD comes true, indicating that the number of bytes now read (two) is even. The parity check and the byte count operation is performed each time that a data byte in the record is read.

In a read backward operation, flip-flop ODD is dc-reset at the beginning of the record if the CRC character parity is odd. It is also clocked only during data character times and not during CRC and LRC character times.

M/ODD = REV RCP CRCT PC4

C/ODD = (ODD CLOCK)

(ODD CLOCK) = REV RCP NCRCT NLRCT + ...

4-37 CRC Character Parity Check (Read Forward). After checking the parity of the data bytes in the record, the CRC character parity is compared to the number of bytes in the record. If the record contains an odd number of bytes, the CRC character should have even parity. If the number is even, the CRC character should have odd parity.

If we assume that the record contains an odd number of bytes, then flip-flop ODD is set on the last character checked, and the CRC character parity should be even if it is correct. If it is even, PC4 is false, the CRC error signal, CRC ER is false, and parity error indicator PE is disabled. If, however, the CRC character parity is odd, PC4, CRC ER, and PE come true, indicating an error in the CRC parity.

CRC ER = ODD PC4 + ...

PE = CRC ER RCP CRCT FWD READ2 NLRCT + ...

If the number of bytes checked is even, flip-flop ODD is reset and the CRC character parity should be odd. If it is odd, NPC4, CRC ER, and PE are false. If the CRC character has even parity, NPC4, CRC ER and PE come true, indicating an error in the CRC parity.

CRC ER = NODD NPC4 + ...

4-38 CRC Character Parity Check (Read Backward). In the read backward operation, the CRC character parity is checked prior to the data bytes. The comparison of the CRC character parity to the number of bytes is made in a different manner than for read forward.

If the CRC character parity is odd, flip-flop ODD is dc-set at the beginning of the record, as previously described. It

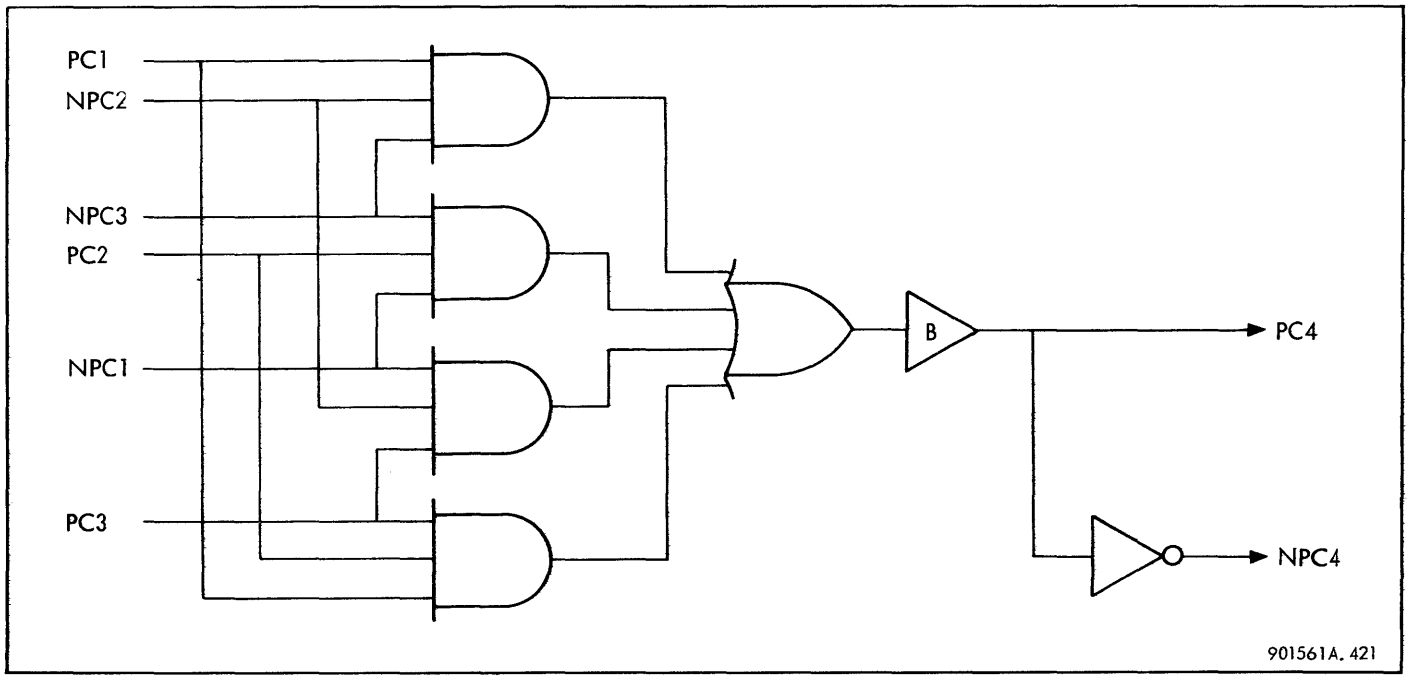


Figure 4-33. Parity Error, Detector, Simplified Comparison Logic Diagram

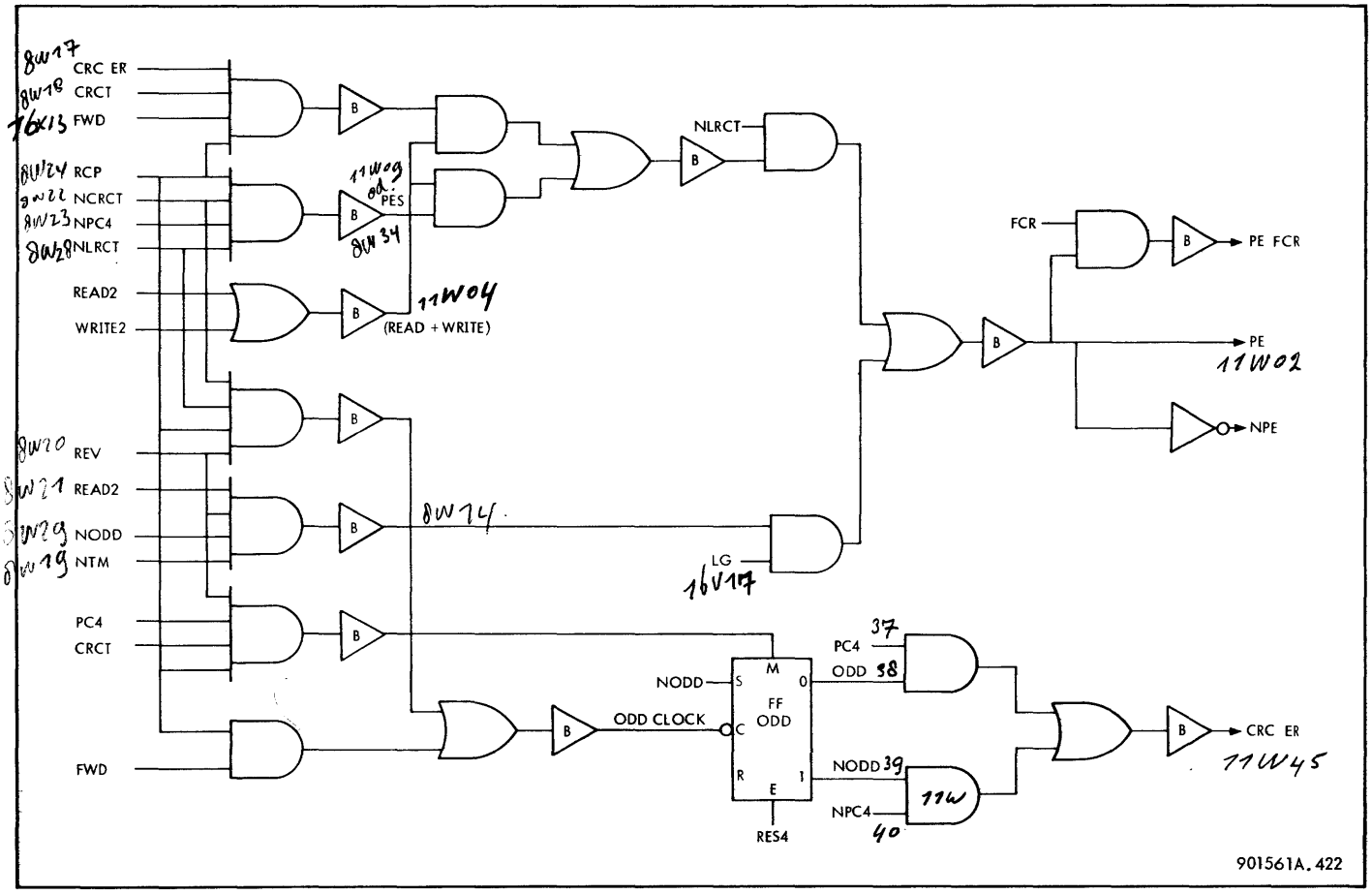


Figure 4-34. Parity Error Detector, Simplified Output Logic Diagram

is clocked only at data byte times. If there is no CRC error, ODD is true when the final character is read and is stored. At LG time, the parity check is made, and PE is thus false. If NODD is true at LG time, PE comes true, indicating an error in the CRC character parity.

PE = LG REV READ2 NTM NODD

See figure 4-35 for a timing diagram of the read backward operation with CRC odd parity.

For a read backward operation in which the CRC character parity is even, ODD is still true at LG time if there is no error. If NODD is true, PE comes true, indicating the CRC character parity error. See figure 4-36 for a timing diagram of the read backward operation with CRC even parity.

4-39 LRC Character Parity Check. The parity of the LRC character is not checked in the parity error detector.

4-40 ERROR PATTERN REGISTER. The error pattern (EP) register is essentially a nine-stage shift register used in the following manner:

a. In a read without correction operation with parity error indication from the parity error pattern stored in the register

b. To hold the track-in-error byte during read with correction operation

c. In LRC character verification during a write operation

The input signal for a read without correction operation is PE from the parity error detector. PE is true each time that there is a parity error in the character being read.

The track-in-error byte is received by the register on the DAxR lines during a set correction order.

The input signals for a LRC character verification during a read-after-write operation are received by the register on the data bus RRCx lines.

Two control signals, EPRL and EPRS, control the shift and the load operations in the register. When EPRL is true (read after write), the next clock signal loads through exclusive OR gating the input data at each stage of the register with the contents of the register. When EPRS is true (read without correction), the next clock signal shifts the data in the register by one stage in a particular pattern.

4-41 Error Pattern Register Operation, Read Without Correction. The EP register flip-flops are initially dc-reset at the start of operations.

E/E1-E9 = EPR RESET

EPR RESET = RES2 NFCR

The EP register flip-flops all have a common clock signal and are clocked each time that DLEP goes false.

C/E1-E9 = EP REG CLOCK

EP REG CLOCK = SHIFT READ1 NFCR DLEP NLRCT + . . .

Signal PE comes true when a parity error is detected and is the only input accepted by the register in a read without correction operation. Together with the reset output of the eighth stage (E8), and control signal EPRS, signal PE is connected through exclusive OR gating to the set input of the ninth stage (E9) in the register. In accordance with the function of an exclusive OR gate, E9 is clocked and set at the next clock signal because only one gate is enabled while all others are disabled. See figure 4-37 for a simplified logic diagram of the ninth stage, which is typical for all stages of the register.

A unique method of shifting the parity error bits through the register is used. Each time that E9 is set, E4, E5, E6, and E7 are set to the complement of their preceding stage at the next clock time, and E1 sets. If E9 is reset, the register shifts at the next clock time with no complementing.

Assume that a parity error has been detected and that PE is true at the first character in a record. Control signal EPRS is then true and remains true for the entire read without correction process.

EPRS = READ1 NFCR

EPRS gates PE on the set input of E9.

S/E9 = NE8 PE EPRS + . . .

E9 sets when the register is clocked by EP REG CLOCK. The set output of E9 is connected to the set input of the first stage, E1.

S/E1 = E9 EPRS + . . .

At the next clock signal, E1 sets. E9 is also connected to the set inputs of stages four, five, six, and seven. Each time that E9 is true, E4, E5, E6, and E7 set to the complement of their prededing stage at the next clock time, and E1 sets.

S/E4 = NE3 E9 EPRS + . . .

S/E5 = NE4 E9 EPRS + . . .

S/E6 = NE5 E9 EPRS + . . .

S/E7 = NE6 E9 EPRS + . . .

The remaining stages in the register operate as a normal shift register.

Assume that a record is being read and that a series of parity errors are detected. Refer to table 4-3 for a representation

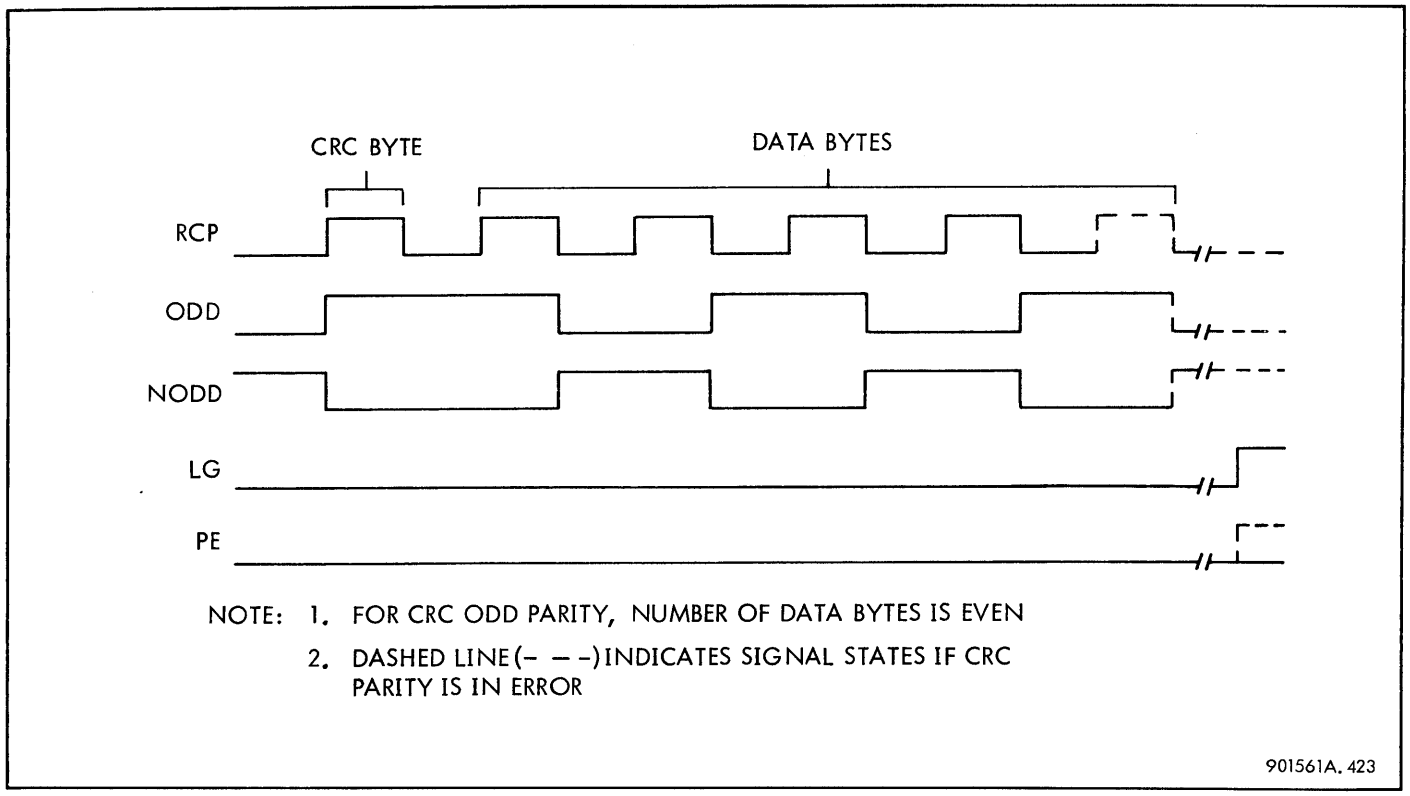


Figure 4-35. CRC Character Parity Check (Read Backward, CRC Parity = Odd), Timing Diagram

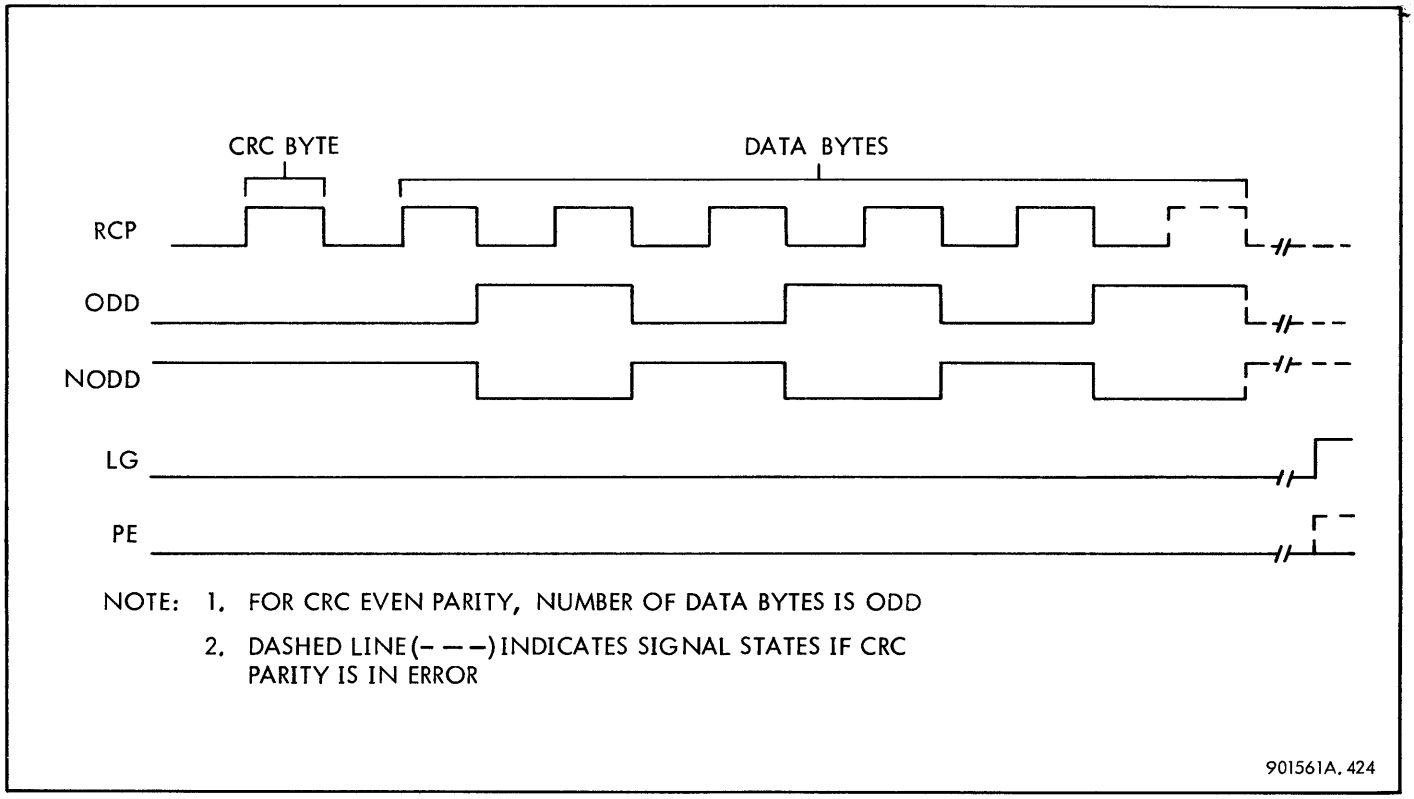


Figure 4-36. CRC Character Parity Check (Read Backward, CRC Parity = Even), Timing Diagram

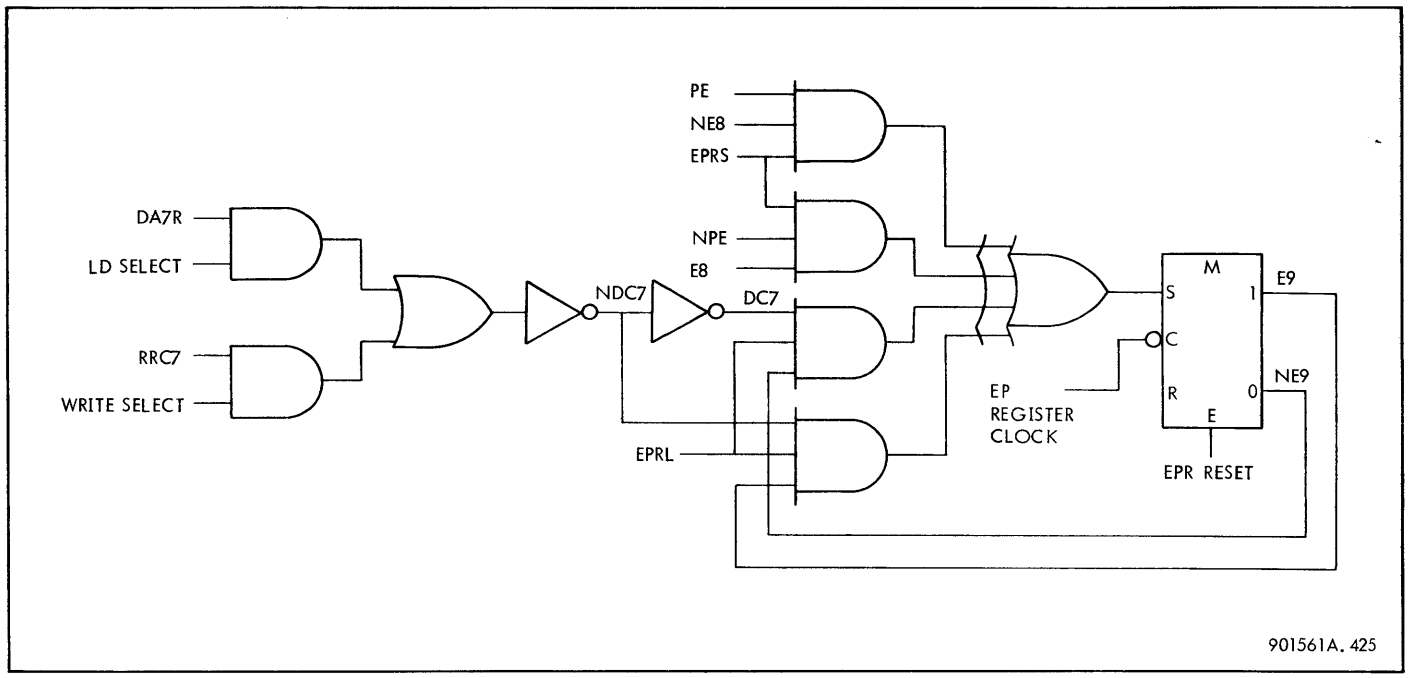


Figure 4-37. Error Pattern Register (Ninth Stage, Typical), Simplified Logic Diagram

of the parity bits as they are shifted through the error pattern register.

Note that the first parity error in a record causes E9 to set when it is clocked. At the following exclusive OR and clock time, E1 is set, and E4, E5, E6, and E7 are set to the complement of their preceding stages. The CRC character is read following the record and, if it has a parity error (as compared to the number of characters in the record), it also causes a parity error indication to be stored by the EP-register. The LRC character does not change the EP-register, even if it contains an error. After the record (including the CRC character) has been stored in the EP-register, the pattern in the EP-register is compared with the pattern at that time in the CRC register. If the two registers do not agree, an error has occurred in the record that was read. (To agree, the CRC register must contain the pattern 111010111, and the EP-register must contain the pattern 000000000.) An attempt is then made to determine which channel contains the error. The CRC register is shifted, one channel at a time, until its pattern agrees with the EP-register. A marker bit is inserted in the write register at the beginning of the comparison process and is shifted through the write register at the same time that the CRC register is being shifted. When the CRC and the EP registers agree, the marker bit in the write register is used to identify the channel that is in error.

4-42 LRC Character Error Check (Read-After-Write). The LRC character is checked in the EP-register during a read-after-write operation. The input signals to the EP-register appear on the RRCx lines and are the flip-flop outputs of the read register. RRCx is AND-gated with enable signal

Table 4-3. Error Pattern Register Shift Cycle, Parity Error Input of 01010011101000011

PE	Exclusive OR, Clock and Shift	E1	E2	E3	E4	E5	E6	E7	E8	E9
0	X	0	0	0	0	0	0	0	0	0
1	X	0	0	0	0	0	0	0	0	1
0	X	1	0	0	1	1	1	1	0	0
1	X	0	1	0	0	1	1	1	1	1
0	X	1	0	1	1	1	0	0	1	1
0	X	1	1	0	0	0	0	1	0	1
1	X	1	1	1	1	1	1	1	1	1
1	X	1	1	1	0	0	0	0	1	0
1	X	0	1	1	1	0	0	0	0	0
0	X	0	0	1	1	1	0	0	0	0
1	X	0	0	0	1	1	1	0	0	1
0	X	1	0	0	1	0	0	0	0	0
0	X	0	1	0	0	1	0	0	0	0
0	X	0	0	1	0	0	1	0	0	0
0	X	0	0	0	1	0	0	1	0	0
1	X	0	0	0	0	1	0	0	1	1
1	X	1	0	0	1	1	0	1	0	0

WRITE SELECT which is true. The output of the gate is inverted twice and is applied to the set input of Ex in an exclusive OR configuration (see figure 4-37).

WRITE SELECT = N(LD SELECT)

The EP-register is initially dc-reset, and as the record is read, the Ex flip-flops are clocked each time a one appears on the RRCx lines. Assume that the first one is read, and RRCx comes true. At the next clock time, Ex is clocked and set.

S/Ex = DCx NEx EPRL + ...

DCx = RRCx WRITE SELECT

EPRL = NREAD1 + ...

C/Ex = EP REG CLOCK

EP REG CLOCK = WRITE1 RCP + ...

Ex remains set until the next one is read and until RRCx again comes true. Since NEx is now false, the set input to Ex is disabled, and Ex is clocked and reset at the next clock time. This sequence is repeated until the complete record, including the LRC character, has been read. If all characters are correct, the EP-register flip-flops are reset at the end of the record, and signal ERZ (error register zero) comes true.

ERZ = NE1 NE2 NE3 ... NE9

If an error occurs in the record, ERZ is false and signal LRC ERROR comes true.

LRC ERROR = WRITE NERZ + ...

4-43 LRC Character Error Check (Read). In a read operation, the LRC character check takes place in the write (WR) register. The WR logic circuitry is the same as in the EP-register, and the check is made in the manner described in paragraph 4-46. The WR-register is initially dc-reset, and flip-flop WRx is clocked each time that RRCx comes true. If all characters are correct, the WR flip-flops are reset at the end of the record, and signal WRZ (write register zero) comes true.

E/WRx = WR RESET

S/WRx = WIx NWRx (NCCRC LRC CL) + ...

WIx = RRCx READ SEL + ...

READ SEL = NWRITE3

C/WRx = WRITE REG CLOCK

WRITE REG CLOCK = DLEP MD2 + ...

WRZ = NWRP NWR0 NWR1 ... NWR7

If an error occurs in the record, WRZ is false and LRC ERROR comes true.

LRC ERROR = NWRITE2 NWRZ + ...

4-44 READ DATA BUS. Nine individual data bus circuits are used in the read phase to transfer data between the various registers and the eight-byte buffer memory. Each data bus receives an input from the PE detector, the EP-register, and the read register, during the read phase. The outputs of the data bus go to the CRC register, the EP-register, the write register, and the eight-byte buffer memory. See figure 4-38 for a simplified logic diagram of a typical read data bus.

4-45 Data Bus Operation, Read Without Correction. In a read without correction operation, flip-flop FCR is in the reset state, and set output FCR is false. Input PE FCR from the PE detector is AND-gated with input Ex from the EP-register, and since FCR is false, signal PE FCR Ex is also false.

PE FCR Ex = (PE FCR) Ex

An inversion takes place, and signal N(PE FCR Ex) goes true.

Outputs RRCx and xB are identical to the read register output RRx.

RRCx = N(PE FCR Ex) RRx + ...

xB = RRCx READ1 + ...

If RRx is true, both RRCx and xB are true, and they are false if RRx is false. Output RRCx goes to the EP-register, the CRC register, write register, and the eight-byte buffer memory. Output xB goes to the CRC register.

4-46 Data Bus Operation, Read With Correction. In a read with correction operation, flip-flop FCR is set, and output FCR comes true. PE comes true each time that a parity error is detected in the record being read. Ex comes true for the channel corresponding to the EP-register flip-flop set by the track-in-error byte. If a parity error is detected and if the channel in error is determined, signal PE FCR Ex comes true and N(PE FCR Ex) goes false. Outputs RRCx and xB are generated as the complement of read register output RRx and the channel in error is corrected.

RRCx = PE FCR Ex NRRx + ...

If RRx is false and if an error is indicated, RRCx and xB go true and the error is corrected. The inverse occurs if RRx is true and is in error. The read with correction operation is possible only if the parity error in the record occurs in only one channel.

4-47 MEMORY ACCESS. The eight-byte buffer memory (buffer) provides intermediate buffering between the read data bus and the IOP. Access to memory from either the station control logic or the controller, is controlled by priority flip-flop NPRI and memory clock cycle control flip-flop SRA. Both flip-flops are part of the memory mode select and timing circuitry. See figure 4-39 for a simplified logic diagram of the memory mode select and the timing circuitry. See figure 4-40 for a timing diagram of the memory mode select and timing read operation.

4-48 Memory Access, Station Read Operation. The station control logic gains access to memory through

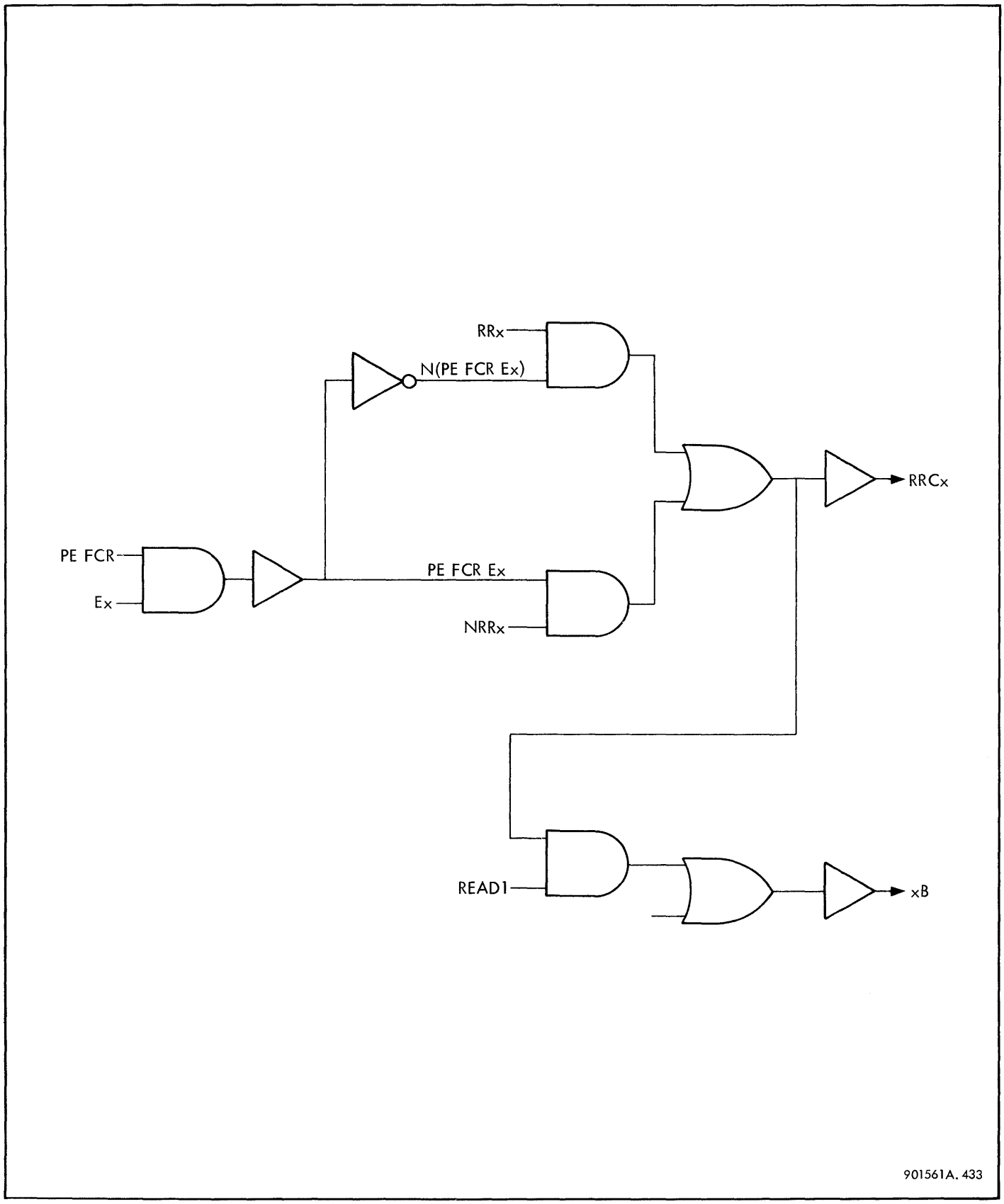


Figure 4-38. Read Data Bus, Simplified Logic Diagram

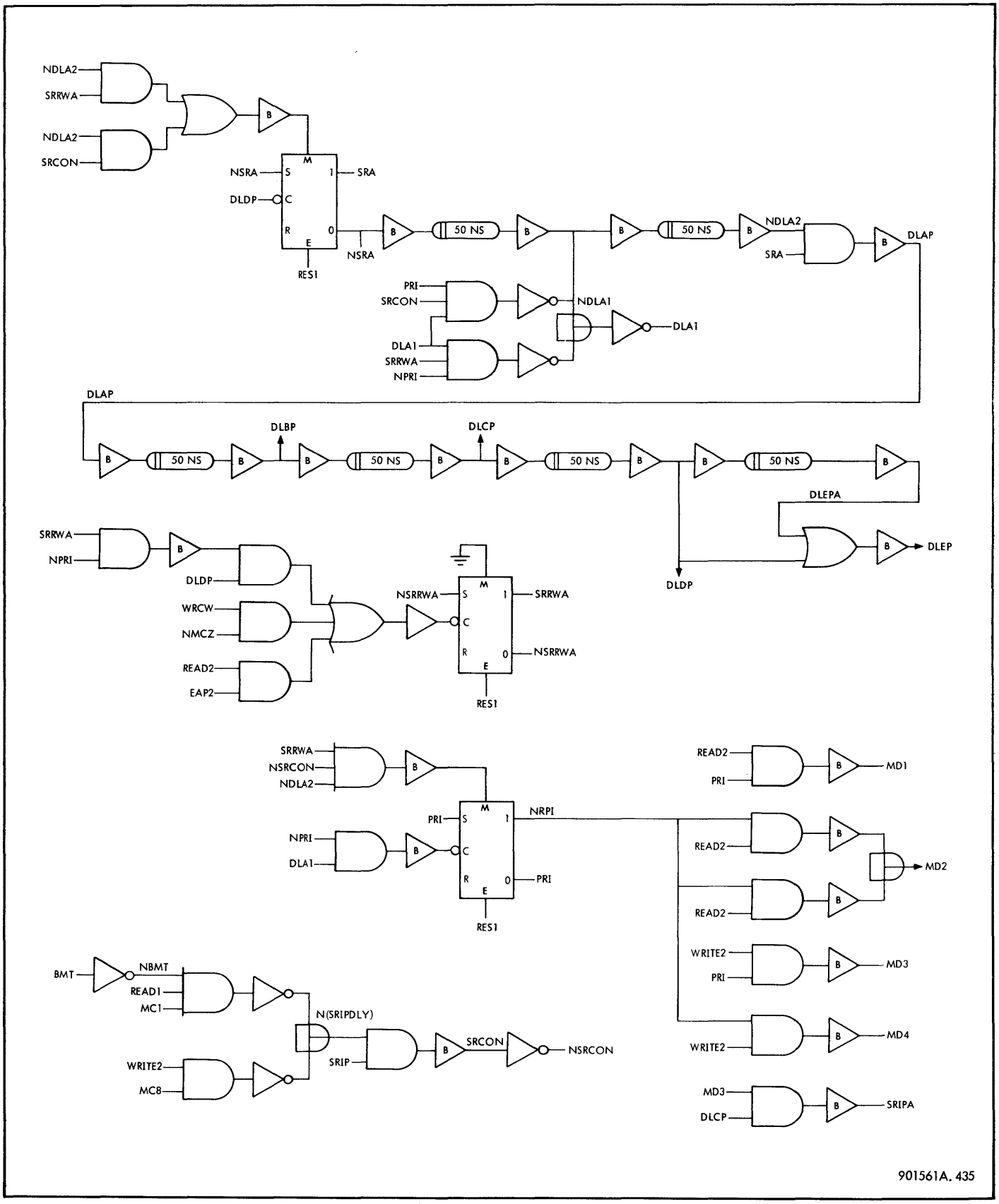


Figure 4-39. Memory Mode Select and Timing, Simplified Logic Diagram

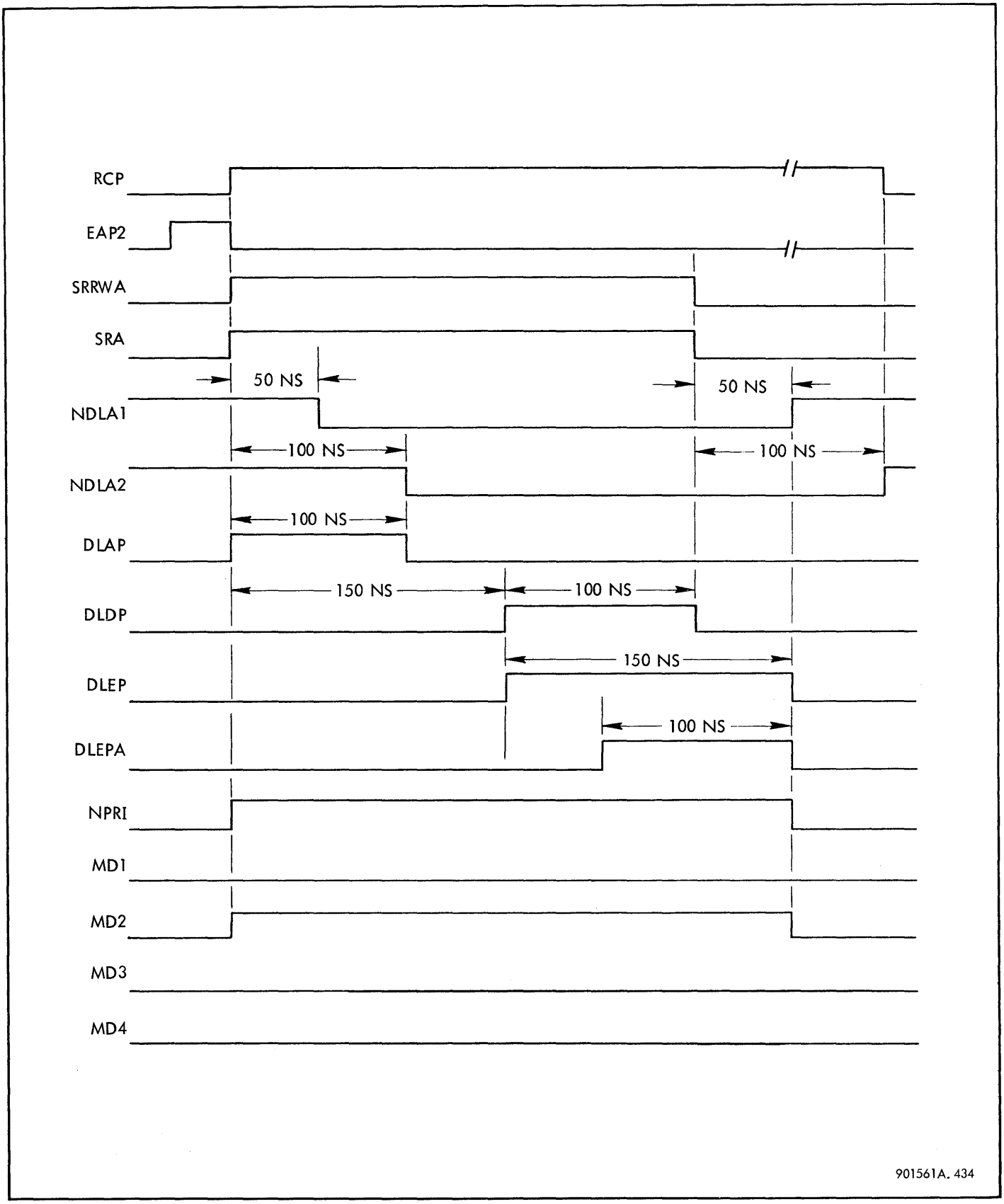


Figure 4-40. Memory Mode Select and Timing Station Read Operation, Timing Diagram

flip-flop SRRWA which is initially dc-reset. When EAP2 goes false, SRRWA is clocked and set.

E/SRRWA = RES1

S/SRRWA = NSRRWA

C/SRRWA = READ2 EAP2 + . . .

If, at this time, the controller does not request access to memory, signal NSRCON comes true and if the previous access cycle is completed, signal NDLA2 also comes true. When SRRWA comes true, flip-flops NPRI and SRA are dc-set.

M/NPRI = SRRWA NSRCON NDLA2

M/SRA = SRRWA NDLA2 + . . .

When SRA and NPRI come true, the memory access cycle begins, and station control of the memory input/output lines is enabled.

4-49 Memory Access Cycle Operation. When SRA comes true, delay line output NDLA1 remains true for 50 ns after NSRA goes false.

NDLA1 = [NSRA (delayed 50 ns)]
(NSRCON + . . .) (NSRRWA + . . .)

Delay line output NDLA2 remains true for 100 ns after SRA comes true and is AND-gated with SRA to generate signal DLAP. DLAP drives a series of 50 ns delay lines which provide the various timing signals needed to operate the memory circuits.

NDLA2 = NDLA1 delayed 50 ns

DLAP = NDLA2 SRA

When NDLA1 goes false, DLA1 comes true and is latched true for the remainder of the cycle.

DLA1 = SRA (delayed 50 ns)
+ DLA1 SRRWA NPRI
+ . . . (latching gate)

DLA1 is AND-gated with NPRI to provide the clock for flip-flop NPRI. Because DLA1 is latched true, NPRI cannot be reset until the cycle is completed. This ensures that timing conflicts do not arise from overlapping access requests.

The mode of memory operation (store or remove) is controlled by the memory mode logic.

MD1 = READ2 PRI (read to IOP)

MD2 = READ2 NPRI (write from RR)

MD3 = WRITE2 PRI (write from IOP)

MD4 = WRITE2 NPRI (read to WR)

The three-bit read memory address counter holds the memory location address when removing data from memory and is incremented each time that a remove operation is completed. The write memory address counter is similar and performs the same function when storing data in memory.

Word Address 0 = (MD1 + MD4) MRA
+ (MD2 + MD3) MWA

Word Address 1 = (MD1 + MD4) MRB
+ (MD2 + MD3) MWB

Word Address 2 = (MD1 + MD4) MRC
+ (MD2 + MD3) MWC

The data stored in memory comes from either the data bus (station) or the data lines (IOP).

Memory Input (data) = MD2 RRCx
+ MD3 DAxR

At DLEP time, the data is stored into memory at the location indicated by the address lines.

Store Memory = DLEP (MD2 + MD3)

DLEP = DLEPA + DLDP

When delay line output DLDP goes false, it clocks and resets flip-flops SRA and SRRWA. The reset takes place because the set inputs are false and the reset inputs are open and are always true.

C/SRA = DLDP

C/SRRWA = DLDP SRRWA NPRI + . . .

DLA1 goes false 50 ns after SRA is reset, and flip-flop NPRI is clocked and reset. The controller now has access to memory.

C/NPRI = NPRI DLA1

NDLA2 goes true 100 ns after SRA is reset and another access cycle can begin. The 100 ns delay provides sufficient time for all circuits to settle down.

In a read operation, the character being read from the tape is stored in a discrete address in the eight-byte buffer in the manner described. The process is repeated when the next character is read from the tape, and this character is stored in the next address in the buffer. The process is repeated until five bytes have been stored in memory.

4-50 DATA TRANSFER FROM MEMORY TO IOP. BMT comes true when the eight-byte buffer has stored five bytes.

BMT = READ1 MCC + . . .

MCC is the output of the third stage in the memory character counter and comes true when there are five bytes stored in

the buffer. BMT is AND-gated with NFUN and generates signal BAND30. BAND30 is AND-gated with BAND24 to generate CSL.

CSL = BAND30 BAND24 + . . .

BAND30 = BMT NFUN

BAND24 = FU1 READ

CSL is the service call request signal and is routed through the subcontroller where its signal designation is changed to SC. SC is the service call request that is sent to the IOP. The IOP answers the service call request and returns signal ASC (acknowledge service call), FS (function strobe), and priority signal AVI (available input). The subcontroller changes the signal designations from ASC to ASCR, FS to FSR, and AVI to AVIR. These signals are used to generate ASCB.

ASCB = ASCR FSR AVIR ASCM + . . .

ASCM = LSH + LSL NHPSL

ASCB is the set input for service connect flip-flop FSC.

S/FSC = ASCB

C/FSC = FSR NFSC + . . .

Flip-flop FSC is clocked and set when FSR goes false. FSC is AND-gated with NMAN to generate signal NFSCC which is inverted to FSCC. FSCC is AND-gated with FU2 and BAND31 to generate signal SRIP. SRIP is the control signal that initiates the transfer of data from the buffer to the IOP. See figure 4-41 for a buffer to IOP data transfer timing diagram.

SRIP = FSCC FU2 BAND31 + . . .

BAND31 = FU1 NFU3 READ

When SRIP comes true, SRCON also comes true.

SRCON = SRIP NSRIPDLY

NSRIPDLY = (NWRITR2 + . . .) (BMT + . . .)

SRCON is AND-gated with NDLA2 to dc-set flip-flop SRA. When SRA comes true, the delay lines in the memory mode and the select timing circuitry are again activated to provide the clocks necessary for the data transfer operation.

M/SRA = SRCON NDLA2 + . . .

NPRI remains in a reset condition during the transfer of data from the eight-byte buffer to the IOP, since the mark input is inhibited by NSRCON. This causes MD1 to be true and MD2, MD3, and MD4 to be false.

MD1 and DLA1 clock the read memory address counter, and the counter selects the address of the data byte that is to be read from memory.

MRA CLOCK = MD1 DLA1 + . . .

The buffer memory output lines (FM0 through FMP) now contain the data byte from the location selected by the read address lines. The FMx lines are AND-gated with ABD (enable read data), and the signal designation is changed to DAxD. The DAxD lines are connected to the subcontroller, and the signal designation is changed to DAx. The DAx lines are connected to the IOP for data transfer.

When the delay line signal at DLBP goes false (approximately 150 ns after SRA), it clocks the read data present flip-flop.

S/RDP = NRDP

C/RDP = MD1 DLBP + SRIP RDP

The reset input of the flip-flop is wired true so that it functions in a toggle mode. When RDP comes true, it indicates that a data byte is present on the FMx lines of the eight-byte buffer.

When RDP comes true, it causes RSD to come true.

RSD = RSDX + RSDY

RSDX = NRSAR (RDP BAND24 + . . .)

RSDY = BAND22 FU2 NFU3

BAND22 = Write NWTM FU1

When the IOP has strobed the data from the FMx lines, it causes RSAR (request strobe acknowledged) to come true.

RSAR = RSARC NMAN + RSDD MANA

RSAR = RSARC NMAN + RS D0 MANA

RSARC = RSA

RSDD = FRSDD NFRSD + RSDIX RSDD

NMAN = NMANC (Selector toggle switch)

MANA = Inverted output of NMAN

When RSAR comes true, NRSAR goes false and causes RSD to go false. When RSD goes false, it clocks flip-flop FU2 and causes it to reset.

S/FU2 = ESR 01F00U + NESR 01F01U

C/FU2 = CFU2X RSD + . . .

CFU2X = 01F00U

MFU2 = BAND25 + BAND26

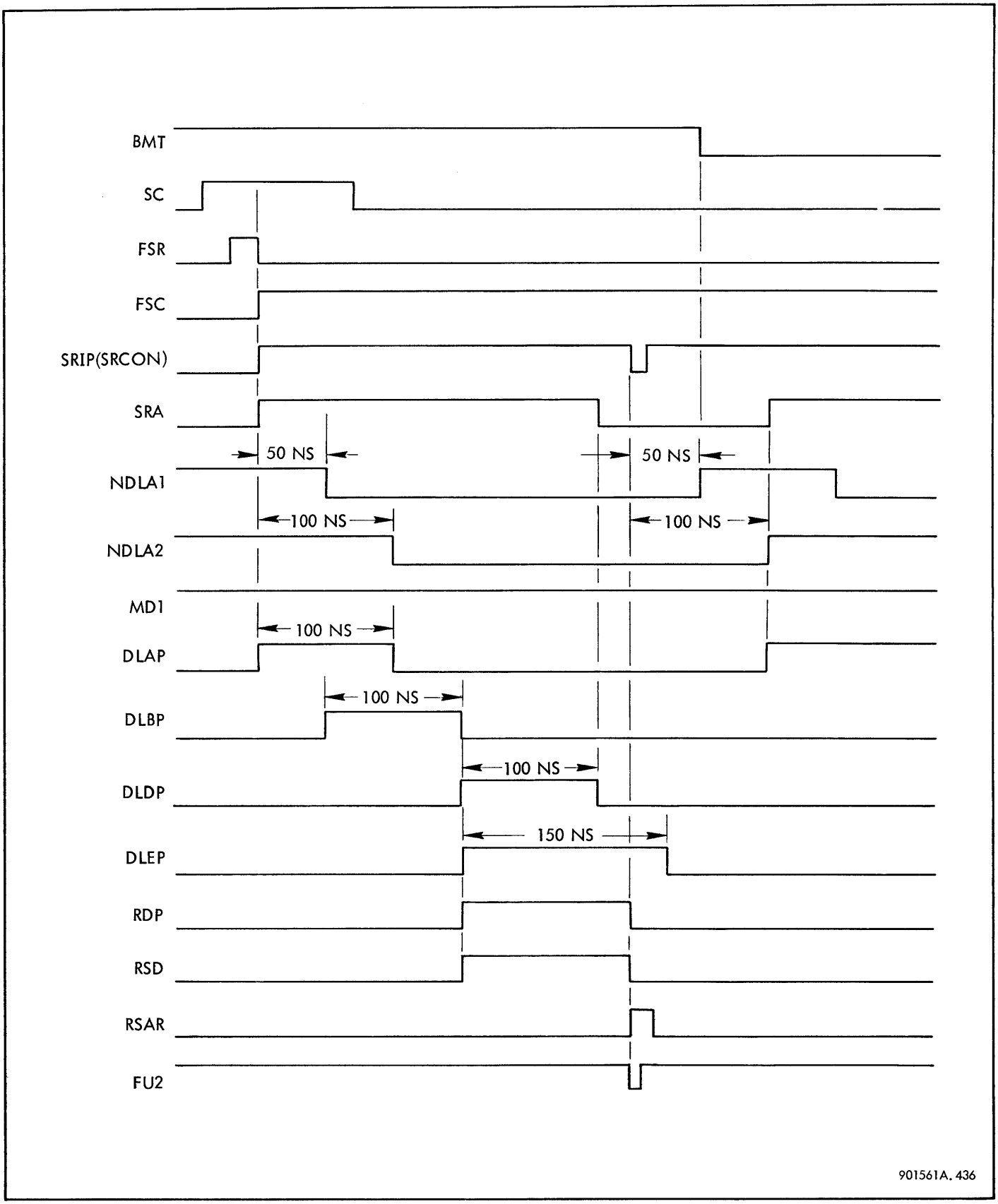


Figure 4-41. Buffer to IOP Data Transfer (Read), Timing Diagram

E/FU2 = RSTS

M/FU2 = MFU2

BAND25 = BAND22 NRSAR

BAND26 = BAND24 NRDP

The reset input of the flip-flop is wired true so that it operates in a toggle mode.

When FU2 resets, it causes SRIP to go false. NDLA1 goes true 50 ns later, and in another 50 ns, NDLA2 goes true.

One byte has now been read from the eight-byte buffer and has been transferred to the IOP, leaving four in memory. Three more bytes must be transferred to the IOP before the service connect flip-flop FSC is reset.

When SRIP goes false, its trailing edge clocks and resets flip-flop RDP. NRDP is connected to the dc-set input of flip-flop FU2, and when RDP resets, NRDP sets FU2.

When FU2 is set, SRIP comes true. SRA is now false, having been reset by DLDP, and NDLA2 goes true 100 ns after SRIP goes false. When NDLA2 goes true, it allows SRA to set, and a new pulse is started down the memory delay line. This sequence is repeated until a total of four bytes have been read from the eight-byte buffer and have been transferred to the IOP. See figure 4-42 for a timing diagram of the four byte buffer to IOP transfer.

After the third byte has been read by the IOP, the IOP raises ED (end data) and ES (end service). This indicates that the last byte is to be transferred; service is then disconnected. The signal designations are changed from ED and ES to EDRC and ESRC, respectively, in the subcontroller. When EDRC and ESRC come true, they cause EDR and ESR to come true.

EDR = NMAN EDRC + . . .

ESR = NMAN ESRC + . . .

When both EDR and ESR are true simultaneously, the last byte is transferred and FSC is reset. ESR is connected to the reset input of FSC, and RSD is connected to its reset clock input.

R/FSC = ESR FSC

C/FSC = RSD FSC + . . .

When RSD goes false, it clocks FSC. FSC resets, disconnecting service.

The eight-byte buffer now stores four more bytes until a total of five is stored in memory, BMT comes true, and the transfer process is then repeated. If the last record that is read stores less than five bytes in memory, BMT cannot come true and start the transfer process. If this is the case, however, NMCZ and CCRC come true and force BMT to come true to read the final bytes from memory.

BMT = NMCZ CCRC + . . .

When the last bytes are being read from memory, the process is repeated as previously explained until the read memory character counter has one count left (that is, one byte is left in memory). When this happens, EDB is raised and causes EDD to come true.

EDB = CCRC MC1

EDD = FSCC EDB BAND31 + . . .

EDD causes EDR to come true, and EDR notifies the IOP that this is the last byte. EDR is connected to the set input of FU3 (terminal order flip-flop), where it is AND-gated with NESR and FSCC.

S/FU3 = EDR NESR FSCC
+ BAND20 DSG NFU3

C/FU3 = FSCC RSD + DST CLK

E/FU3 = RSRS

The reset input of the flip-flop is wired true so that it operates in a toggle configuration. When RSD goes false after the last byte is read from memory, it clocks FU3. FU3 now sets. RDP is not set again, since it was the last character read from the memory. Since RDP was used to raise RSD, another signal must now be substituted. FU3 is AND-gated with FU1 and READ and raises RSD again. RSD indicates a request for a terminal order.

RSD = RSDX + RSDY

RSDX = FU3 FU1 READ + . . .

The IOP now raises ESR. EDD and EDR are made false when FU3 is set.

EDD = FSCC EDB (FU1 NFU3 READ) + . . .

EDR = MANA (EDD + . . .)

ESR indicates that no further bytes follow this operation. The terminal order is given, and the FIN flip-flop is set.

S/FIN = SRWFIN

SRWFIN = ESR (DA1R + DA3R) + FUN + LG

C/FIN = CRWFIN

CRWFIN = RWFIN

RWFIN = RSD FU3 (BAND22 + BAND24)
+ FU1 BOR07

E/FIN = RES1

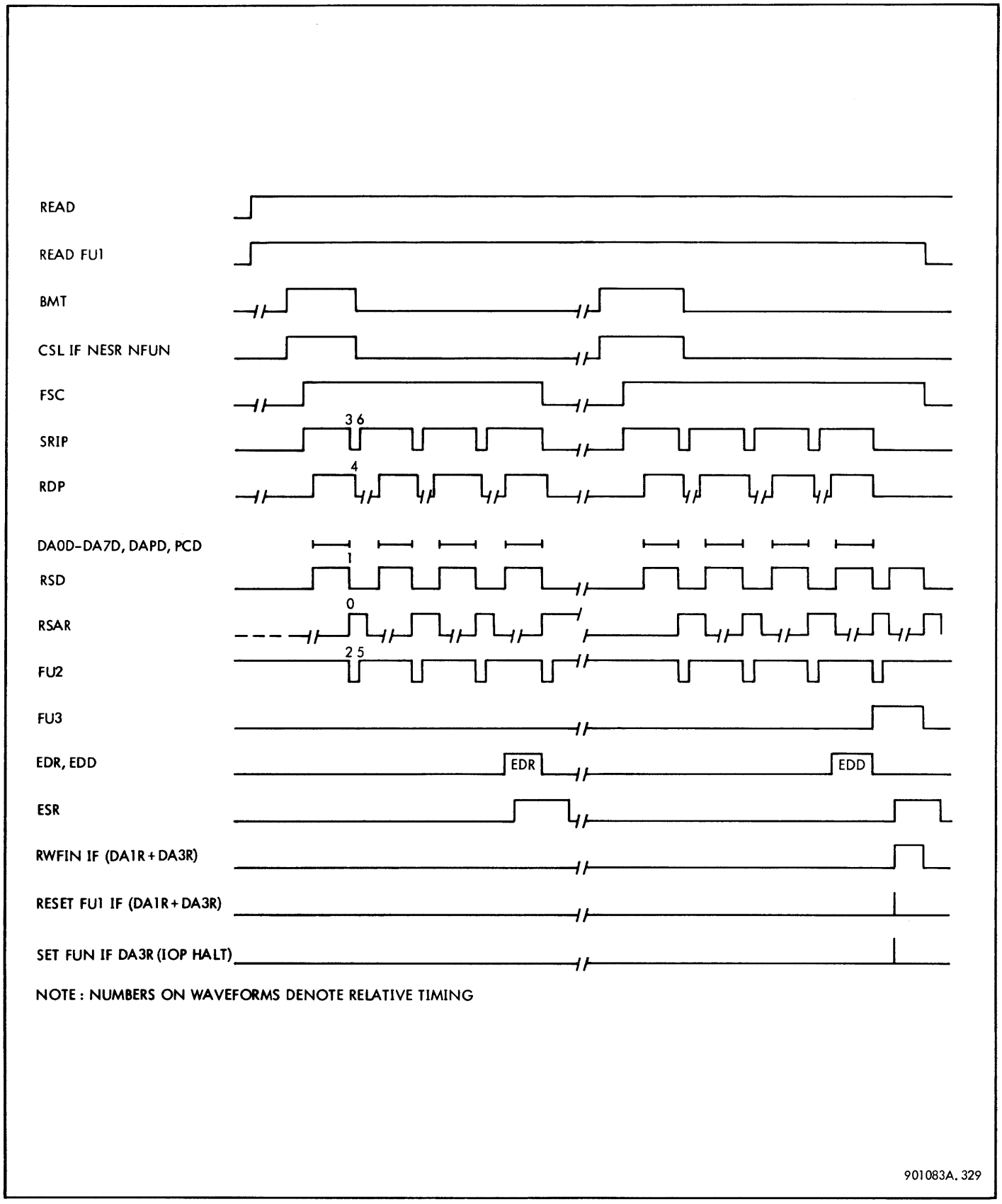


Figure 4-42. Buffer to IOP Four Byte Transfer (Read), Timing Diagram

The IOP now causes RSAR to come true. When RSAR comes true, NRSAR goes false and causes RSD to go false. When RSD goes false, it clocks FU3 and FSC and causes them to reset. (The FIN flip-flop resets at the start of the next order.) ESR goes false, and the data transfer from the eight-byte buffer to the IOP is complete for that record.

4-51 Memory Timing and Data Transfer Using the Selector IOP (Read). The 7320 magnetic tape unit is designed to use either a multiplexer or a selector IOP. The only difference in operation of the tape unit for either type is in the method used to transfer data. The previous example of memory timing and of data transfer was based on the use of a multiplexer IOP.

The method of data transfer between the eight-byte buffer and a selector IOP is as follows: When reading, the request for service is the same as in the previous example, and BMT comes true when there are five bytes in memory. When BMT comes true, it causes SRIP to come true, and when SRIP comes true, the access cycle begins. Data is now transferred from the buffer memory to the IOP as was done in the previous example, with the exception that service connect flip-flop FSC does not reset after four bytes have been transferred from memory.

Since the buffer to IOP transfer rate is much faster than the station to buffer transfer rate, MC1 goes true as soon as only one byte remains in memory, and NSRIPDLY goes false. Signal SRCON is inhibited and also goes false. This occurs because MC1, NBMT, and READ1 are AND-gated and are inverted.

READ1 and NBMT were true prior to MC1. BMT had previously gone false when the first byte was transferred from memory, causing NBMT to go true. When MC1 came true, the inverted output of the gate went false to inhibit SRCON. With SRCON inhibited, no further IOP data transfer occurs.

If a character is in the assembly register, flip-flop RCP sets when it is transferred to the read register and causes SRRWA to set. When SRRWA sets, the memory cycle is started and the character is entered in the eight-byte buffer.

When the character is entered in memory, MC1 goes false. This causes SRIPDLY to go false. Since SRIP is still true, SRCON comes true, the memory cycle is repeated, and the byte is transferred from memory to the IOP. This process is repeated as long as there is data to be read from the tape. When there is no more data in the read register, MC1 comes true. NMCZ is true and is AND-gated with CCRC to force BMT true.

BMT = CCRC NMCZ + . . .

MC1 is AND-gated with CCRC to raise EDB.

EDB = CCRC MC1

EDD is raised by EDB, and EDD causes EDR to come true. EDR notifies the IOP that this is the last byte, and the terminal order sequence is initiated and followed as previously explained. The final byte is transferred from memory and the service connect flip-flop FSC is reset.

4-52 READ AND WRITE MEMORY ADDRESS COUNTERS (figure 4-43). The read memory address counter selects the address of a byte to be read from the eight-byte buffer; the write memory address counter selects the address of a byte that is to be stored in the eight-byte buffer.

Each counter is a three-stage binary counter that receives its clock input from the memory mode select and timing circuits. The output of the counters is connected to the eight-byte buffer, and a clock signal is generated in the counters and is connected to the memory character counter.

4-53 Write Memory Address Counter Operation, Tape to Eight-Byte Buffer. The write memory address counter is initially reset by RES1 (see figure 4-43).

RES1 = RES

When MD2 comes true, it is AND-gated with NCRCT and NLRCT. The true output of this gate is then AND-gated with DLA1 to generate MWA CLOCK. MWA CLOCK is connected to the memory character counter and to the clock inputs of the flip-flops in this counter.

MWA CLOCK = DLA1 (MD2 NCRCT NLRCT) + DLA1 MD3

When DLA1 goes false, the first flip-flop in the counter is clocked and sets. MWA is now true and is used to select an address for a byte to be stored in the eight-byte buffer. MD2 comes true again. When DLA1 goes false the next time, MWA resets and MWB sets, selecting the second address. This process is continued, with the counter operating in a normal binary manner. Each time that the counter advances by one state, its count establishes the new address for the byte that is to be stored in memory. When the counter is full, it resets and starts the count cycle again, continuing to count as long as control signals MD2 and DLA1 are present. When the write process is completed, the counter is reset by RES1.

4-54 Write Memory Address Counter Operation, IOP to Eight-Byte Buffer. The operation of the counter for a transfer of data from the IOP to the eight-byte buffer is exactly the same as in the previous example, except that control signals MD3 and DLA1 are now used. DLA1 comes true and is AND-gated with MD3. When DLA1 goes false, the flip-flops are clocked, and they select the address of the byte that is to be stored in memory from the IOP.

4-55 Read Memory Address Counter Operation, Eight-Byte Buffer to Tape. The read memory address counter is initially reset by RES1 (see figure 4-43). When MD4 comes true, it is AND-gated with DLA1. The output of this gate then generates MRA CLOCK, which is connected to the memory character counter and to the clock inputs of the flip-flops in this counter.

MRA CLOCK = MD4 DLA1 + MD1 DLA1

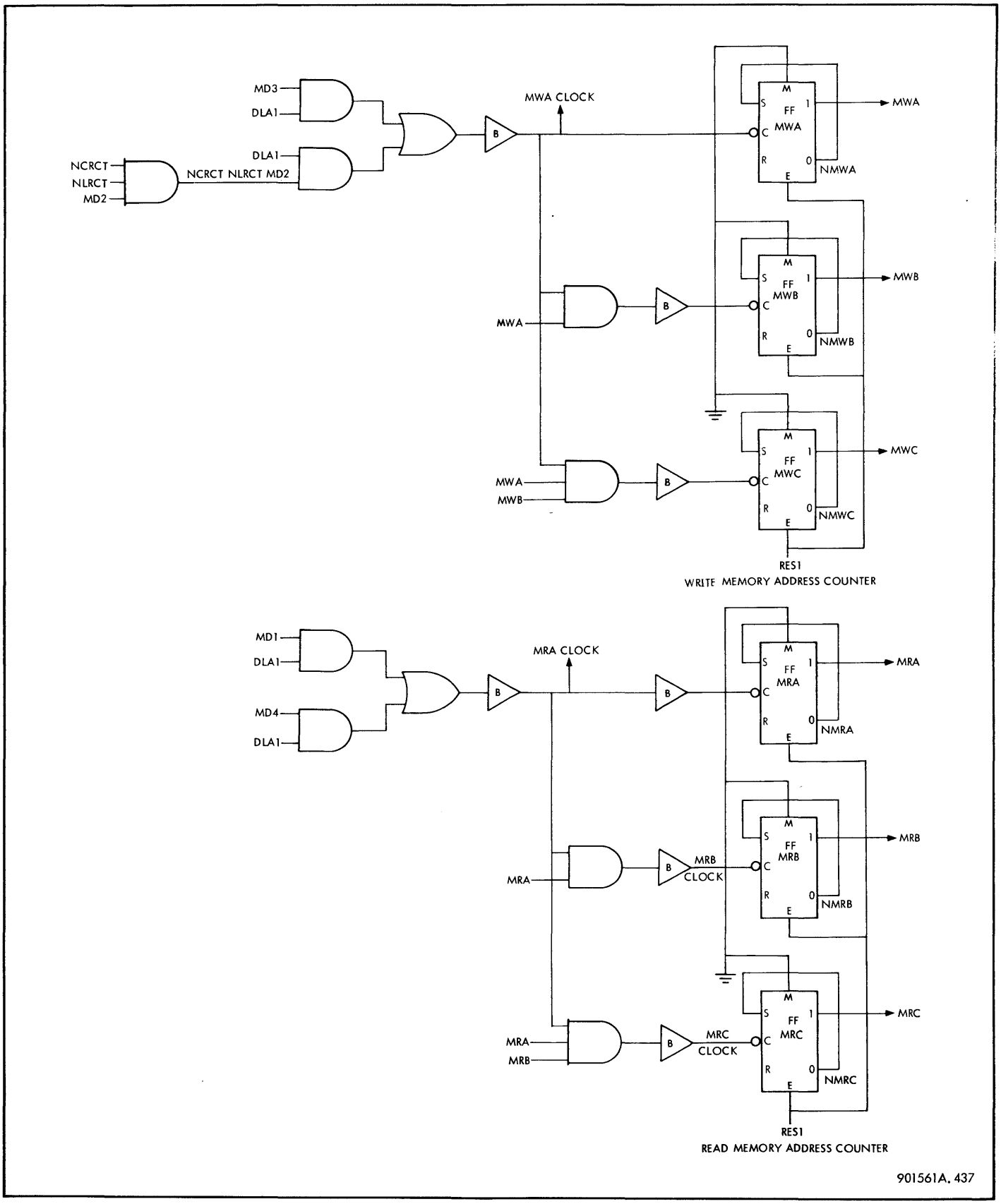


Figure 4-43. Read and Write Memory Address Counter, Simplified Logic Diagram

When DLA1 goes false, the first flip-flop in the counter is clocked and sets. MRA now is true and is used to select the address of a byte that is to be read from memory. MD4 comes true again; when DLA1 goes false, the next time, MRA resets and MRB sets, selecting the second address. This process is continued with the counter operating in a normal binary manner. Each time that the counter advances by one state, its count establishes the new address for the byte that is to be read from memory. When the counter is full, it resets and starts the count cycle again, continuing to count as long as control signals MD4 and DLA1 are present. When the write process is completed, the counter is reset by RES1.

4-56 Read Memory Address Counter Operation, Eight-Byte Buffer to IOP. The operation of the counter for a transfer of data from the eight-byte buffer to the IOP is exactly the same as in the previous example except that control signals MD1 and DLA1 are used. MD1 comes true and is AND-gated with DLA1. When DLA1 goes false, the flip-flops are clocked and select the address of the byte that is to be read from the memory for transfer to the IOP.

4-57 MEMORY CHARACTER COUNTER. The memory character counter is a three-stage, up-down binary counter. It indicates the number of characters that the eight-byte buffer contains at any given time, and calls for either additional data to be stored in the buffer or data to be read from the buffer.

Two clock signals are used to trigger the counter. One clock signal causes it to count up, and the other causes it to count down. Clock signal MWA is generated in the write memory address counter and is the signal that causes the counter to count up when storing a character in the buffer memory. Clock signal MRA is generated in the read memory address counter and is the signal that causes the counter to count down when reading a character from the buffer memory.

Other logic circuitry is included in the counter that is used to control the transfer of data to and from the buffer memory.

4-58 Memory Character Counter Operation, Data Transfer to Memory. The flip-flops in the counter are initially reset by RES1. One flip-flop in the counter indicates that at least one character is stored in the buffer memory. This flip-flop (NMCZ) is initially reset, and NMCZ is false. MCZ is true and is AND-gated with MWA CLOCK at the clock input of NMCZ.

S/NMCZ = MCZ

C/NMCZ = MCZ MWA CLOCK
+ MC1 MRA CLOCK

When MWA CLOCK comes true, the clock input for NMCZ is made true. When MWA CLOCK goes false, NMCZ is clocked and sets. NMCZ is now true and is AND-gated with MWA CLOCK at the clock input of the first stage in the counter.

When MWA CLOCK comes true, the clock input for the first stage in the counter (MCA) is made true. MCA is clocked and sets when MWA CLOCK goes false.

S/MCA = NMCA

C/MCA = NMCZ MWA CLOCK
+ NMC1 MRA CLOCK

NMCZ now remains set as long as any stage in the counter is set or until a dc-reset signal is applied.

C/NMCZ = MC1 MRA CLOCK
+ MCZ MWA CLOCK

MC1 = NMCA NMCB NMCC NMCZ

MC1 is inverted to NMC1, which is true. NMC1 is AND-gated with MRA CLOCK at the clock input of MCA and MCB.

S/MCB = NMCB

C/MCB = NMCA NMC1 MRA CLOCK
+ MCA MWA CLOCK

The counter now continues to count in a normal binary manner as data is stored in the buffer memory. If the buffer should be filled, MC8 comes true, indicating that there are eight bytes in memory.

MC8 = MCA MCB MCC NMCZ

Note that NMCZ is used as an additional count of one to be added to the total count established by MCA, MCB, and MCC. See table 4-4 for an illustration of the states of the flip-flops in the counter for the complete range of storage in the memory.

Table 4-4. Memory Character Counter Flip-Flop States

BYTES IN MEMORY	MCC	MCB	MCA	NMCZ	BMT	
					Read	Write
0	0	0	0	0	0	1
1	0	0	0	1	0	1
2	0	0	1	1	0	1
3	0	1	0	1	0	1
4	0	1	1	1	0	1
5	1	0	0	1	1	0
6	1	0	1	1	1	0
7	1	1	0	1	1	0
8	1	1	1	1	1	0

Note that MCC comes true when there are five bytes in memory. When MCC comes true, it causes BMT to come true (in a read operation). BMT is used to control the transfer of information between the buffer memory and the IOP in a read operation.

BMT = MCC READ1
+ NMCC WRITE1

Thus, any time that there are five or more bytes in memory during a read process, BMT comes true and causes data to be transferred from the buffer memory to the IOP. BMT also comes true during a write operation when there are four or fewer bytes in memory and causes the transfer of data from the IOP to memory.

4-59 Memory Character Counter Operation, Data Transfer From Memory. Data is transferred from the eight-byte buffer during a read or a write operation when the number of characters in memory reaches a certain level.

The memory character counter has counted each character as it has been stored in memory. A request now is made to transfer data from memory to either the IOP (read) or the tape (write). The memory character counter now counts down one each time that a character is read from memory.

BMT comes true when the eight-byte buffer contains five bytes in a read process. BMT initiates the timing process that transfers data from memory. MRA CLOCK now comes true and is connected to the clock inputs of MCA, MCB, and MCC.

C/MCA = MRA CLOCK NMC1
+ . . .

C/MCB = MRA CLOCK NMCA NMC1
+ . . .

C/MCC = MRA CLOCK NMCA NMCB
+ . . .

When MRA CLOCK comes true, the clock inputs of MCA, MCB, and MCC are made true. When MRA CLOCK goes false, MCA and MCB set and MCC resets (see table 4-4). The counter now counts down in a normal binary manner each time that MRA CLOCK triggers the flip-flops. The counter stops its down-count when it reaches a count of one, since additional data is stored in memory. As data is stored in memory, the counter is clocked by MWA CLOCK, and the flip-flops now start to count up.

This process is repeated as data is alternately added or is removed from memory; the counter always reflects the number of characters present in memory at any given time.

In a write process, BMT is true any time that there are four or fewer characters in memory and initiates a data transfer cycle. The eight-byte buffer then stores four additional characters.

The write deskew counter then generates a signal that initiates a data transfer cycle and causes data to be transferred from memory to tape. MD4 and DLEP are AND-gated to make MRA CLOCK true. MRA CLOCK is used to clock the flip-flops in the memory character counter and the read and write address counters. The memory character counter counts down as each character is read from memory and always reflects the number of characters in memory at any time.

4-60 EIGHT-BYTE BUFFER MEMORY (See figure 4-44). The eight-byte buffer memory stores up to eight nine-bit characters for the transfer of data from the tape unit to the IOP in a read process or from the IOP to the tape unit in a write process.

The buffer memory contains nine integrated circuit memory elements and other control circuitry. Each of the nine memory elements can hold up to eight bits of data, with each bit having a separate address, accessible only by external address lines.

The input signals for the buffer memory originate in the read register in a read operation and in the IOP in a write operation. The address of the characters stored in memory is determined by signals generated in the read address counter and in the write address counter. Timing signals from the memory mode select and the timing circuitry are used to clock the buffer memory and to time the address selection process. No clock signal is required, however, to read data from the buffer memory; address selection is the only requirement.

4-61 Eight-Byte Buffer Memory Operation, Data Transfer In (Read). A character is read from the tape and is presented to the eight-byte buffer on the nine RRCx lines from the data bus (see figure 4-44).

The RRCx lines are AND-gated with timing and control signal MD2. When MD2 comes true, the character is impressed on the input lines of the memory. The particular address of this character is established by the WORD ADDRESS EXP lines. MD2 is AND-gated with signals from the write address counter to establish the WORD ADDRESS EXP outputs.

WORD ADDRESS EXP 0 = (MD2 + . . .) MWA + . . .

WORD ADDRESS EXP 1 = (MD2 + . . .) MWB + . . .

WORD ADDRESS EXP 2 = (MD2 + . . .) MWC + . . .

The clock signal for the eight-byte buffer is derived from the memory mode select and the timing circuitry.

C/eight-byte buffer = (MD2 + . . .) DLEP

When DLEP comes true, the clock input for the memory is made true; when DLEP goes false, the character that is on the RRCx lines at that time is clocked into the memory at the address established by the WORD ADDRESS EXP lines.

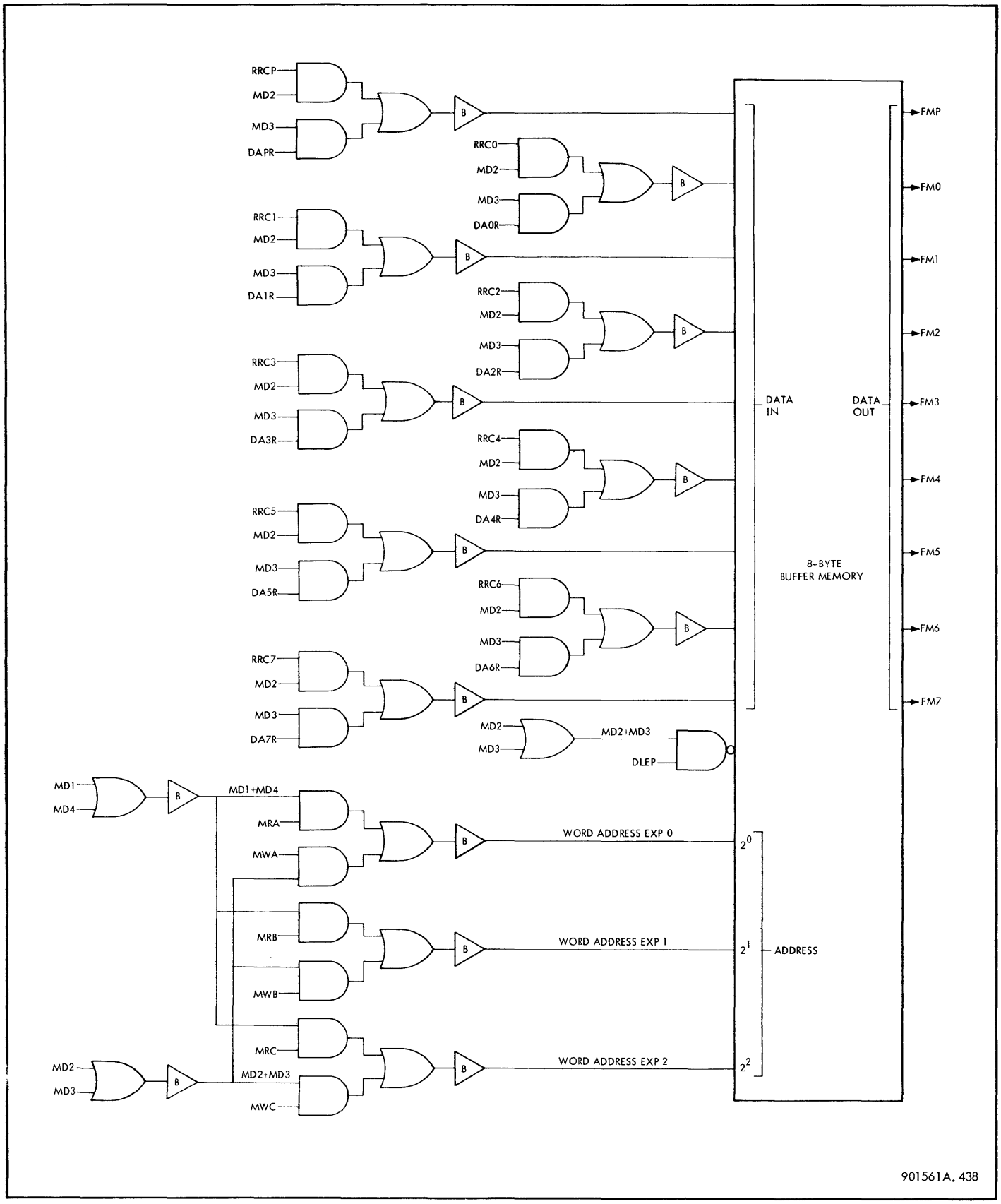


Figure 4-44. Eight-Byte Buffer Memory, Simplified Logic Diagram

4-62 Eight-Byte Buffer Memory Operation, Data Transfer Out (Read). When the eight-byte buffer contains five bytes in a read operation, a control signal is generated in the memory character counter that causes four of the five bytes to be transferred to the IOP before any additional bytes are stored in memory.

MD1 comes true and is AND-gated with MRA, MRB, and MRC from the read address counter in order to establish the WORD ADDRESS EXP outputs. The WORD ADDRESS EXP lines establish the address of the character that is to be read from memory.

The FMx lines of the buffer memory contain the character that is selected by the WORD ADDRESS EXP lines. It is not necessary to have a clock signal to read this character, since it is on the FMx lines as long as its address is selected. The read address counter now selects MRA, MRB, and MRC, and the address of the next character to be read from memory is determined. This process is repeated until one byte is left in memory; then four more bytes are stored in memory and the process is repeated. (See figure 4-45 for a simplified block diagram of the eight-byte buffer input/output configuration.)

4-63 WRITE REGISTER. The write register is essentially a nine-stage shift register. The shift mode is used only in a read operation to determine a track-in-error indication. In a write operation, the flip-flops in the register toggle on input data changes; they transfer data from the eight-byte buffer to the write data register in the station. The register also checks the validity of the LRC character in a read operation.

The input signals for the write register in a read operation originate in the read register and are presented to the write register on the nine RxB lines from the data bus. One single input line, WRZ, is used in a read operation to insert a marker bit in the write register where it is used to determine which track contains the error. In a write operation, the input signals originate in the eight-byte buffer and are presented to the write register on the nine RxB lines from the data bus.

The clock signal for the register is WRITE REG CLOCK.

$$\text{WRITE REG CLOCK} = (\text{MD2} + \text{MD4})\ \text{DLEP} + (\text{LRC ER} + \text{CLKE})$$

The reset inputs for the flip-flops in the register are wired true, and the dc-reset input is WR RESET.

$$\text{WR RESET} = \text{RES3} + \text{WRCW}\ (\text{NMCZ} + \text{FIN})$$

$$\text{RES3} = \text{RES}$$

4-64 Write Register Operation, Read – LRC Character Check. The recorded data is read from the tape and is stored in the read register. The read register outputs, through the data bus, are connected as RRCx lines to the input gating circuits of the write register which are shown in figure 4-46.

RRCx is AND-gated with READ SEL which is true, and the output of this gate is inverted to NWIx.

$$\text{READ SEL} = \text{NWRITE3}$$

$$\text{WIx} = \text{READ SEL RRCx} + \ldots$$

NWIx is AND-gated with NCCRC LRC CL and the set output from that particular stage in the register. NWIx is inverted to WIx and is AND-gated with NCCRC LRC CL and the reset output of that particular stage in the register. The outputs of these two gates are connected in an exclusive OR configuration, and are OR-gated with (CCRC NCRCC) WRx-1. (CCRC NCRCC) is false. The output of this gate is connected to the set input of that particular stage in the register.

$$\text{S/WRx} = (\text{NCCRC LRC CL})\ \text{NWRx WIx} + (\text{NCCRC LRC CL})\ \text{WRx NWIx} + (\text{CCRC NCRCC})\ \text{WRx-1}$$

Clock signal WRITE REG CLOCK comes true when MD2 and DLEP come true and makes the clock inputs of all the stages in the register true. When MD2 goes false, WRITE REG CLOCK goes false, and the flip-flops are clocked and accept the data that is on their set input lines.

Thus, as each character appears on the output of the read register, WRITE REG CLOCK triggers the flip-flops in the write register and the character is stored in the register. The flip-flops in the register follow the input data by one clock time, and change states only when the input data is true. (See figure 4-47 for a representation of typical input/output waveforms.)

When the LRC character is read from the tape, WRITE REG CLOCK triggers the flip-flops in the register. If the LRC character is not in error, all flip-flops in the register reset at this time. All reset outputs from the flip-flops are AND-gated to generate signal WRZ.

$$\text{WRZ} = \text{NWRP NWR0 NWR1} \ldots \text{NWR7}$$

If the LRC character is incorrect, WRZ remains false. WRZ is used in the generation of an LRC error signal.

$$\text{LRC ERROR} = (\text{WRITE} + \text{NWRZ}) + (\text{NWRITE2} + \text{NERZ})\ (\text{CLOCK CCRC/CRCC CLOCK} + \text{SG TM})$$

LRC ERROR is used as a clock signal for the DATE (Data Error) flip-flop. LRC ERROR is held at a false level by CCRC/CRCC CLOCK during the time that the record is being read. If the LRC character is not correct, WRZ is false at the time that CCRC/CRCC CLOCK comes true. When CLOCK comes true, LRC ERROR comes true. If the

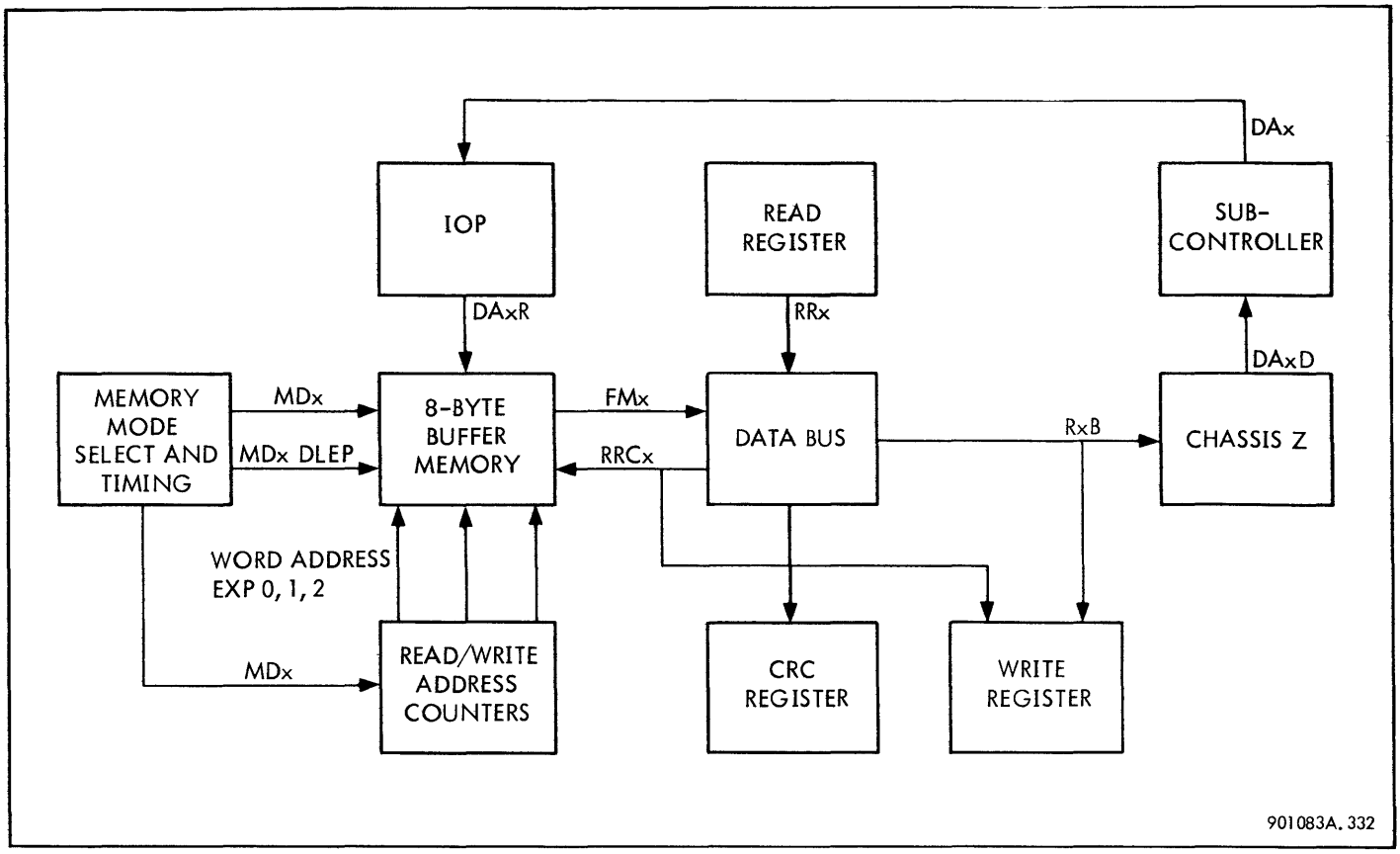


Figure 4-45. Eight-Byte Buffer Memory Input/Output Configuration, Simplified Block Diagram

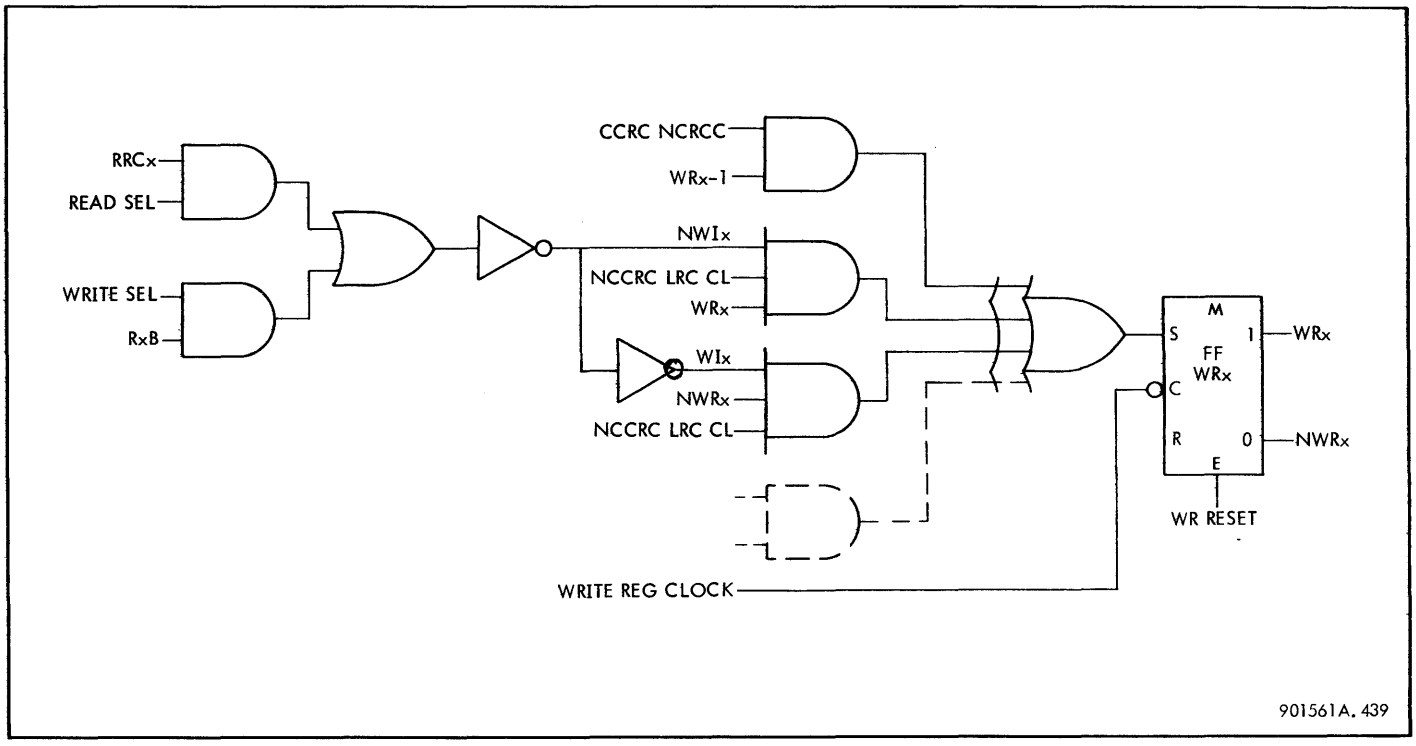


Figure 4-46. Write Register, Simplified Input Gating Logic Diagram

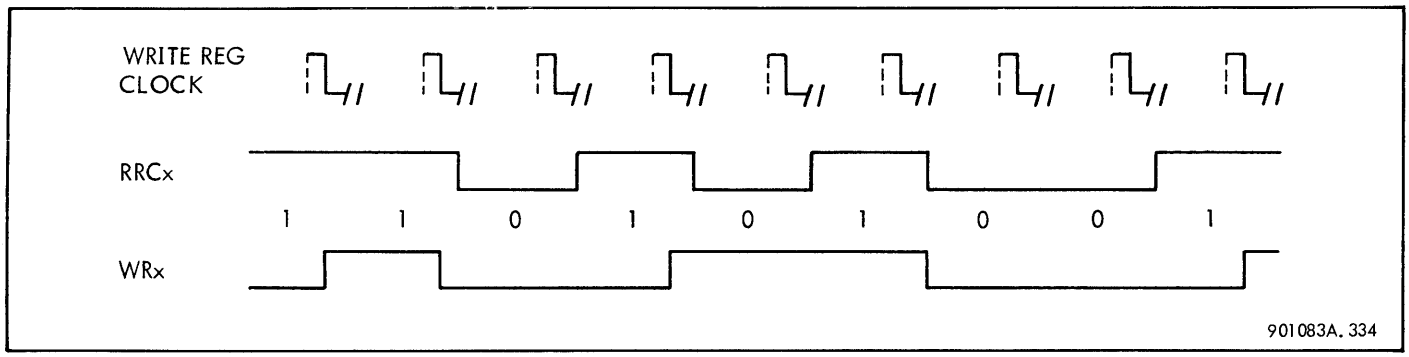


Figure 4-47. Write Register, Input/Output Waveforms with 110101001 Input, Timing Diagram

LRC character had been acceptable, WRZ would have come true and LRC ERROR would have remained false. LRC ERROR now makes WRITE REG CLOCK true and the clock input of DATE true. (See figure 4-48 for a timing sequence of the write register operation on an LRC validity check, when LRC is in error.)

WRITE REG CLOCK = (LRC ER + CLKE) + . . .

When CLOCK goes false, LRC ERROR goes false, causing the DATE flip-flop to set and WRITE REG CLOCK to go false. When WRITE REG CLOCK goes false, it clocks the flip-flops in the register and causes them to reset (since NCCRC LRC CL is false), and the set inputs then are held false.

4-65 Write Register Operation, Read – Track-In-Error Determination. When NCRCC goes false (see figure 4-48), the validity of the CRC character is checked. The pattern in the CRC register is compared to the pattern in the error pattern register. (The CRC register must have the pattern 111010111 and the EP register must have the pattern 000000000.) If the patterns do not compare, a channel on the tape is in error.

At this time, if LRC is good, WRZ is true, since the write register is reset, and WRZ is AND-gated with control signal CCRC NCRCC at the set input of WRP. Signal CCRC NCRCC is true when the CRC register contains the wrong character. Thus, the set input of WRP is true. CLKE then comes true and causes WRITE REG CLOCK to come true. When CLKE goes false, WRITE REG CLOCK goes false and clocks the flip-flops in the register. WRP then sets.

S/WRP = WRZ CCRC NCRCC + . . .

WRITE REG CLOCK = CLKE + LRC ERROR + . . .

CLKE = (CCRC NCRCC) READ2 NWR7 (NCMPR + CMPR WRZ) (CRCCMPRCLK)

CRCCMPRCLK comes true if the CRC register contains the wrong character. It is the set ouput of flip-flop CRCCMPRCLK, which is clocked and set by the master clock oscillator (CLOCK). The use of CRCCMPRCLK prevents time pacing problems associated with the long propagation delays through the write register and associated circuits.

The CRC register is again compared to the EP-register. At the comparison gates, the E-register appears selectively complemented, that is, all bits except 4 and 6 are complemented. If they agree, the marker bit that is now in the WRP flip-flop is used to indicate the channel that is in error. If they do not agree, CLKE again comes true and causes WRITE REG CLOCK and CRC REG CLOCK to come true. When CLKE goes false, both the write and the CRC-registers are shifted. The CRC register is compared again to the EP-register. The marker bit is now in stage WR0. If the comparison is successful, the channel that is in error can be identified.

If they do not compare, the process is repeated again. This operation is repeated until the CRC and the EP-registers agree or until the marker bit has been shifted completely through the write register to stage WR7. If this occurs, SNCRE comes true. SNCRE is the set input for the NCRE flip-flop. NCRE is clocked when READ1 goes false, and causes NCORE (noncorrectable read error) to come true. When NCORE comes true, it indicates that there are errors in more than one channel.

4-66 CRC REGISTER. The CRC register is primarily a nine-stage shift register. During a read operation, the register generates a CRC character from data that is read from tape and compares that character with the CRC character that is written on tape. During a write operation, the register stores and processes data from the input signals and generates a CRC character to be written on tape.

In a read operation, the input signals for the register originate in the read register. They are presented to the CRC register on the nine xB lines from the data bus for a read forward function and on the nine RRCx lines from the data bus for a read backward function.

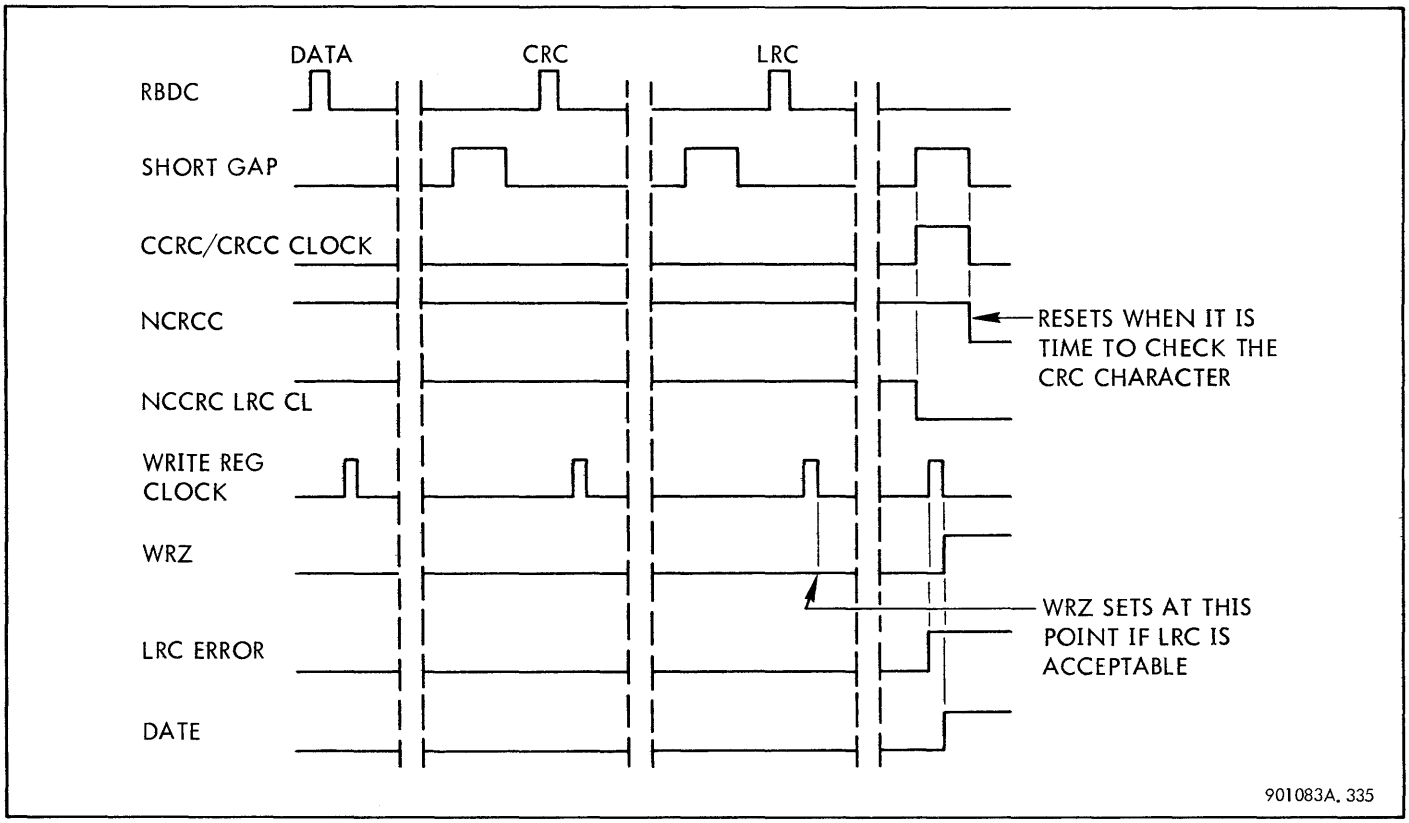


Figure 4-48. Write Register, LRC Validity Check with LRC Character in Error, Timing Diagram

In a write operation, the input signals for the register originate in the eight-byte buffer memory and are presented to the CRC register on the nine xB lines from the data bus.

Two shift control signals, SHIFT and NSHIFT, are used to control the exclusive OR and shift functions of the register. When SHIFT is true, the clock signal is in an exclusive OR configuration with the input date at each stage in the register with the contents of the register. When NSHIFT is true, the clock signal shifts the data in the register by one stage.

Two directional control signals are provided: FWD SEL and REV SEL. FWD SEL is used when reading or writing in a forward direction, and REV SEL is used when reading in a reverse direction.

The clock signal for the register is CRC REG CLOCK.

CRC REG CLOCK = SHIFT CLOCK
(CRCT + NLRCT)
+ WCRC CLOCK
+ (CLKE NWRZ)

SHIFT CLOCK = SHIFT DLEP + DLAP
(NPRI SRRWA NWCRC)

WCRC CLOCK = MCZ NWCRC WRITE2
(FIN + WTM)

CLKE = (CCRC NCRCC) (READ2 NWR7)
(CRCCMPRCLK) NCMPR
+ (CMPR WRZ)

NWRZ = Inverted output from WRZ

NWRZ = WRP + WR0 + WR1 + WR2
+ WR3 + . . . WR7

The reset inputs for the flip-flops are wired true, and the dc-reset input for the flip-flops is RES2.

RES2 = RES

4-67 CRC Register Operation, Read Forward Without Correction. A unique method is employed to shift the data through the register. Any time that the ninth stage in the register (C9) is set, C4, C5, C6, and C7 are set to the complement of their preceding stage at the next clock time, and C1 sets. The other stages of the register operate as a normal shift register at that time, with the output of C9 connected back to the input of C1. If C9 is not set, however, at the next clock time the register shifts by one stage without complementing.

All flip-flops in the register are initially reset by RES2. The first character from the read register is presented to the CRC register on the nine xB lines from the data bus. In the

example being used (a read forward operation), FWD SEL is true. FWD SEL is the inverted output of REV SEL, and REV SEL is the inverted output of FWD. FWD is the inverted output of REV (which the IOP generates) and is false. Figure 4-49 shows the direction control signal generation.

The input data is AND-gated with FWD SEL in the input gating circuits. (See figure 4-50)

The output of this gate is inverted to NDBx.

NDBx = NFWD SEL + NxB
+ NREV SEL + NRRCx

NDBx is inverted to DBx, and DBx is AND-gated with NCx and SHIFT. The output of DBx NCx SHIFT is connected with NDBx Cx SHIFT in an exclusive OR configuration. SHIFT is the set output of the SHIFT flip-flop.

S/SHIFT = NSHIFT

C/SHIFT = SHIFT CLOCK

E/SHIFT = RES2

(NWCRC is the reset output of the WCRC flip-flop. NWCRC goes false when the CRC character is written.)

SHIFT is gated with NDBx and DBx in an exclusive OR gate. The output of this gate is an OR-gate with NSHIFT (which is true) and the set output of the preceding stage. In addition, stages C4, C5, C6, and C7 have an additional gate (indicated by dotted lines in figure 4-49) which accomplishes the complementing function previously explained. The output of these gates is connected to the set input of that particular state in the register (Cx). Each stage in the register now has the set output of its preceding stage connected to its set input.

SHIFT CLOCK comes true when DLAP and SRRWA come true and makes the clock input of the SHIFT flip-flop and CRC REG CLOCK true (see figure 4-51 for a timing diagram).

When DLAP goes false, it causes SHIFT CLOCK and CRC REG CLOCK to go false. The flip-flops in the register are clocked, and SHIFT is set. Since this is the first character of a record in the example being used, all flip-flops in the register are still reset. Thus, all of their set inputs are held false by SHIFT and Cx-1; they remain reset at the clock signal.

DLEP now comes true and causes SHIFT CLOCK and CRC REG CLOCK to come true. When DLEP goes false, the flip-flops in the register are clocked, and SHIFT is reset. Since SHIFT was true when the flip-flops were clocked, the input data is entered into the register as an exclusive OR. The register assumes the state of the first character read.

When the second character arrives, the process is repeated. DLEP and SRRWA cause the SHIFT CLOCK and CRC REG CLOCK to come true, and when DLEP goes false, the register flip-flops are clocked and SHIFT sets. This time, however, data is present in the register. This data is shifted by one stage. If C9 was set, stages C4, C5, C6, and C7 set to the complement of their preceding stage, C1 sets, and the other stages just shift. If C9 was reset, all stages in the register shift. DLEP comes true and causes SHIFT CLOCK and CRC REG CLOCK to come true. When DLEP goes false, the flip-flops in the register are clocked, and SHIFT is reset. The data that the flip-flops contain before they are clocked is connected in an exclusive OR at their set inputs with the new input data, and the flip-flops set accordingly when they are clocked. See table 4-5 for a typical representation of input data and the states of the CRC register. This process is repeated until the entire record is read and is stored in the CRC register.

Following the last character in the record, three blank spaces are encountered on the tape; the CRC character is then read from the fourth space. Three more blank spaces are encountered, and then the LRC character is read from the fourth space. See figure 4-52 for a physical representation of the CRC and the LRC spacing.

After the last character in a record has been read, the assembly period counter continues to count instead of being reset.

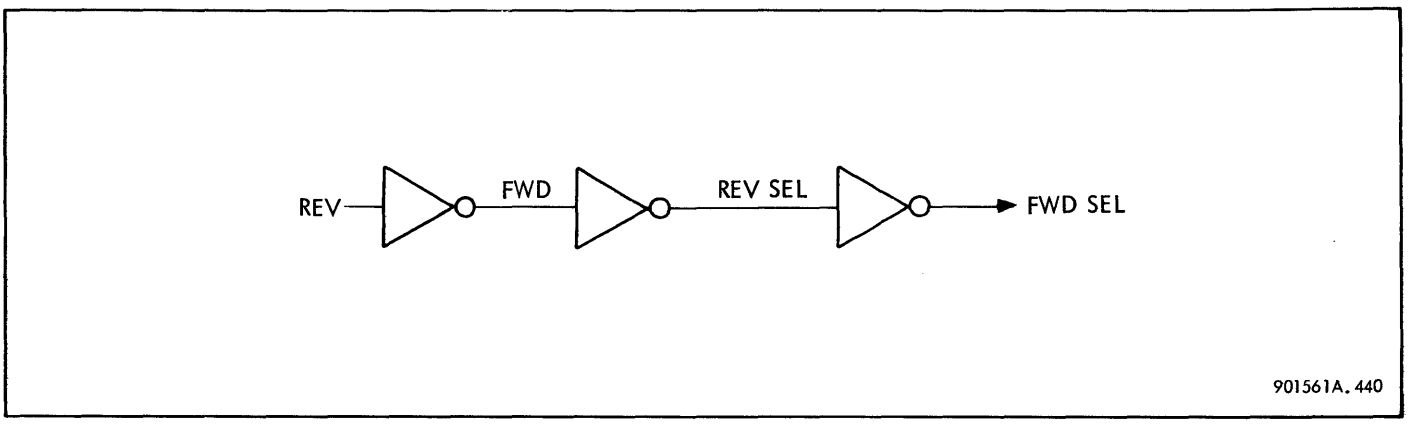


Figure 4-49. CRC Register, Direction Control Signal Generation, Simplified Logic Diagram

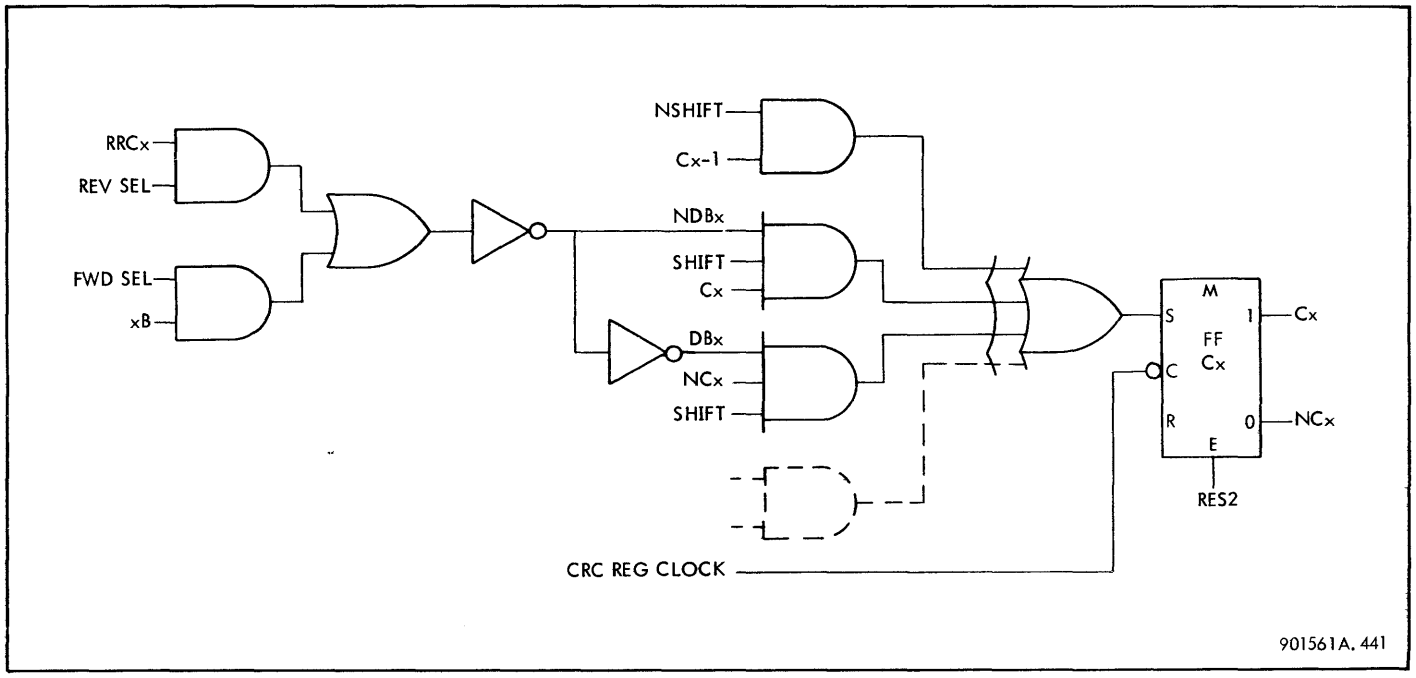


Figure 4-50. CRC Register, Simplified Input Gating Logic Diagram

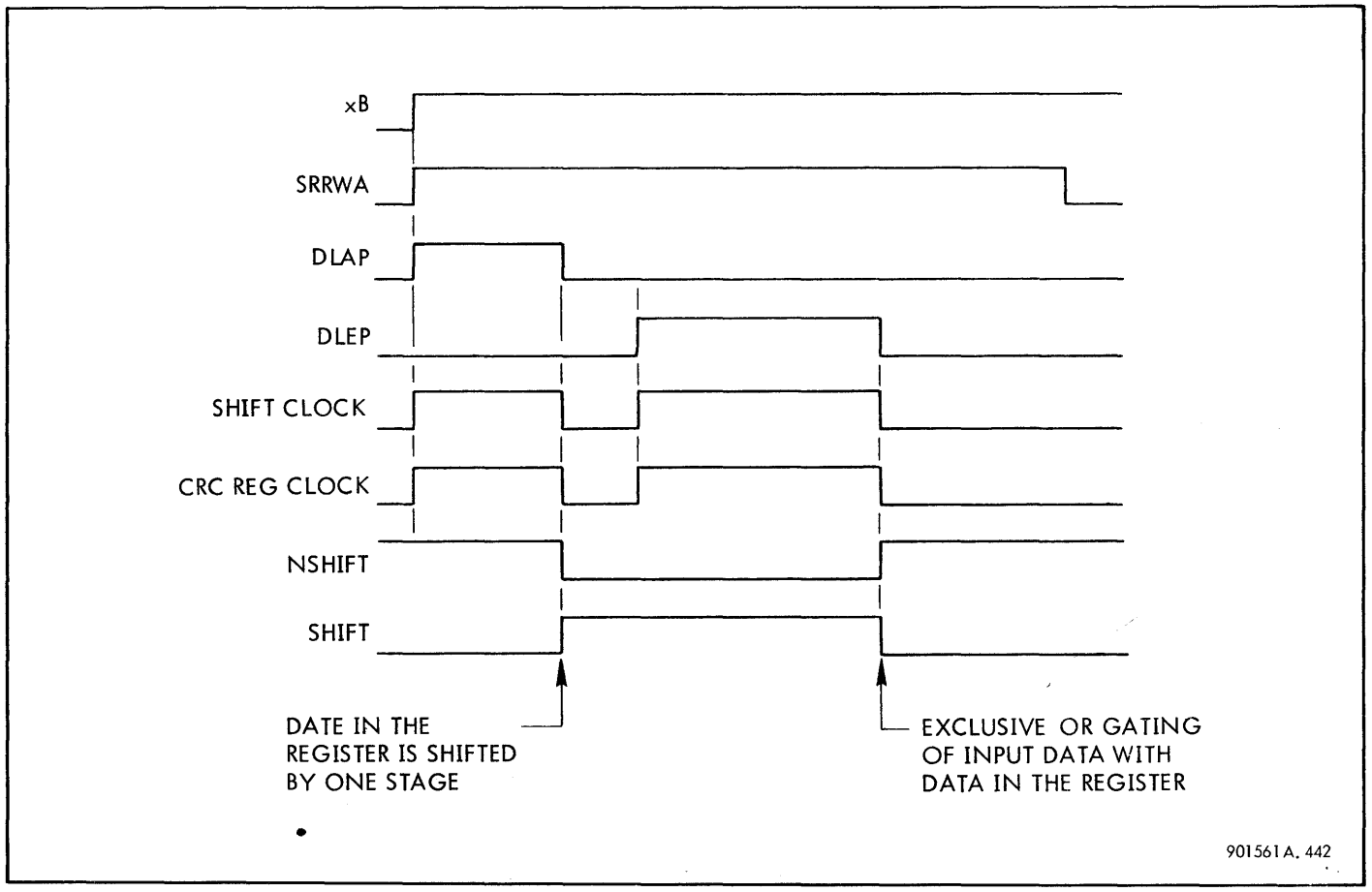


Figure 4-51. CRC Register (Read Forward) Exclusive OR and Shift, Timing Diagram

Table 4-5. Read Forward CRC and Write Register States

TYPE	INPUT SIGNALS (RR)									CRC REGISTER									WRITE REGISTER (LRC CHECK)								FUNCTION	
	P	0	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	P	0	1	2	3	4	5	6	7	Shift	Ex OR
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0										X	
										0	0	1	0	1	0	0	0	1	0	0	1	0	1	0	0	0	1		X
	1	1	1	1	1	1	0	0	1	1	0	0	0	1	0	1	0	0										X	
										0	1	1	1	0	1	1	0	1	1	1	0	1	0	1	0	0	0		X
	1	1	0	0	1	1	1	1	1	1	0	1	0	0	1	0	1	0										X	
										0	1	1	0	1	0	1	0	1	0	0	0	1	1	0	1	1	1		X
	0	0	0	0	1	0	1	0	1	1	0	1	0	1	0	1	1	0										X	
										1	0	1	0	0	0	0	1	1	0	0	0	1	0	0	0	1	0		X
	1	1	0	1	0	0	0	1	1	1	1	0	0	1	1	1	0	1										X	
										0	0	0	1	1	1	1	1	0	1	1	0	0	0	0	0	0	1		X
	1	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	1										X	
										1	1	1	1	0	1	1	0	0	0	0	1	1	1	1	0	1	0		X
	0	0	0	1	1	1	1	1	0	0	1	1	1	1	1	0	1	0										X	
										0	1	1	0	0	1	0	0	0	0	0	1	0	0	1	1	0	0		X
	0	0	0	1	0	1	0	1	0	0	0	1	1	0	0	1	0	0										X	
										0	0	1	0	0	1	1	1	0	0	0	1	1	0	0	1	1	0		X
DATA	1	0	1	0	0	0	1	1	1	0	0	0	1	0	0	1	1	1										X	
										1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1		X
	0	1	1	1	0	0	1	1	0	0	1	0	1	1	0	0	0	0										X	
										0	0	1	0	1	0	1	1	0	1	1	1	0	0	0	1	1	1		X
	0	0	1	1	1	1	1	0	0	0	0	0	1	0	1	0	1	1										X	
										0	0	1	0	1	0	1	1	1	1	1	0	1	1	0	1	1	1		X
	1	0	1	0	1	0	1	0	1	1	0	0	0	1	0	1	1	1										X	
										0	0	1	0	0	0	0	1	0	0	1	1	1	0	1	1	1	0		X
	0	1	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	1										X	
										0	1	0	1	0	1	1	1	0	0	0	1	1	0	0	0	0	1		X
	1	1	1	0	0	1	1	0	0	0	0	1	0	1	0	1	1	1										X	
										1	1	0	0	1	1	0	1	1	1	1	0	1	0	1	1	0	1		X
	0	1	1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	1										X	
										1	0	0	0	0	1	0	0	1	1	0	1	0	1	0	1	0	1		X
	1	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0	0	0										X	
										0	0	0	0	1	0	0	1	0	0	1	1	1	1	1	1	1	1		X
CRC	1	1	1	0	1	1	1	1	0	0	0	0	0	0	1	0	0	1										X	
										1	1	1	0	1	0	1	1	1	1	0	0	1	0	0	0	0	1		X
LRC	1	0	0	1	0	0	0	0	1	CORRECT									0	0	0	0	0	0	0	0	0		X
																			CORRECT										

NOTE: The E-register contains all zeros, since errors did not occur (not shown)

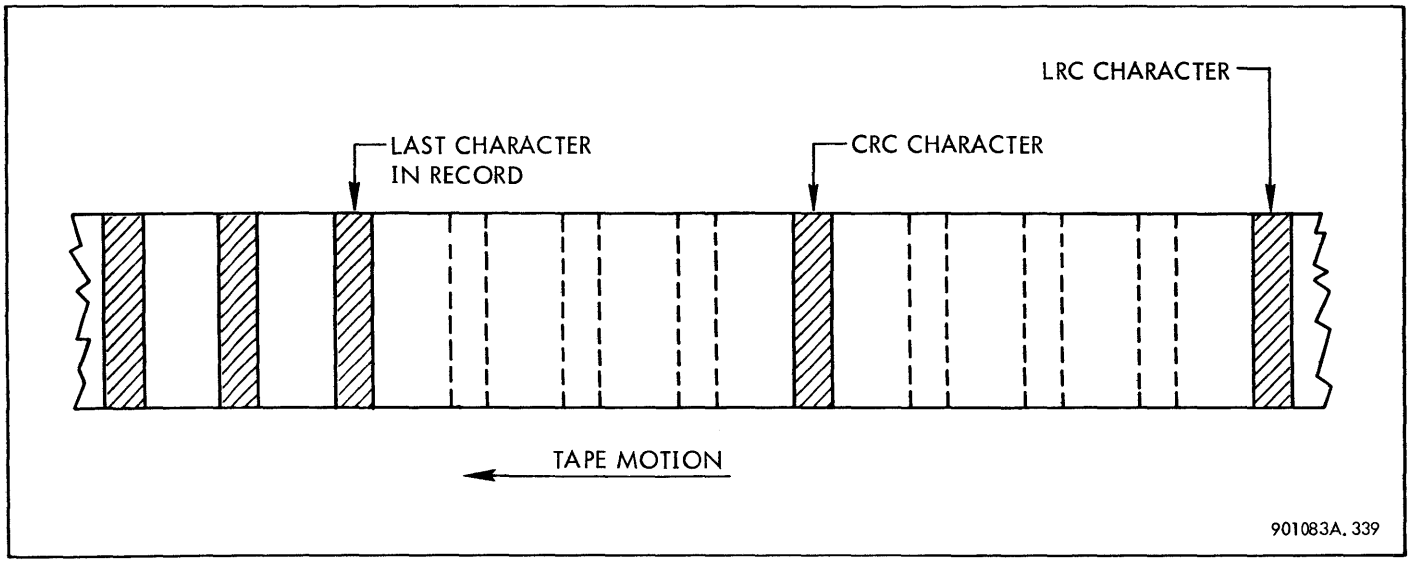


Figure 4-52. CRC and LRC Character Spacing

When the counter reaches a certain count, SG comes true. (SG is used as a control signal in the timing of the CRC and the LRC character detect circuits.) When SG goes true, it makes the clock input of flip-flop CRCT true (see figure 4-53 for a timing diagram).

S/CRCT = NCRCT

C/CRCT = SG

E/CRCT = RES4

RES4 = RES

When SG goes false, flip-flop CRCT is clocked and sets. CRCT is used in the clock input circuit of flip-flop LRCT.

S/LRCT = NLRCT

C/LRCT = (FWD SG) CRCT
+ REV (RES4 + CRCT SG)

E/LRCT = RES4

The CRC character is now read from the tape and is stored in the CRC register where the shift and the exclusive OR process is performed as previously explained. The CRC register at this time contains the final CRC character that is checked for validity. If the record is correct, the register contains the pattern 111010111.

When SG comes true for the second time, the clock inputs of CRCT and LRCT are made true, and when SG goes false, CRCT and LRCT are clocked. CRCT goes false, and LRCT goes true.

The LRC character is now read from the tape. It does not change the CRC register because CRC REG CLOCK is held false by the following term:

CRC REG CLOCK = (CRCT + NLRCT) + . . .

SG now comes true for the third time. When it comes true, it makes the clock input of CRCT true, and when it goes false, CRCT comes true. At this time, the validity of the CRC character is checked. The outputs of the flip-flops in the CRC register are connected to the set input of flip-flop CRCC as signal CROK.

S/CRCC = CROK

CROK = C1 C2 C3 NC4 C5 NC6 C7 C8 C9

C/CRCC = CCRC/CRCC CLOCK

CCRC/CRCC CLOCK = (NTMD FWD LRCT) SG
+ (NTMD REV NCRCT NLRCT) SG

E/CRCC = RES4

CCRC/CRCC CLOCK is also connected to the clock input of flip-flop CCRC.

S/CCRC = NCCRC

C/CCRC = CCRC/CRCC CLOCK

E/CCRC = RES4

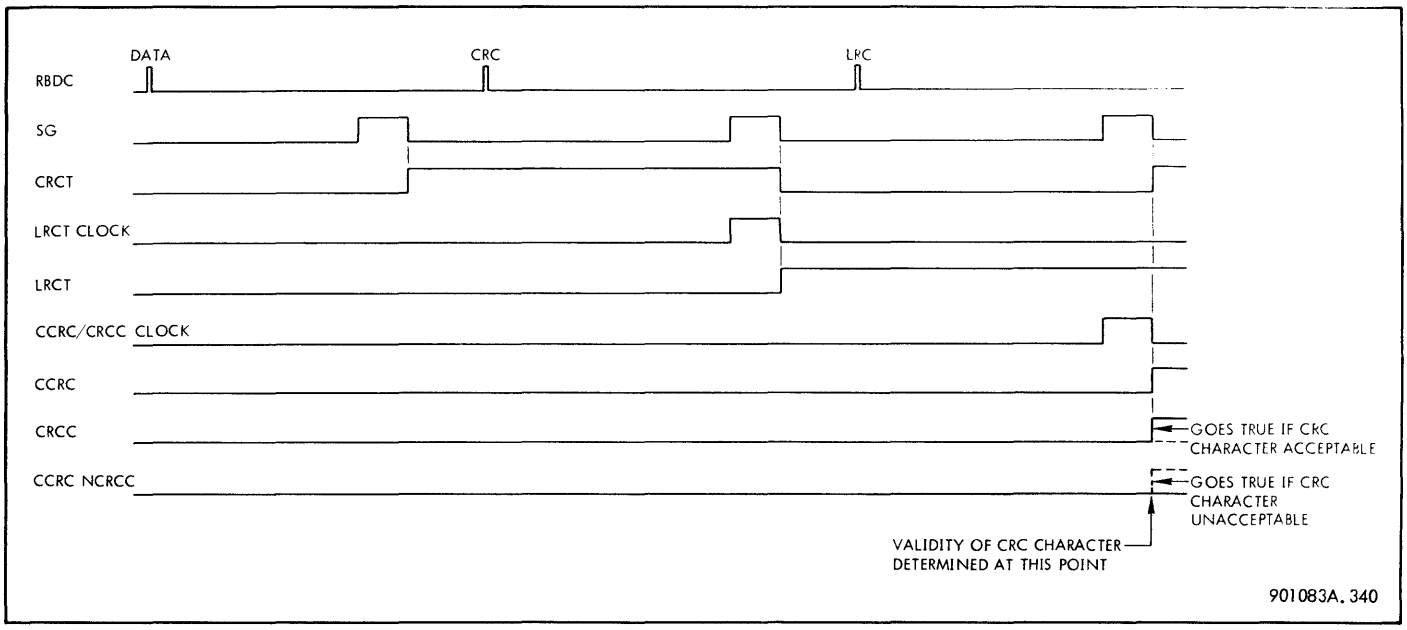


Figure 4-53. CRC and LRC Control Signals, Timing Diagram

The reset output of CRCC and the set output of CCRC are AND-gated to generate CCRC NCRCC. Figure 4-54 shows a simplified logic diagram of the CRC verification circuit.

If the pattern in the CRC register is correct, CROK is true, and the set input of CRCC is true. When SG comes true for the third time, CCRC/CRCC CLOCK is true; when SG goes false, CRCC/CRCC CLOCK goes false and clocks CRCC and CCRC. If the CRC character is correct, CRCC sets and NCRCC goes false. CCRC sets regardless of the state of the CRC character, and CCRC comes true. If the CRC character is acceptable, CCRC NCRCC remains false, if it is not acceptable, CCRC NCRCC goes true.

The outputs from the flip-flops in the CRC register are also connected to the CRC error comparator; there they are compared with the flip-flop outputs from the error pattern register. CMPR is true if the character in the CRC register and the character in the error pattern register agree. The pattern in the CRC register must be 111010111, and the pattern in the error pattern register must be 000000000 to get a comparison through the error comparator. (See figure 4-55.)

CMPR = (NC1 E1) + (C1 NE1)
(NC2 E2) + (C2 NE2)
(NC3 E3) + (C3 NE3)
(C4 E4) + (NC4 NE4)
(NC5 E5) + (C5 NE5)
(C6 E6) + (NC6 NE6)
(NC7 E7) + (C7 NE7)
(NC8 E8) + (C8 NE8)
(NC9 E9) + (C9 NE9)

CMPR FWD = CMPR FOR

CMPR REV = CMPR NFOR

S/FOR = FWR

C/FOR = RES

R/FOR = REV

In a read forward operation, CMPR FWD is true when the CRC register is correct, and in a read backward operation, CMPR REV is true when the CRC register is correct. NCMPR is true any time that the CRC register does not have the correct pattern. NCMPR is the inverted output of CMPR.

4-68 CRC Register Operation, Read Forward – Track-In-Error Determination. Assume that, instead of a comparison, NCMPR comes true, indicating an error. The CRC, error pattern, and write registers are used together to determine which channel (or track) is in error. NCMPR causes CLKE to come true.

CLKE = (NCMPR + CMPR WRZ) CRCCMPRCLK
(CCRC NCRCC) READ2 NWR7

When CRCCMPRCLK comes true, CLKE comes true and causes WRITE REG CLOCK to come true. When CLKE goes false, WRITE REG CLOCK goes false and clocks the flip-flops in the write register. The set input of the first stage in the register (WRP) is held true by WRZ, and WRP is set when the flip-flops are clocked.

S/WRP = WRZ (CCRC NCRCC) + ...

(CCRC NCRCC) is true, since this example assumes a CRC error.

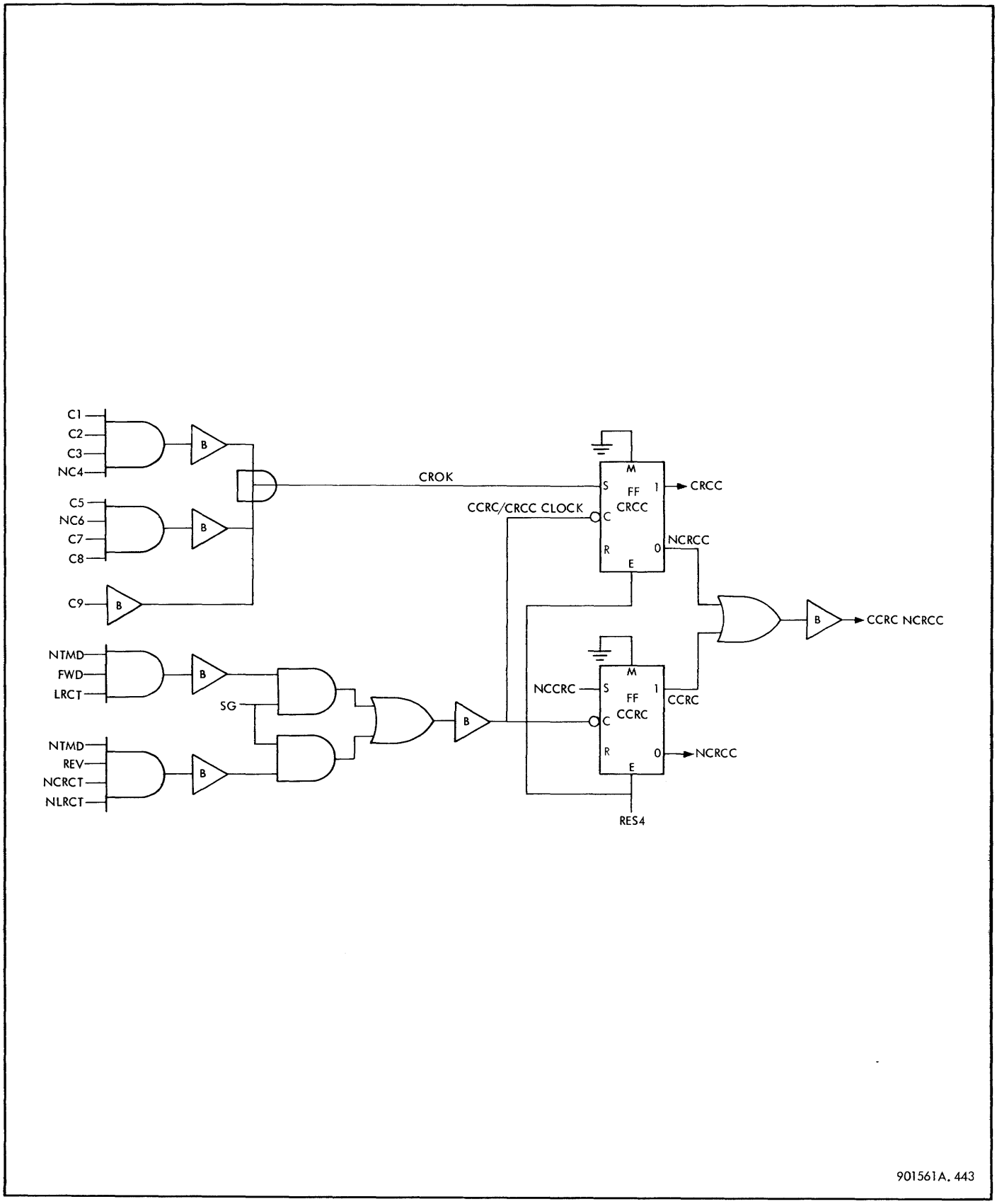


Figure 4-54. CRC Character Verification, Simplified Logic Diagram

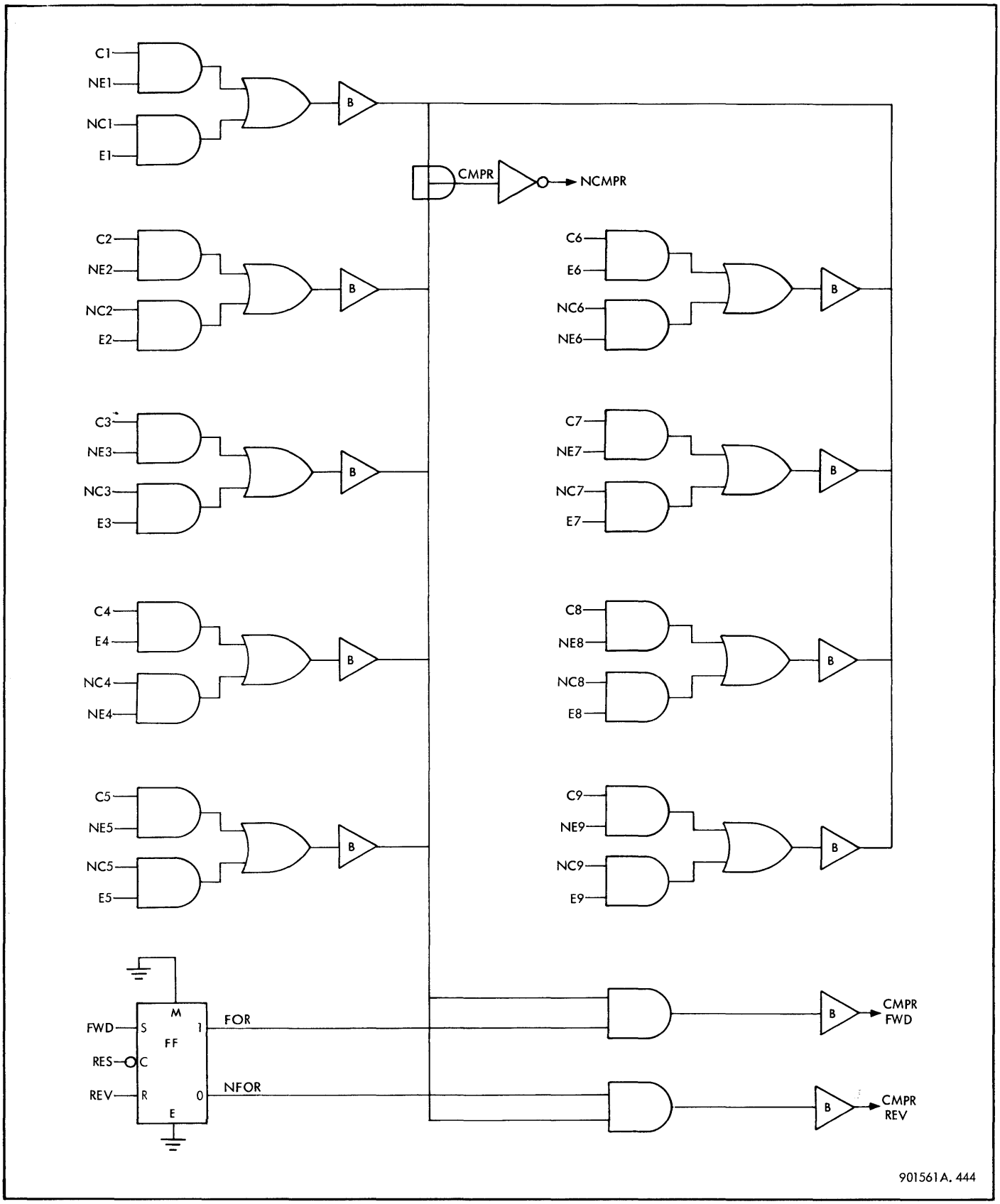


Figure 4-55. CRC Error Comparator, Simplified Logic Diagram

A comparison is again made between the CRC and the error pattern registers. If they do not compare, CLKE again comes true. This time CRC REG CLOCK and WRITE REG CLOCK are both made true.

CRC REG CLOCK = CLKE NWRZ + . . .

NWRZ is now true, since the write register is no longer reset. When CLKE goes false, the marker bit that is in channel WRP of the write register is shifted by one stage, and the pattern that is in the CRC register is shifted by one stage. The shift control signals CCRC NCRCC for the write register and NSHIFT for the CRC register are true.

A comparison is made again between the CRC and the EP-registers. If they compare, the marker bit that is in the write register is used to indicate the channel (or track) where the error is located, and the comparison process is terminated. If they do not compare, the process is repeated until they do or until the marker bit has been shifted completely through the write register without a comparison. If the latter occurs, WR7 comes true and is AND-gated with NCMPR and CCRC to generate signal SNCRE.

SNCRE = (WR7 NCMPR) CCRC + . . .

SNCRE is connected to the set input of flip-flop NCRE.

S/NCRE = SNCRE

C/NCRE = READ1 CRE

READ1 = READ

READ = 03F NF01 NWRT

E/NCRE = RESIN

RESIN = RSTB + BAND05

BAND05 = GBS FSD

When READ1 goes false, NCRE sets and NCRE comes true. NCRE is AND-gated with DATE which is true and causes NCORE to come true. See tables 4-6 and 4-7 for examples of a CRC validity check and a track-in-error determination process in a read forward operation, and see figure 4-56 for a timing diagram.

In table 4-6, the error is located in channel 2. Note that as each character is gated as an exclusive OR into the CRC register, the pattern in the error pattern register is shifted. The final character in the CRC register is 011000111, and the final character in the error pattern register is 100110100. The character that is in the EP-register is changed in the error comparator circuits to 011100011 for the comparison process (all bits with the exception of E4 and E6 are complemented). The character in the CRC register now is shifted until it has the pattern 011100011. (See table 4-7.) As shown in the table, the CRC register is shifted five times when the two registers agree. The marker bit in the write register has also been shifted five times, and is now in channel WR4. This marker bit is now used to identify the channel in error and to generate a track-in-error byte (which consists of only one bit) that is used to correct the channel in error.

The following cross reference can be used to determine the track in error from the position of the marker bit in the write register.

Operation	Channels
Write Register	WR P 0 1 2 3 4 5 6 7
Track In Error (Read forward)	DA 7 6 5 4 3 2 1 0 P
Track In Error (Read backward)	DA P 0 1 2 3 4 5 6 7

Table 4-6. CRC Validity Check in a Read Forward Operation

TYPE	READ REGISTER INPUT SIGNALS									CRC REGISTER									ERROR PATTERN REGISTER									FUNCTION	
	P	0	1	2*	3	4	5	6	7	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	Shift	Ex OR
DATA	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0										X	
										0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0		X
	1	1	1	[0]	1	1	0	0	1	1	0	0	0	1	0	1	0	0										X	
										0	1	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	1		X
	1	1	0	0	1	1	1	1	1	1	0	1	0	1	1	0	1	0										X	
										0	1	1	0	0	0	1	0	1	1	0	0	1	1	1	1	0	0		X
	0	0	0	[1]	1	0	1	0	1	1	0	1	0	1	1	1	1	0										X	
										1	0	1	1	0	1	0	1	1	0	1	0	0	0	1	1	1	1		X

*When channel 2 is in error, [ ] designates the error

†If there were no errors in the record, the final CRC character would have been 111010111, and the final error pattern character would have been 000000000.

(Continued)

Table 4-6. CRC Validity Check in a Read Forward Operation (Cont.)

TYPE	READ REGISTER INPUT SIGNALS									CRC REGISTER									ERROR PATTERN REGISTER									FUNCTION	
	P	0	1	2*	3	4	5	6	7	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	Shift	Ex OR
DATA										1	1	0	0	0	1	0	0	1										X	
	1	1	0	1	0	0	0	1	1	0	0	0	1	0	1	0	1	0	1	0	1	1	1	0	0	1	1		X
										0	0	0	0	1	0	1	0	1										X	
	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	0	1	1	0	0	0	0	1	0	1		X
										0	1	1	1	1	0	0	1	1										X	
	0	0	0	[0]	1	1	1	1	0	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1		X
										1	0	1	0	0	1	0	1	0										X	
	0	0	0	[0]	0	1	0	1	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0		X
										0	1	0	1	0	0	0	0	0										X	
	1	0	1	[1]	0	0	1	1	1	1	1	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0		X
										1	1	1	1	0	1	1	1	1										X	
	0	1	1	1	0	0	1	1	0	1	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	0	0		X
										1	1	0	1	0	0	0	0	0										X	
	0	0	1	[0]	1	1	1	0	0	1	1	1	1	1	1	1	0	0	0	0	0	1	1	1	0	0	1		X
										0	1	1	1	1	1	1	1	0										X	
	1	0	1	0	1	0	1	0	1	1	1	0	1	0	1	0	1	1	1	0	0	1	0	0	0	0	0		X
										1	1	1	1	0	1	0	0	1										X	
	0	1	0	0	0	1	1	1	1	1	0	1	1	0	0	1	1	0	0	1	0	0	1	0	0	0	0		X
										0	1	0	1	1	0	0	1	1										X	
	1	1	1	0	0	1	1	0	0	1	0	1	1	1	1	1	1	1	0	0	1	0	0	1	0	0	0		X
										1	1	0	0	0	0	0	1	1										X	
	0	1	1	1	1	1	0	0	0	1	0	1	1	1	1	0	1	1	0	0	0	1	0	0	1	0	0		X
										1	1	0	0	0	0	0	0	1										X	
	1	1	0	[0]	0	1	0	1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	1	1		X
CRC†										1	0	0	1	1	1	0	0	1										X	
	1	1	1	[1]	1	1	1	1	0	0	1	1	0	0	0	1	1	1	1	0	0	1	1	0	1	0	0		X

*When channel 2 is in error, ☐ designates the error

†If there were no errors in the record, the final CRC character would have been 111010111, and the final error pattern character would have been 000000000

(Continued)

Table 4-7. Track-In-Error Determination – Read Forward

COMPARE SIGNAL	CRC REGISTER	ERROR PATTERN REGISTER	WRITE REGISTER	SHIFT FUNCTION
	1 2 3 4 5 6 7 8 9	1 2 3 4 5 6 7 8 9	P 0 1 2 3 4 5 6 7	
NCMPR	0 1 1 0 0 0 1 1 1	1 0 0 1 1 0 1 0 0	1 0 0 0 0 0 0 0 0	
	1 0 1 0 1 1 1 1 1	Complement all but E4 and E6 for comparison	0 1 0 0 0 0 0 0 0	X
	1 1 0 0 1 0 0 1 1		0 0 1 0 0 0 0 0 0	X
	1 1 1 1 1 0 1 0 1	0 1 1 1 0 0 0 1 1	0 0 0 1 0 0 0 0 0	X
	1 1 1 0 0 0 1 1 0		0 0 0 0 1 0 0 0 0	X
CMPR	0 1 1 1 0 0 0 1 1	0 1 1 1 0 0 0 1 1	0 0 0 0 0 1 0 0 0	X

Since the marker bit is in channel WR4 of the write register, a cross-reference shows that the error is actually in channel 2, which agrees with the table. A sense byte is now generated by the IOP from the marker bit and is used to insert a correction bit into the correct DAx line of the error pattern register during a read with correction process.

4-69 CRC Register Operation, Read Backward – Track-In-Error Determination. In a read backward operation, the process of gating each character into the CRC, and the error pattern registers is the same as explained for the read forward operation with one exception. The CRC character is the first character to be sampled for the registers. The parity of the CRC character in a read forward operation is compared to the number of bytes in a record that has already been read. This cannot be accomplished in a read backward operation, since the record has not been read when the CRC character is sampled. Therefore, the CRC character is stored in the CRC register but is not stored in the error pattern register. This does not affect the operation of the CRC character validation and the track-in-error determination unless the CRC character should have a parity error itself. In that case, it is impossible to get a comparison between the CRC and the EP-registers. The result is a noncorrectable read error indication. Also, in a read backward operation, the CRC character itself cannot be corrected by the sense byte correction bit; this limitation does not exist in a read forward operation.

Refer to tables 4-8 and 4-9 for a typical representation of a CRC validity check and a track-in-error determination process for a read backward operation. Note that the error is again in channel 2 in table 4-8. The final CRC character is 111100101, and the final character in the error pattern register is 010011000. The character that is in the EP-register is complemented to 101001111, as was done in a read forward. The CRC register is shifted as before. After three shifts, the two registers agree. The marker bit is in channel WR2 of the write register. Using the cross-reference supplied previously, this establishes the error in channel 2, which agrees with table 4-8. A sense byte is generated and is used to correct channel 2 during a read with correction process.

4-70 CRC Register Operation, Read With Correction. Assume that a comparison between the CRC register and the error pattern register has been made. The CRC, EP, and write registers are reset, and a set correction order is made. The set correction order causes the set input of the read with correction flip-flop FCR to go true.

S/FCR = BAND18 + SFCROFRS

BAND18 = SCO NFU3

SCO = 01F02U SCOD

SFCROFRS = (NBOR06 + NESR) 01F 00U

C/FCR = CFCRX RSD + CFCRY CLK

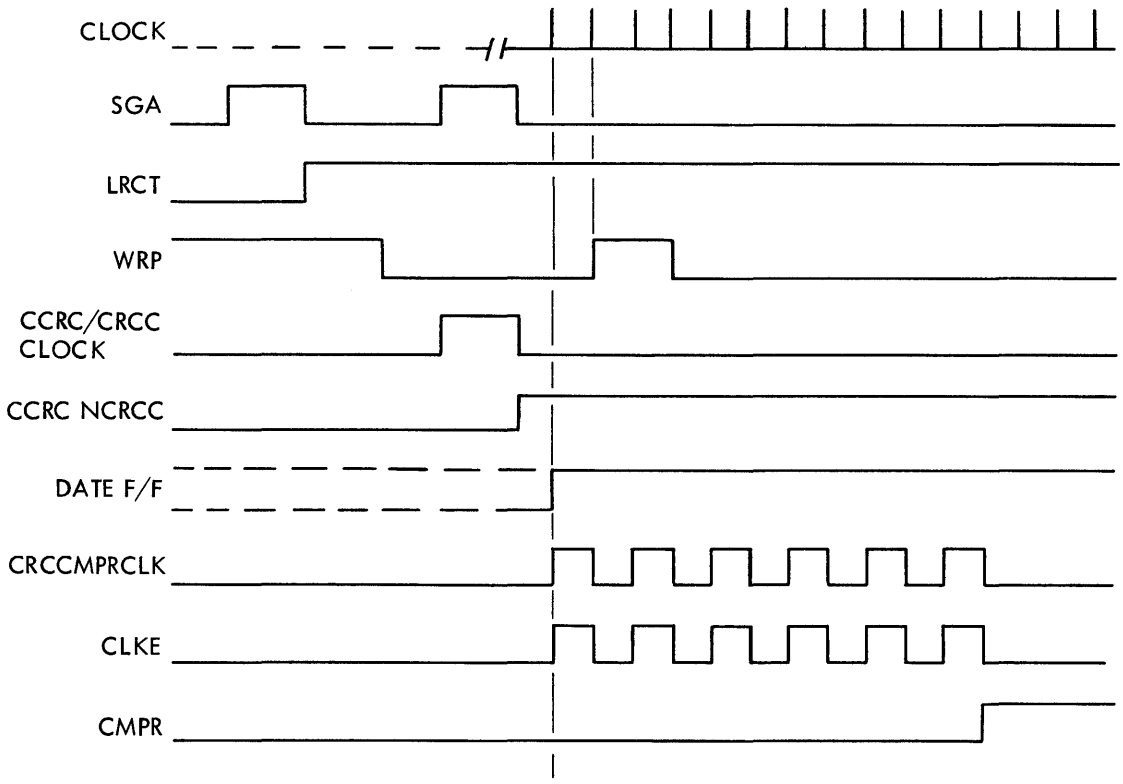
CFRX = BAND18 + (FCR 01F00U)

CFCRY = 03F00U + DSTWRT

When RSD goes false, the FCR flip-flop is clocked and sets. The IOP has been given the track-in-error byte and has raised the correct DAx line on the input of the error pattern register. The DAx line is AND-gated with LD SELECT which is true.

LD SELECT = NWRITE1

This is the only input gate in the EP-register that has been made true, and the output of this gate is inverted to NDCx and is AND-gated with EPRL and with the set output of that stage in the register. NDCx is inverted to DCx and is



NOTES:

1. THE FIRST CLKE LOADS WRP WITH THE MARKER BIT; NO SHIFT IN CRC REGISTER
2. THE FOLLOWING CLKE 'S SHIFT THE WR AND THE EP REGISTERS
3. CMPR IS FOUND ONLY AFTER THE 5TH SHIFT, TRACK IN ERROR IS IN THE WR; NO MORE CLKE'S

901561A. 445

Figure 4-56. Track-In-Error Byte Generation, Timing Diagram

Table 4-8. CRC Validity Check in a Read Backward Operation

TYPE	READ REGISTER INPUT SIGNALS 7	6	5	4	3	2*	1	0	P	CRC REGISTER 1	2	3	4	5	6	7	8	9	ERROR PATTERN REGISTER 1	2	3	4	5	6	7	8	9	FUNCTION Shift	Ex OR
CRC[t]	0	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0										X	
										0	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0		X
DATA	0	1	0	1	0	[0]	0	1	1	1	0	1	0	0	0	1	1	1										X	
										1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1		X
	0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	0	1	0										X	
										0	1	1	0	0	1	1	0	0	1	0	0	1	1	1	1	0	0		X
	0	0	1	1	0	0	1	1	1	0	0	1	1	0	0	1	1	0										X	
										0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0		X
	1	1	1	1	0	0	0	1	0	1	0	0	1	1	1	1	0	0										X	
										0	1	1	0	1	1	1	1	0	0	0	1	0	0	1	1	1	1		X
	1	0	1	0	1	0	1	0	1	0	0	1	1	0	1	1	1	1										X	
										1	0	0	1	1	1	0	1	0	1	0	0	0	1	1	0	1	1		X
	0	0	1	1	1	[0]	1	0	0	0	1	0	0	1	1	1	0	1										X	
										0	1	1	1	0	1	0	0	1	1	1	0	1	1	0	0	0	0		X
	0	1	1	0	0	1	1	1	0	1	0	1	0	0	1	0	0	0										X	
										1	1	0	0	0	0	1	1	0	0	1	1	0	0	0	1	1	0		X
	1	1	1	0	0	[1]	1	0	1	0	1	1	0	0	0	0	1	1										X	
										1	0	0	0	0	1	1	1	0	0	0	1	1	0	1	1	0	1		X
	0	1	0	1	0	[0]	0	0	0	0	1	0	0	0	0	1	1	1										X	
										0	0	0	1	0	0	1	1	1	1	0	0	0	0	1	0	1	1		X
	0	1	1	1	1	[0]	0	0	0	1	0	0	1	0	1	1	1	1										X	
										1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	0	0		X
	1	1	0	0	1	1	1	1	1	1	1	1	0	1	0	0	1	1										X	
										0	0	1	0	0	1	1	0	0	0	1	1	0	1	1	1	0	0		X
	1	1	0	0	0	1	0	1	1	0	0	0	1	0	0	1	1	0										X	
										1	1	0	1	0	1	1	0	1	0	0	1	1	0	1	1	1	0		X
	1	0	1	0	1	[1]	0	0	0	1	1	1	1	0	1	0	1	0										X	
										0	1	0	1	1	0	0	1	0	0	0	0	1	1	0	1	1	0		X

*When channel 2 is in error, [ ] designates the error

[t]If there were no errors in the record, the final CRC character would have been 111010111 and the final EP character would have been 000000000. Note that there is no parity error indication for the CRC character in a read backward operation

(Continued)

Table 4-8. CRC Validity Check in a Read Backward Operation (Cont.)

TYPE	READ REGISTER INPUT SIGNALS									CRC REGISTER									ERROR PATTERN REGISTER									FUNCTION	
	7	6	5	4	3	2*	1	0	P	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	Shift	Ex OR
DATA	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1										X	
										1	1	0	1	0	1	0	1	0	0	0	0	0	1	1	0	1	1		X
	1	0	0	1	1	[0]	1	1	1	0	1	1	0	1	0	1	0	1										X	
										1	1	1	1	0	0	0	1	0	1	0	0	1	1	0	0	0	0		X
	1	0	0	0	1	0	1	0	0	0	1	1	1	1	0	0	0	1										X	
										1	1	1	1	0	0	1	0	1	0	1	0	0	1	1	0	0	0		X

*When channel 2 is in error, [ ] designates the error

†If there were no errors in the record, the final CRC character would have been 111010111 and the final EP character would have been 000000000. Note that there is no parity error indication for the CRC character in a read backward operation

Table 4-9. Track-In-Error Determination, Read Backward Operation

CRC REGISTER									ERROR PATTERN REGISTER	WRITE REGISTER									FUNCTION SHIFT
1	2	3	4	5	6	7	8	9	1 2 3 4 5 6 7 8 9	P	0	1	2	3	4	5	6	7	
1	1	1	1	0	0	1	0	1	0 1 0 0 1 1 0 0 0	1	0	0	0	0	0	0	0	0	
1	1	1	0	0	1	1	1	0	Complement all but E4 and E6 for comparison	0	1	0	0	0	0	0	0	0	X
0	1	1	1	0	0	1	1	1		0	0	1	0	0	0	0	0	0	X
1	0	1	0	0	1	1	1	1	1 0 1 0 0 1 1 1 1	0	0	0	1	0	0	0	0	0	X

AND-gated with EPRL and with the reset output of that stage in the register. These two gates are connected in an exclusive OR configuration and, when the CRC register shifts the first time in this operation, EP REG CLOCK triggers the flip-flop and sets it.

EPRL = FCR READ1

EP REG CLOCK = LD NSHIFT

LD = SCO NFU3 RSD1

This flip-flop in the EP-register remains set for the duration of the read with correction process.

The error pattern register does not accept any parity error bits during this operation, since control signal EPRS is held false.

EPRS = READ1 NFCR

When a parity error is detected, PE FCR in the parity error circuits goes true. PE FCR is AND-gated with the appropriate flip-flop output from the EP-register at each of the nine data bus circuits. The only gate that is true, however, is the one that has been made true by the one flip-flop that is set in the EP-register. The output of this gate (PE FCR Ex) is AND-gated with the reset output of the appropriate flip-flop in the read register, instead of with the set output (which would be used for a bit that does not require correction). The output of this gate (RRCx) now contains the

required correction bit. This bit, which is the complement of the bit that is present in a read without correction, is routed through the normal paths to the eight-byte buffer memory.

This operation can be performed in either a read forward or a read backward operation. The only difference is that the CRC character itself cannot be corrected in a read backward.

4-71 TAPE MARK RECORD, READ. A tape mark record separates adjacent files of information. The tape mark record follows the last LRC character in the file and consists of a tape mark character which is equivalent to a hexadecimal 13, followed by an LRC character, which is a hexadecimal 13 (000010011). See figure 4-71 for a representation of the tape mark record.

4-72 Tape Mark Record, Read Forward. When the tape mark character is read, the assembly period counter does not reset but continues to count. The read register contains the tape mark character (NRR0, NRR1, NRR2, RR3, NRR4, NRR5, RR6, RR7, and NRRP). The read register outputs are AND-gated and generate signal TMC. TMC is inverted to NTMC (which is false after the tape mark character is stored in the read register).

TMC = NRR0 NRR1 NRR2 RR3 NRR4 NRR5 RR6 RR7 NRRP (LRCT + NODD)

NTMC is AND-gated with RCP and generates signal NTMD.

NTMD = NTMC RCP + NRES NTMD

NTMD is inverted to TMD, which is true when a tape mark character is stored in the read register. TMD is connected to the set input of flip-flop LTMD.

S/LTMD = TMD

C/LTMD = SGB

When the assembly period counter reaches a certain count, SGA comes true and makes SG come true (see figure 4-57 for a timing diagram).

SG = SGA + SGB

SGA = APC5 NAPC6 NAPC7 APC8

When SG comes true, it makes the clock input of flip-flop CRCT true. When SG goes false, CRCT is clocked and sets.

S/CRCT = NCRCT

C/CRCT = SG

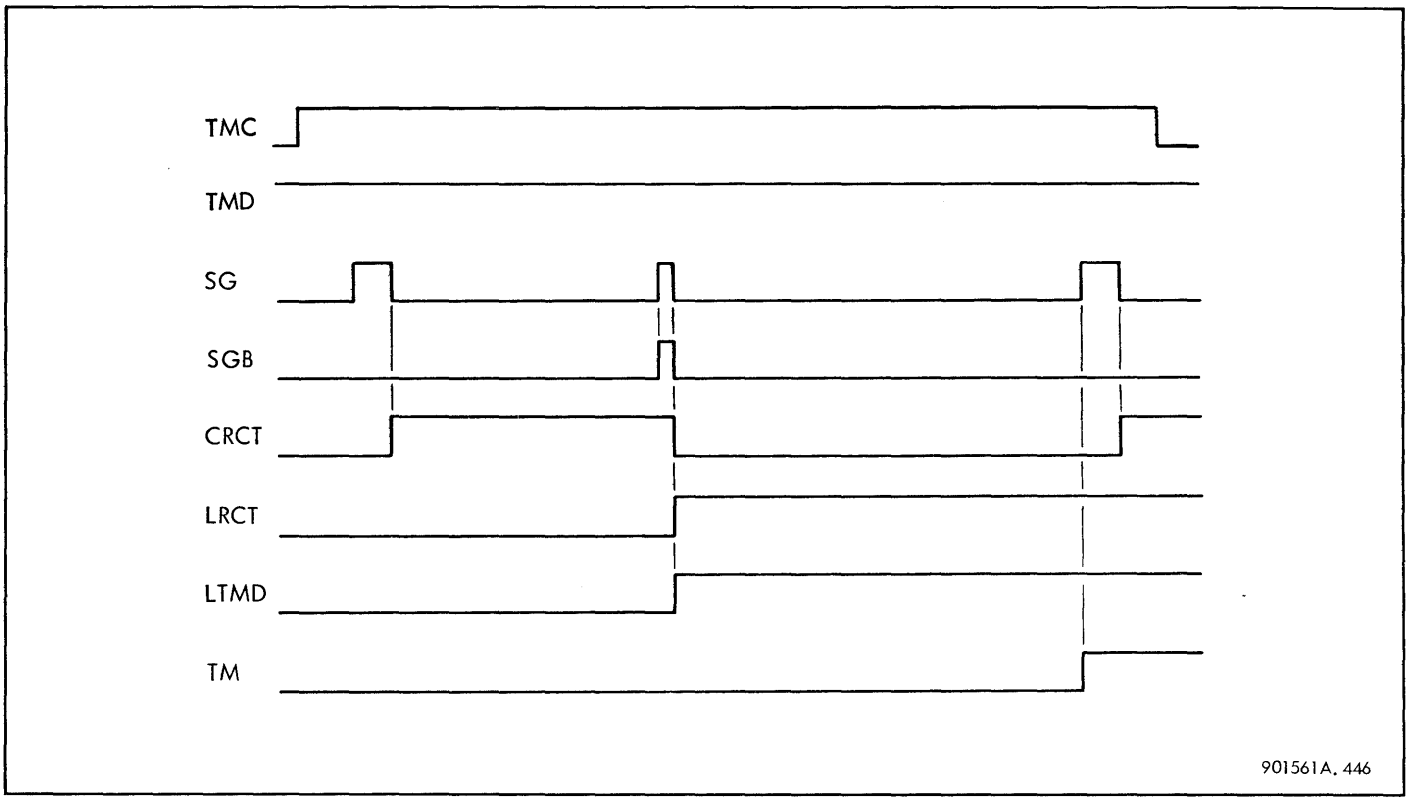


Figure 4-57. Tape Mark Record (Read Forward), Timing Diagram

CRCT is AND-gated with FWD and SG and makes the clock input of flip-flop LRCT true.

S/LRCT = NLRCT

C/LRCT = CRCT (FWD SG) + REV (RES4 + CRCT SG)

RES4 = RES

RES = 01F03U NDPR

There are seven blank spaces between the tape mark character and the LRC character that follows. SGB comes true after SG and initializes the assembly period counter.

SGB = APC5 APC6 NAPC8 CRCT CLA

SGB causes SG and the clock input of LTMD to come true. When SGB goes false, it clocks and sets LTMD and causes SG to go false. When SG goes false, it resets CRCT and sets LRCT.

The LRC character is read in the eighth character space after the tape mark character. The LRC character (a hexadecimal 13) is stored in the read register and causes TMC to come true and NTMC to go false. NTMC causes NTMD to go false and TMD to go true. SG comes true after the LRC character has been read and is used to dc-set flip-flop TM.

M/TM = (SG LTMD) TMD + . . .

When SG comes true, it also makes the clock input of CRCT true; when it goes false, CRCT is clocked and sets.

4-73 Tape Mark Record, Read Backward. When reading backward, first the LRC character and then the tape mark character are read. The sequence of events is identical to that of a read forward operation, except that LRCT is set initially instead of being reset. See figure 4-58.

When the read backward operation is started, RES4 momentarily comes true and clocks LRCT, and LRCT sets.

S/LRCT = NLRCT

C/LRCT = REV RES4 + CRCT SG

When SG comes true after the LRC character is read, the clock input of CRCT is made true; when SG goes false, CRCT is clocked and sets.

When SGB comes true, it makes SG and the clock input of LTMD true. SG is AND-gated with CRCT at the clock input of LRCT and makes it true. When SGB goes false, SG goes false; LRCT and CRCT are clocked and reset, and LTMD is clocked and set.

The tape mark character is read and is stored in the read register. TMC comes true, and NTMC goes false. NTMC causes NTMD to go false and TMD to go true. SG comes true after the tape mark character is read and is used to dc-set flip-flop TM. When SG comes true, it also makes the clock input of CRCT true; when it goes false, CRCT is clocked and set.

M/TM = (SG LTMD) TMD + REV BOTC

C/TM = RES3 TM CLOCK

E/TM = RESIN

RES3 = RES

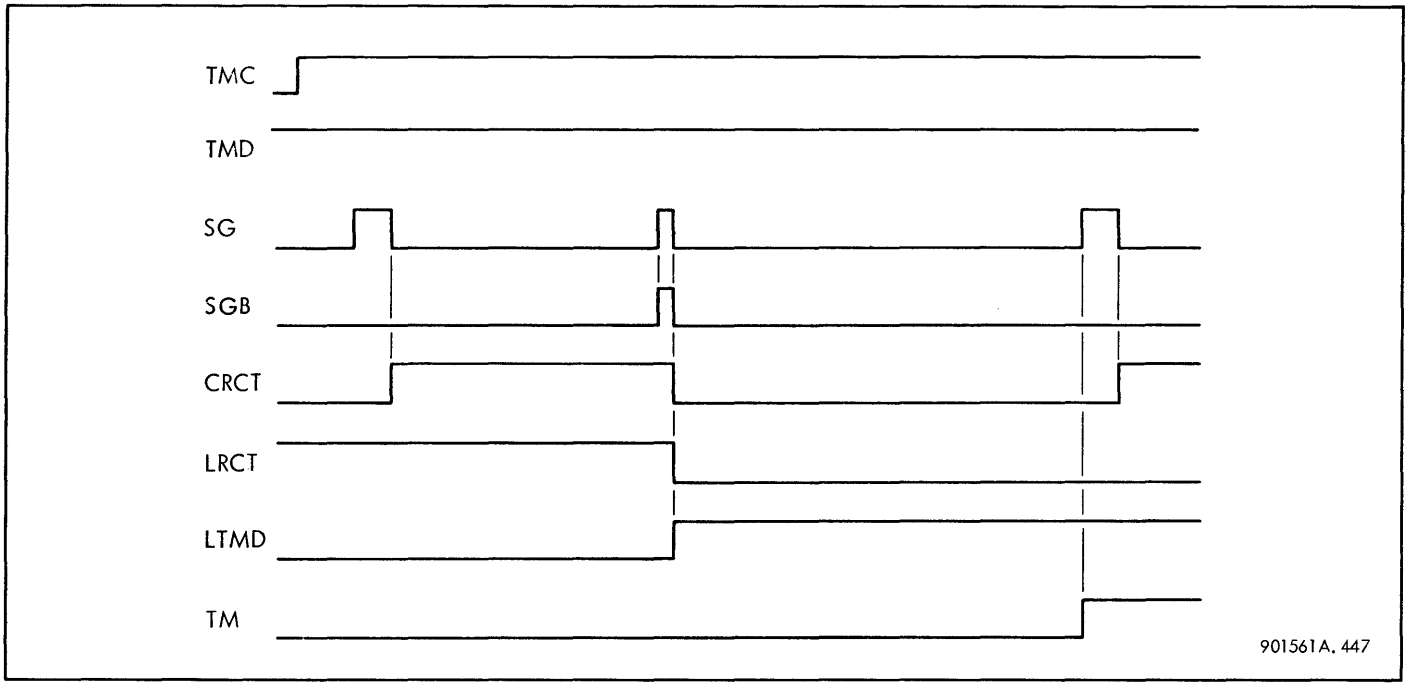


Figure 4-58. Tape Mark Record (Read Backward), Timing Diagram

4-74 Tape Mark Detection on Read Order. When the controller advances to state 03F03U-03U and when WRT and F01 are false, the controller enters the read phase. If TM comes true, FSC remains false, and unusual end flip-flop FUN sets.

S/FUN = TM READ FU1 + ...

C/FUN = TM READ FU1 CLK + ...

TM is connected to the set and the clock inputs of FUN. When TM comes true, it makes the set input of FUN true. When CLK comes true, it makes the clock input of FUN true, and when CLK goes false, it clocks and sets FUN.

FUN is connected to the clock input of FU1. When CLK comes true, it makes the clock input of FU1 true. When CLK goes false, it clocks and resets FU1. The controller now advances to state 03F01U from a tape mark.

C/FU1 = FUN CLK NFSCC 03F + ...

4-75 READ AFTER WRITE. During a write operation, information written on the tape is immediately read back for checking. The error pattern register is used to indicate errors detected in the record after the LRC character has been read. The CRC register is not used to check the CRC character during a read-after-write operation, since it is required to generate the CRC character to be recorded.

The read-after-write operation is very similar to that of a normal read operation. The read heads and the read amplifiers in the station detect the data that has just been recorded on the tape and send it to the input of the controller.

The data received from the station is presented to the peak detector counter in the controller. When the input signal (RDAMP) goes true, it causes the set input of the first stage in the peak detector counter to go true. See figure 4-59 for a simplified block diagram of the read-after-write operation.

When RDAMP goes true, the counter starts its count cycle. A pulse width detector circuit (PDQ) is used to stop the count cycle if the counter has started to count on a short duration noise spike instead of on a data bit. PDQ requires a count of seven in the register, then comes true on the eighth clock signal and allows the counter to continue. In a read-after-write operation, PDQ requires a count of 12 in the register, then comes true when the 13th clock signal goes false.

S/PDQ = WRITE2 N1PKDET N2PKDET 3PKDET 4PKDET NPDT + ...

C/PDQ = CLx

When PDQ goes true, the counter continues its count cycle. PDT comes true, and dc sets the bit crowding counter that follows to one-half the count contained in the peak detector counter.

The operation of the other counters and registers in the read circuitry is identical to that of the read operation, except for the error pattern register. The reader is referred to those paragraphs in this section that are applicable if a detailed circuit analysis is required.

The flip-flop outputs of the read register are presented to the input gating circuits of the error pattern register on the RRCx lines. RRCx is AND-gated with WRITE SELECT (see figure 4-37). The output of this gate is inverted to NDCx. NDCx is AND-gated with EPRL, which is true, and with the set output of that particular stage in the register. NDCx is inverted to DCx, and DCx is AND-gated with EPRL and with the reset output of that particular stage in the register. The outputs from these two gates are connected in an exclusive OR configuration and are connected to the set input of that particular stage in the register.

The flip-flops in the error pattern register now toggle at each input level change. After the LRC character has

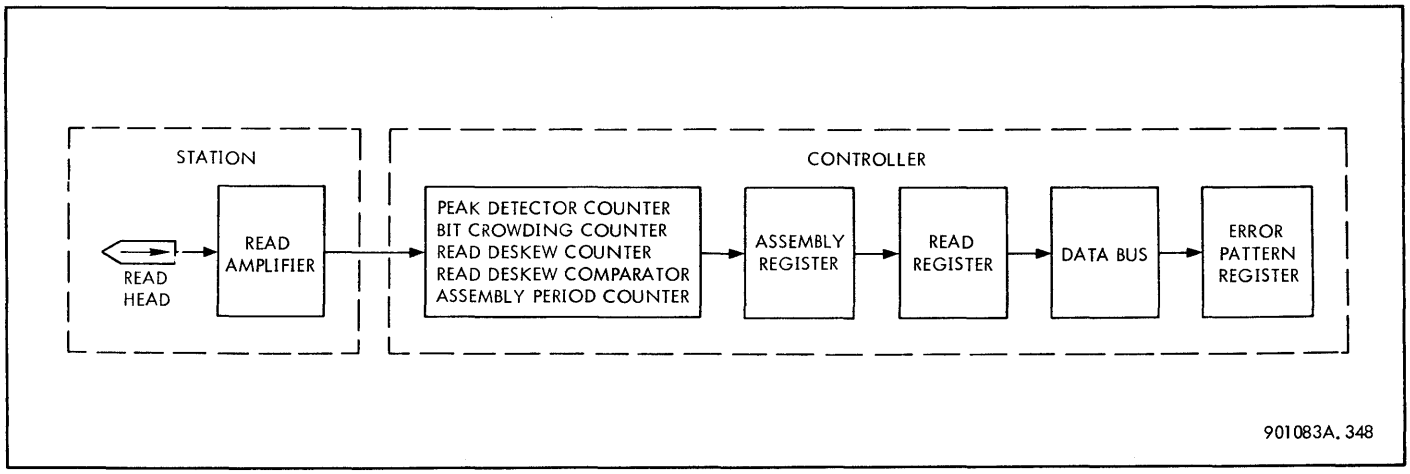


Figure 4-59. Read After Write, Simplified Block Diagram

been read and, if all characters in the record are correct (including the LRC character), each flip-flop in the register is reset, and ERZ is true.

ERZ = NE1 NE2 NE3 NE4 NE5 NE6 NE7 NE8 NE9

ERZ is used to indicate an error in the record. It is AND-gated with WRITE2 to generate NLRC ERROR. NLRC ERROR is inverted to LRC ERROR.

LRC ERROR = NWRITE NWRZ + NERZ WRITE2

If an error occurs in the record, ERZ is false, NLRC ERROR is false, and LRC ERROR is true.

4-76 LENGTH ERROR DETECTION ON READ ORDER. When the controller advances to state 03F03U-02U and when WRT and F01 are false, the controller enters the read phase. If FSC remains false and LG comes true, length error flip-flop FLE and finish flip-flop FIN set, and FU1 resets. This indicates a length error in which the byte count is longer than the record being read.

S/FLE = BAND24

BAND24 = FU1 READ

R/FLE = FLE

C/FLE = CFLE

CFLE = BAND24 LG NFSCC CLK

M/FLE = MFLE

MFLE = BMT READ 01U + READ 01U FUN

E/FLE = RSTB

S/FIN = LG + . . .

C/FIN = READ LG FU1 + . . .

C/FU1 = CLK LG READ NFSCC FU1 + . . .

LG is connected to the clock inputs of FLE and FIN and to the set input of FIN. When CLK comes true, it makes the clock inputs of FLE and FU1 true. When CLK goes false, it clocks and sets FLE and resets FU1. When FU1 goes false, it clocks and sets FIN.

The controller now advances to state 03F01U from a length error indication.

4-77 READ ORDER TERMINATION. RSD goes false when the IOP signals count done by raising DA1R. When RSD goes false, it clocks and sets FIN and clocks and resets FU1.

C/FIN = FU1 READ FU3 RSD + . . .

S/FIN = DA1R ESR + . . .

C/FU1 = FU1 READ FU3 RSD

See figure 4-42 for the timing sequence. The controller has now advanced to state 03F01U (order finalization) from the read phase.

RSD goes false if the IOP raises the halt command (DA3R). When RSD goes false, it clocks and sets FUN and FIN and clocks and resets FU1.

S/FUN = DA3R ESR NDST FU3 + . . .

C/FUN = FU3 NFUN NDST RSD1 + . . .

S/FIN = DA3R ESR + . . .

The controller now advances to state 03F01U from an unusual end condition.

4-78 Space File Order

When the controller advances to stage 03F03U-02U and it contains a space file order, WRT is false and F01 and SPF are true.

S/F01 = DA6R DA7R

SPF = BAND21 F05

BAND21 = 03F F01 F03 NF04

The tape continues to move until the tape mark record is detected. TM comes true. When CLK comes true, it makes the clock input of FU1 true; when it goes false, it clocks and resets FU1. See figure 4-60 for a timing diagram of the space file operation.

C/FU1 = CLK SPF TM + . . .

The controller now advances to state 03F01U on a space file order.

If TM does not come true on a space file order, LG comes true and causes RES to come true.

RES = NTM LG SPF + . . .

RES is used as a reset signal in the data logic circuits.

4-79 Space Record Order

WRT and SPF are false and SPR is true, when the controller advances to state 03F03U-02U and it contains a space record order.

SPR = BAND21 NF05

BAND21 = 03F F01 F03 NF04

LG is connected to the clock input of FU1. When LG and CLK come true, the clock input of FU1 is made true; when CLK goes false, FU1 is clocked and resets.

C/FU1 = SPR LG CLK + . . .

The controller now advances to state 03F01U on a space record order. See figure 4-61 for a timing diagram of the space record operation.

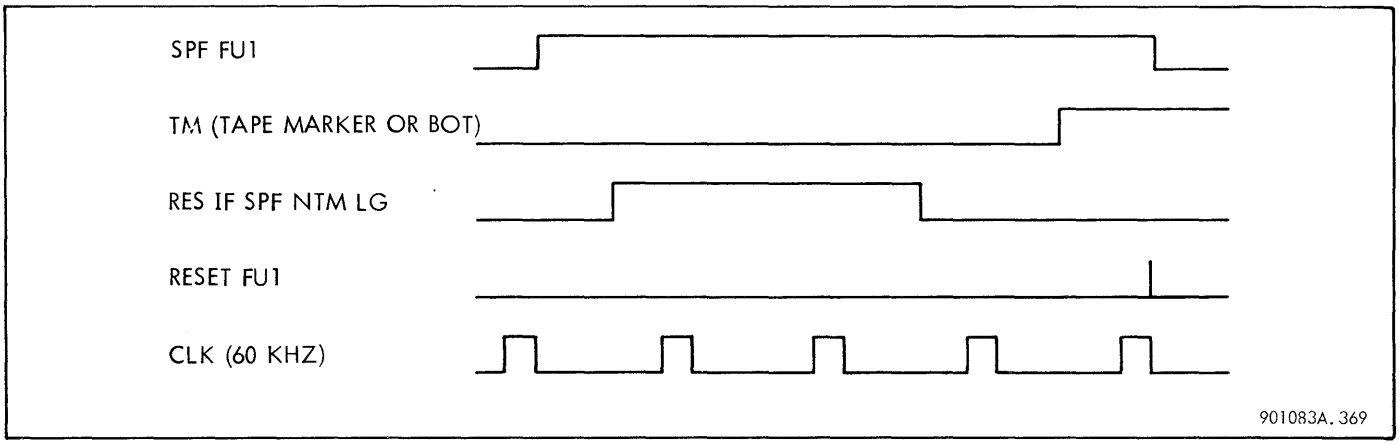


Figure 4-60. 03FFU1 State (Space File), Timing Diagram

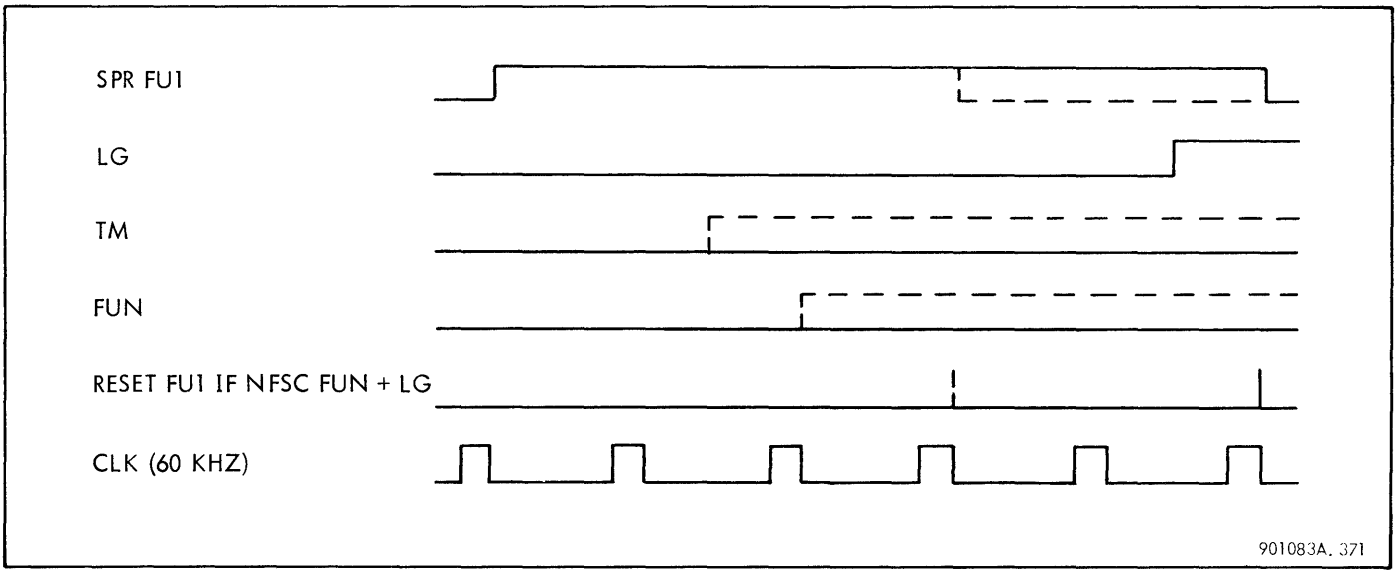


Figure 4-61. 03FFU1 State (Space Record), Timing Diagram

If TM comes true (before LG), unusual end flip-flop FUN sets.

S/FUN = TM READ FU1 + . . .

C/FUN = TM READ FU1 CLK + . . .

TM is connected to the set and the clock inputs of FUN. When TM comes true, it makes the set input of FUN true. When CLK comes true, it makes the clock input of FUN true; when it goes false, it clocks and sets FUN.

FUN is connected to the clock input of FU1. When CLK comes true, it makes the clock input of FU1 true. When CLK goes false, FU1 is clocked and resets. The controller has now advanced to state 03F01U on a space record error.

C/FU1 = FUN CLK NFSCC 03F + . . .

4-80 Unusual End

If FUN (unusual end) is true and FSC (service connect) is false when the controller enters state 03F03U-02U, it immediately advances to state 03F01U. FUN is connected to the clock inputs of FU1 and FIN.

C/FU1 = FUN CLK NFSCC 03F + . . .

C/FIN = FU1 FUN 03F + . . .

When CLK comes true, the clock input of FU1 is made true, and when it goes false, it clocks and resets FU1. When FU1 goes false, it clocks and sets FIN. The controller now advances to state 03F01U on an unusual end condition.

If, when the controller advances to state 03F03U-02U, DCA and TTSH, AIOR, and AUTO are false and FUN sets.

S/FUN = 03F BAND29

C/FUN = 03F BAND29 CLK

BAND29 = NDCA NTTSH NAIOR NAUT

When CLK comes true, it makes the set input of FUN true; when it goes false, it clocks and sets FUN.

FUN is connected to the clock input of FU1. When it goes false, it clocks and resets FU1, advancing the controller to state 03F01U.

4-81 Write Phase

If the controller contains a write order when it enters state 03F03U-02U, WRT is true, and the controller enters the write phase. See figure 4-62 for a flow diagram of the 03F03U-02U write phase.

WRT = F01 F03 F03 F05 + ...

Paragraphs 4-87 through 4-106 present a detailed explanation to the write data flow through the Model 7320 magnetic tape controller.

4-82 WRITE CIRCUITRY AND DATA PATH. The magnetic tape controller receives its write input data from the IOP over nine individual data lines. Each line transfers a data bit, made up of eight data bits and a parity bit, that is used to form a complete data byte. The nine bits which make up one complete character are submitted in parallel to the controller from the IOP.

The output of the IOP is connected to an eight-byte buffer memory in the magnetic tape controller. This buffer memory stores up to eight bytes of information and transfers data through the write register and the associated circuitry to the magnetic tape station. The magnetic tape station contains the circuitry required to transfer this data to the magnetic tape. See figure 4-63 for a block diagram of the write data flow in the magnetic tape controller.

Since many of the registers and the counters used in the write operation are the same that are used for the read operation, their operation has already been described in the paragraphs covering the read circuitry. Therefore, their basic operation is not described here in great detail.

4-83 WRITE DESKEW COUNTER. The write deskew counter is a five-bit binary ring counter. It divides the output of the master clock signal to provide reference timing signals required for the write operation.

The counter receives its input signal directly from the clock circuit. This input (CLOCK) is connected to the clock input of the first stage in the counter. The first stage is a simple bi-stable flip-flop that divides the CLOCK input signal in half. The other stages in the register operate in a normal binary manner with their reset outputs connected back to their set inputs and qualifying terms from the other stages in the counter connected to their clock inputs. The reset inputs of stages 2 through 5 are wired true, and all stages have their dc-reset inputs grounded.

The outputs of the counter are used to initiate the transfer of data from the IOP to the buffer memory in the magnetic tape controller and to control the flip-flops in the write data register in the magnetic tape station.

4-84 Write Deskew Counter Operation. The input signal for the first stage in the counter (WDC1C) is the CLOCK signal.

S/WDC1C = NWDC1C

R/WDC1C = WDC1C

C/WDC1C = CLOCK

WDC1C is connected as a bi-stable flip-flop and changes states each time that CLOCK goes false.

WDC1C is connected to an AND gate with CLOCK. The output of this gate is connected to the clock input of the second stage in the counter, WDC2C.

S/WDC2C = NWDC2C

C/WDC2C = WDC1C CLOCK

When WDC1C is true and CLOCK goes false, WDC2C is clocked. The frequency of WDC2C is one half of WDC1C.

The operation of the other stages in the counter is similar with each stage dividing the output of its preceding stage in half. The output of the last stage in the counter is (WDC5C).

S/WDC3C = NWDC3C

C/WDC3C = WDC1C WDC2C CLOCK

S/WDC4C = NWDC4C

C/WDC4C = WDC1C WDC2C WDC3C CLOCK

S/WDC5C = NWDC5C

C/WDC5C = WDC1C WDC2C WDC3C WDC4C CLOCK

When the counter has completed its count operation and all stages are reset, the next CLOCK signal starts the count cycle again.

Four outputs from the write deskew counter are used to control the transfer of data from the IOP to the write data register in the tape station. These four outputs are: WDC45C, CWR, WDCxC, and (WLRC CWR).

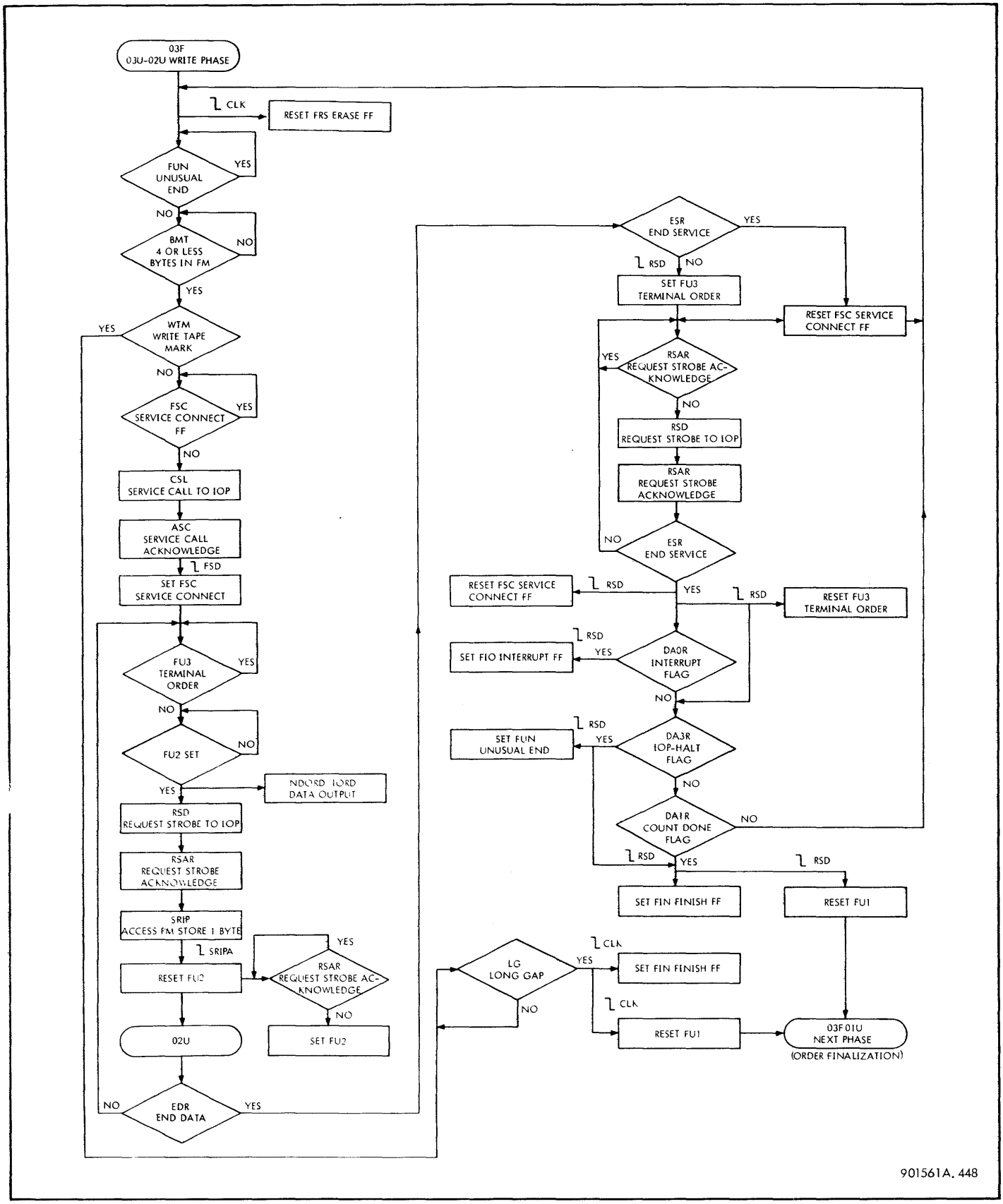


Figure 4-62. 03F03U-02U State (Write), Flow Diagram

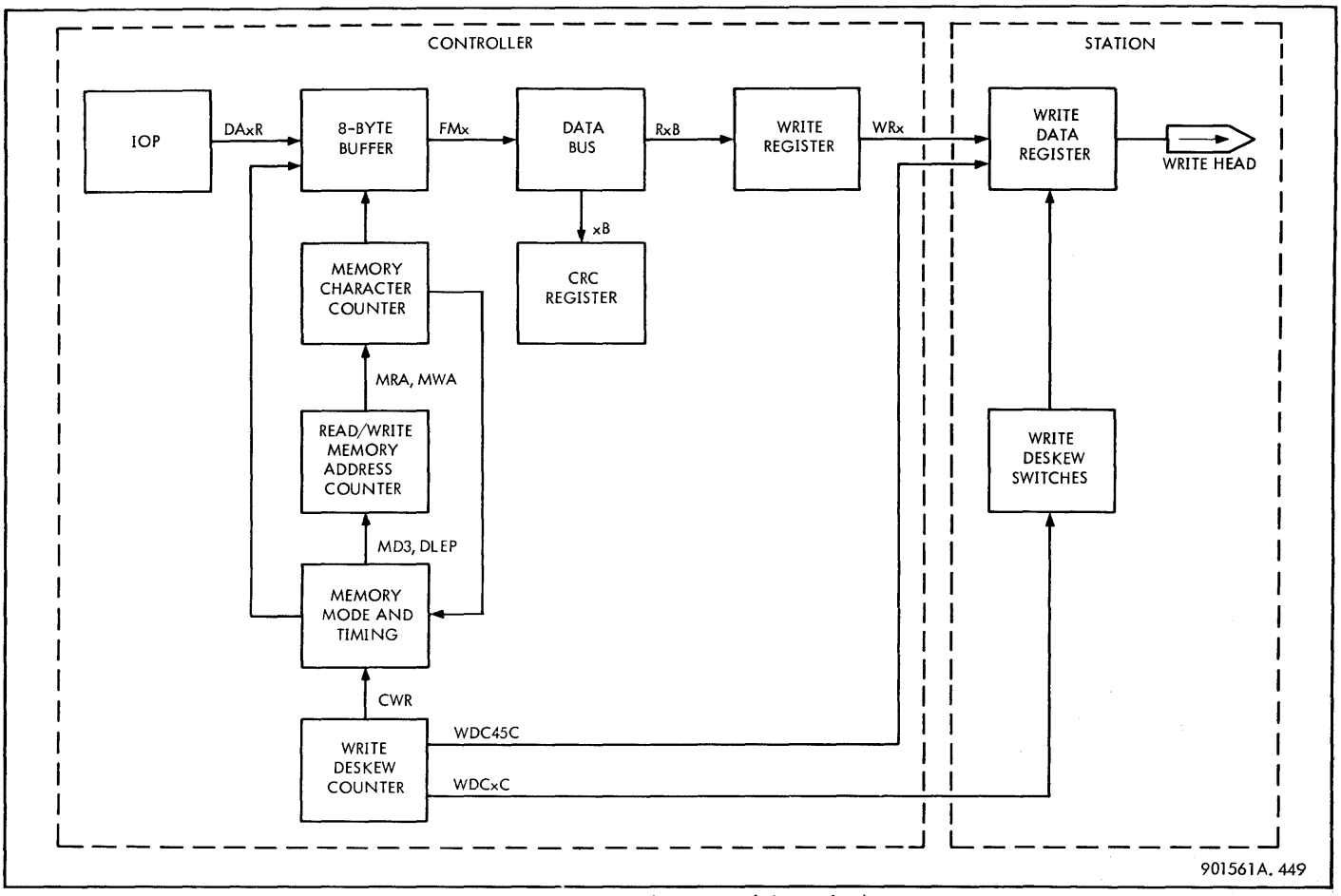


Figure 4-63. Write Data Flow, Simplified Block Diagram

WDC4C and WDC5C are AND-gated to generate signal WDC45C. WDC45C is AND-gated as CLKS with FCTS to form signal WRDE.

WRDE = CLKS FCTS

CLKS = WDC45C

CLKS determines the data recording rate onto tape during write.

WRDE is AND-gated at the reset inputs of the flip-flops in the write data register with NWLRC. WRDE is also AND-gated at the set inputs of the same flip-flops with the reset output of each particular flip-flop.

CWR is true when all stages of the write deskew counter are false.

CWR = NWDC1 NWDC2 NWDC3 NWDC4 NWDC5

CWR is AND-gated with WRITE3 to generate signal WRCW.

WRCW = CWR WRITE3

When CWR comes true, WRCW comes true. WRCW is AND-gated with NMCZ in the memory mode select and timing circuitry. When CWR goes false at the next CLOCK signal, WRCW goes false and initiates a timing sequence that transfers a byte from the buffer memory in the magnetic tape controller to the write data register in the tape station.

WDC1C, WDC2C, and WDC3C are individually connected to buffer amplifiers and to line drivers and are sent to the write deskew switches in the write data register. Each write channel in the station has its own set of write deskew switches. When the output of WDC1C, WDC2C, and WDC3C match the switch settings for a particular channel, a clock signal is generated for the flip-flop in that channel. CWR is AND-gated with WLRC to generate signal (WLRC CWR).

(WLRC CWR) = CWR WLRC + NRES (WLRC CWR)

(WLRC CWR) is connected to a buffer amplifier and line driver and is sent to the write data register where it is inverted to NWLRC. NWLRC is AND-gated at the reset input of the flip-flops in the write data register with WRDE.

(WLRC CWR) is false until WLRC times, and NWLRC is true, allowing the reset inputs of the flip-flops to be true. In

this manner, each one transferred from the WR complements the corresponding flip-flop in the WDR. When (WLRC CWR) comes true, it is time to write the LRC character. NWLRC goes false at that time, and the next clock signal sets any flip-flop in the write data register that is reset. This causes all of the flip-flops in the write data register to be in the set condition (the erase mode) following the LRC character.

4-85 MEMORY MODE SELECT AND TIMING (WRITE). See paragraphs 4-52 through 4-58 in the read circuit analysis for a basic explanation of the operation of the memory mode select and timing circuits.

4-86 Data Transfer From IOP to Eight-Byte Buffer. When a write operation is started for the first time, there is no data stored in the eight-byte buffer. The memory character counter operation (see paragraphs 4-62 through 4-64) is reset and indicates zero count. MCZ (memory character counter at zero) is true, and NMCZ is false. NMCC is the output of the third stage in the counter and is true until there are five bytes stored in memory. BMT is true in a write operation when there are four or fewer bytes stored in memory.

BMT = NMCC WRITE1 + MCC READ1

BMT is AND-gated with NFUN to generate signal BAND30. BAND30 is AND-gated with BAND22 to generate signal CSLX.

CSLX = BAND30 BAND22 + . . .

BAND22 = WRITE NWTM FU1

NFUN = Unusual end flip-flop output

CSLX is inverted to NCSLX, and NCSLX is then inverted to CSL. CSL is a service call request signal. CSL is routed through the subcontroller where its signal designation is changed to SC. SC is the service call request signal that is sent to the IOP. The IOP answers the service call request and returns signals ASC (acknowledge service call), FS (function strobe), and AVI (available input), a priority signal. These signal designations are changed in the subcontroller from ASC to ASCR, FS to FSR, and AVI to AVIR. These signals are then used to generate ASCB.

ASCB = ASCR FSR AVIR ASCM

ASCM = LSH + LSL NHPSL

LSH = CSH CSL NFSC INC NASCR + LSH NRSTR ASCR INI NFSC

ASCB is the set input signal for the service connect flip-flop FSC and causes it to be true.

S/FSC = ASCB

R/FSC = ESRC FSC

C/FSC = FSR NFSC + RSD FSC

E/FSC = NINI + RSTR

When FSR goes false, it clocks and sets FSC. (See figure 4-64 for a representation of the signals used in a write, or IOP to eight-byte memory process.) BAND22 (WRITE NWTM FU1) is AND-gated with FU2 and NFU3 and causes RSD (request strobe) to come true.

RSD = BAND22 FU2 NFU3 + . . .

FU3 is the termination flip-flop.

IORD now comes true and indicates to the IOP that the controller is ready for a data transfer.

IORD = BAND22 BAND16 + . . .

BAND16 = NFU3 RSD

When IORD is made true, the IOP starts to put data on the DAxR lines at the input of the eight-byte buffer. When the data is on the DAxR lines, the IOP causes RSA to come true. When RSA comes true, it causes RSAR to come true.

RSAR = RSARC NMAN + . . .

RSARC = RSA

NMAN = NMANC (selector toggle switch output)

RSAR is AND-gated with BAND16 and BAND22 and causes SRIP to come true.

SRIP = RSAR BAND16 BAND22

SRIP is AND-gated with SRIPDLY and causes SRCON to come true.

SRCON = SRIP NSRIPDLY

SRIPDLY = MC8 WRITE + . . .

SCRON sets SRA true.

M/SRA = SRCON NDLA2

MD3 is AND-gated with the DAxR lines at the input to the eight-byte buffer (see figure 4-44). These lines now contain the character from the IOP. This character is stored later in the buffer memory in a discrete address that is selected by the write memory address counter.

SRA is the signal that is used to start a pulse down the memory delay lines. SRA causes DLAP to come true (see figure 4-64). The pulse continues down the delay lines, and DLEP comes true. (DLEP and MD3 are used throughout the following sequence of events to clock the buffer memory.)

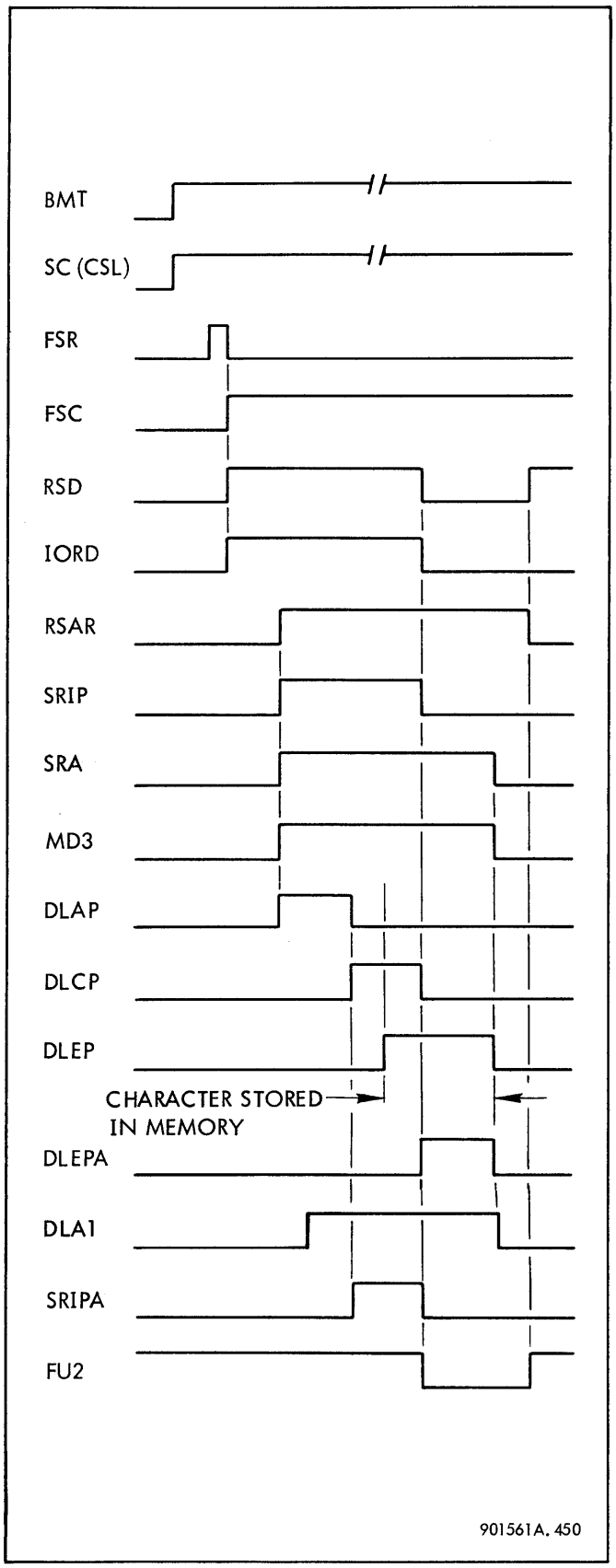


Figure 4-64. Memory Mode Select and Timing (Write), IOP to Buffer Data Transfer, Timing Diagram

DLA1 is AND-gated with MD3 and generates MWA CLOCK (see paragraphs 4-57 through 4-59 and figure 4-43, for the write memory address counter).

MWA CLOCK = DLA1 MD3 + . . .

MWA CLOCK is used to clock the write address counter and the memory character counter. When DLA1 goes false, it causes MWA CLOCK to go false and to clock the flip-flops in both counters. MWA in the write address counter now is true, and MCA and NMCZ in the memory character counter are true.

DLEP is also AND-gated with MD3 at the clock input of the eight-byte buffer.

C/eight-byte buffer = (MD3 + MD2) DEL DLEP

MD3 is AND-gated with the outputs from the write address counter at the word address inputs to the eight-byte buffer. When DLEP goes false, the write address counter selects the address where the character that is on the DAxR lines is to be stored. The buffer is clocked and accepts the character into memory.

MD3 and DLCP are AND-gated and generate SRIPA. SRIPA is used to clock flip-flop FU2.

SRIPA = DLCP MD3

C/FU2 = BAND22 SRIPA + . . .

M/FU2 = BAND25 + . . .

BAND25 = (WRITE NWTM FU1) NRSAR

FU2 is dc-set by BAND25. It is reset when SRIPA goes false; FU2 is used to prevent pulse crowding in the memory delay line. When FU2 goes false, it causes RSD to go false. When RSD goes false, it causes SRIP and IORD to go false. Another memory cycle cannot start until FU2 goes true and DLA1 goes false.

The IOP now causes RSA to go false, and RSA causes RSAR to go false. When RSAR goes false, NRSAR goes true. NRSAR is AND-gated with BAND22 and dc-sets FU2. When FU2 sets, it causes RSD to come true and the process of transferring one character from the IOP to the eight-byte buffer memory is ready to be repeated.

IORD comes true again and indicates to the IOP that the controller is ready for a data transfer. The IOP puts data on the DAxR lines and causes RSA to come true. When RSA comes true, it causes RSAR to come true. RSAR causes SRIP to come true, and the process of transferring a character from the IOP to the buffer memory is repeated as previously explained. See figure 4-65 for a representation of the timing sequence for a four-byte data transfer from the IOP to the eight-byte buffer.

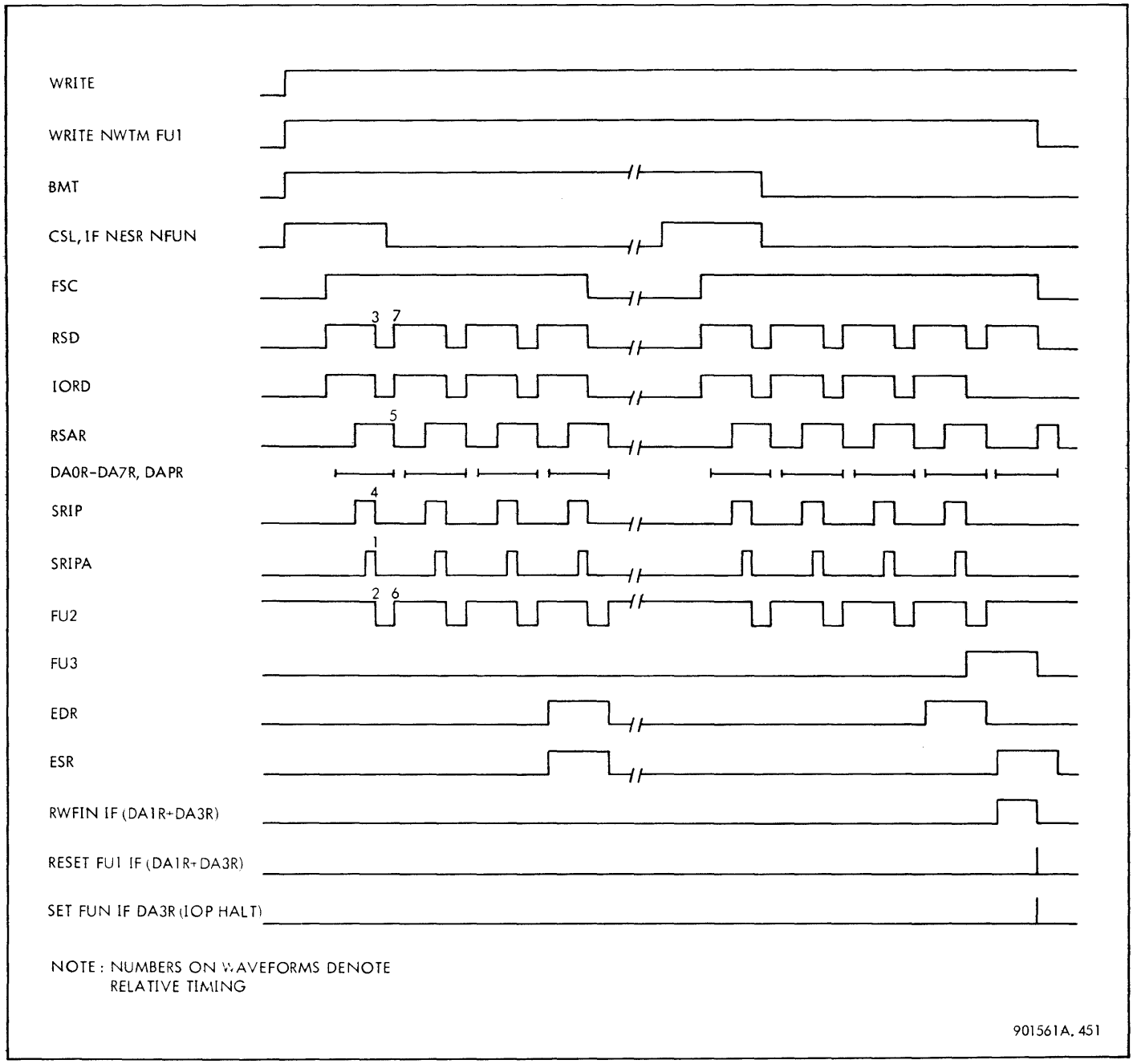


Figure 4-65. Memory Mode Select and Timing (Write), IOP to Buffer Four Byte Transfer, Timing Diagram

A total of four bytes are transferred from the IOP to the memory in one period. After the third byte has been transferred to memory, the IOP raises ED (end data); if no terminal order has been given, it also raises ES (end service). The signal designations are changed from ED and ES to EDRC and ESRC, respectively, in the subcontroller. When EDRC and ESRC come true, they cause EDR and ESR to come true.

EDR = NMAN EDRC + . . .

ESR = NMAN ESRC + . . .

When both EDR and ESR are true simultaneously, the last byte is transferred, and then the service connect flip-flop FSC is reset. ESR is connected to the reset input of FSC, and RSD is connected to its clock input.

R/FSC = ESR FSC

C/FSC = RSD FSC + . . .

When RSD goes false, it clocks and resets FSC which disconnects service.

When the buffer memory has stored four bytes, either data can be read from the memory or four more bytes can be stored in memory, depending on the control timing. If CWR does not come true immediately following the last service cycle, another service request is made by the controller and four more bytes are stored in memory.

It must be remembered that data is stored in memory at a very rapid rate, but is read from memory at a slow rate. It is possible to store four bytes in memory between each byte that is read out.

Assume that after the first transfer of four bytes from the IOP to the buffer memory, CWR does not immediately come true. BMT is still true, since the buffer contains only four bytes, and CSL (service call request) is raised by the controller. The timing sequence repeats, as previously explained, and four more bytes are transferred from the IOP to the buffer memory (see figure 4-64).

After the third byte has been sent to the buffer memory, the IOP raises EDR. If the next byte is the last one for this record, the IOP does not raise ESR until the last byte has been transferred. The controller senses the absence of ESR and sets FU3.

S/FU3 = EDR NESR FSCC + ...

C/FU3 = FSCC RSD + ...

When RSD goes false following the fourth byte, FU3 is clocked and set. The controller now requests a terminal order by raising RSD.

RSD = FU3 BAND22 + ...

The IOP senses RSD, puts the terminal order on the DA0R through DA7R lines, and raises ESR. After a delay, the IOP raises RSAR. When RSAR comes true, it causes RSD to go false (NRSD, ESR and RSAR are AND-gated, the output of this gate is inverted, and is connected to RSD).

When RSD goes false, it clocks and resets FU3 and FSC. The IOP senses that RSD is false and causes RSAR to go false.

When ESR comes true and the terminal order is given, the set input of flip-flop FIN comes true.

S/FIN = SRWFIN

SRWFIN = ESR (DA1R + DA3R) + ...

C/FIN = CRWFIN

CRWFIN = RWFIN

E/FIN = RES1

RWFIN = BAND32 RSD

BAND32 = FU3 BAND22

When RSD goes false, it clocks the FIN flip-flop and sets it. FIN resets at the start of the next order.

The buffer memory now contains eight bytes. After service has been disconnected this time, memory is full and it is impossible to store any more data in the memory. BMT is false, and CSL cannot be raised. CWR now comes true and initiates a data transfer of four bytes from memory.

Assume that, following the first transfer of four bytes from the IOP to memory, CWR comes true immediately following the service cycle. CWR initiates a transfer of one byte from memory, but then BMT initiates a transfer of four more bytes to memory, leaving seven bytes in memory. BMT now is false, and CWR causes three bytes to be read from memory, leaving four in memory, BMT now comes true and causes four more bytes to be stored in memory, filling it. This process of alternately storing and reading data is repeated for the entire record, depending upon the control timing. During each memory cycle, DLA1 updates the address counters and the character counter.

4-87 Data Transfer From Eight-Byte Buffer to Write Register. Assume that the buffer memory has stored four bytes and that CWR comes true immediately. CWR is AND-gated with WRITE3 and generates signal WRCW which is AND-gated with NMCZ. (See figure 4-39.) Signal WRCW clocks flip-flop SRRWA.

C/SRRWA = WRCW NMCZ + ...

SRRWA sets and causes SRA and NPRI to come true.

M/SRA = NDLA2

M/NPRI = SRRWA NSRCON NDLA2

When SRA comes true, a pulse is started down the memory delay line. See figure 4-66 for a timing diagram of this operation.

The pulse continues down the delay line, and DLEP comes true. (DLEP and MD4 are used extensively in the following sequence of events to transfer data from the eight-byte buffer to the write register.)

MD4 is AND-gated with the outputs from the read address counter (MRA, MRB, MRC) at the word address inputs to the eight-byte buffer.

WORD ADDRESS EXP 0 = (MD4 + MD1) MRA + (MD2 + MD3) MWA

WORD ADDRESS EXP 1 = (MD4 + MD1) MRB + (MD2 + MD3) MWB

WORD ADDRESS EXP 2 = (MD4 + MD1) MRC + (MD2 + MD3) MWC

The output from these gates (WORD ADDRESS EXP x) is connected to the address input of the eight-byte buffer and selects the character that appears on the FMx (output) lines of the memory.

Note

It is not necessary to clock the memory in an operation where data is read from the memory; all that is required is that the address of the data be selected.

DLA1 is AND-gated with MD4 and generates signal MRA CLOCK in the read address counter.

MRA CLOCK = DLA1 MD4 + ...

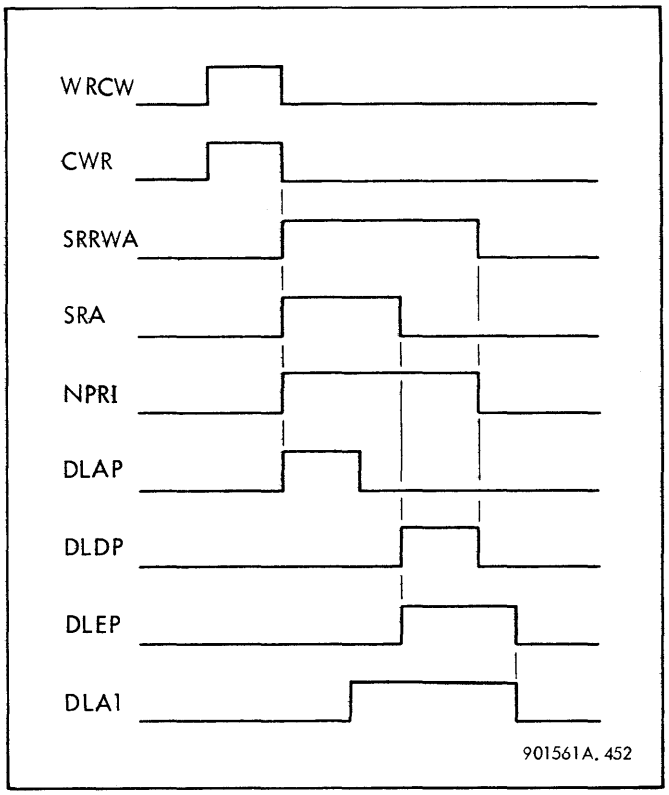


Figure 4-66. Memory Mode Select and Timing (Write), Buffer to Write Register, Timing Diagram

MRA CLOCK is used to clock the read address counter and the memory address counter. When MD4 goes false, it causes MRA CLOCK to go false and to clock the flip-flops in both counters. The read address counter now advances by one state and selects the address of the next character that will be read from memory. The memory character counter counts down by one state, indicating that one character has been removed from memory. (See paragraphs 4-57 through 4-64 for an explanation of the operation of the read and the write memory address counters and of the memory character counter.)

When DLDP goes false, it resets SRA and SRRWA.

The buffer memory now contains three bytes. BMT is true and initiates the timing sequence that causes four more bytes to be stored in memory for a total of seven. CWR comes true and initiates another transfer of data to the write register. Three bytes are read from memory this time; then BMT comes true. BMT initiates the timing sequence that stores four more bytes in memory. This process is typical of the operation used for the entire write process.

Each FMx line contains one bit of the character that is read from memory and is connected to an input gate of a data bus. It is AND-gated at the data bus with NWCRC, (MD1 + MD4) and NWAIT. The output of each gate is RxB.

$$RxB = FMx\ RW1 + \ldots$$

$$RW1 = NWCRC\ (MD1 + MD4)\ NWAIT$$

NWAIT is the reset output of the WAIT flip-flop and is true during a write process after the first byte has been stored from the IOP to the FAM. RxB is connected to the input gates in the write register and is also AND-gated with WRITE1 to generate signal xB. Signal xB is connected to the input gates of the CRC register.

4-88 Memory Timing and Data Transfer Using the Selector IOP (Write). The method of data transfer between the selector IOP and the eight-byte buffer is as follows: BMT is true when there are four or fewer bytes in the buffer memory, and BMT initiates service between the IOP and buffer memory. When SRIP comes true, SRCON comes true, and a pulse is started down the memory delay line. MD3 is made true, and another byte is stored in memory from the IOP.

This process continues until there are eight bytes stored in memory. At this time, MC8 comes true. MC8 is AND-gated with WRITE2, and makes SRIPDLY true. SRIPDLY inhibits SCRON from making further accesses to the memory.

SRRWA now is clocked by WRCW and NMCZ and sets. It causes MD4 to come true and a pulse to be started down the memory delay line. A byte now is transferred from memory to the tape. This process results in MC8 going false, allowing the inhibited IOP access to continue. After the IOP access is completed, MC8 again goes true to inhibit SRCON. This entire operation is repeated until the program is terminated.

4-89 EIGHT-BYTE BUFFER MEMORY OPERATION (WRITE), DATA TRANSFER IN. Refer to paragraphs 4-65 through 4-67 for a basic explanation of the eight-byte buffer memory operation. A character is presented to the eight-byte buffer from the IOP on the nine DAxR lines (see figure 4-44). The DAxR lines are AND-gated with timing and control signal MD3. When MD3 comes true, the character is impressed on the input lines of the memory. The particular address of this character is established by WORD ADDRESS EXP lines. MD3 is AND-gated with signals from the write address counter to establish the WORD ADDRESS EXP outputs.

The clock signal for the eight-byte buffer is derived from the memory mode and the select timing circuitry.

$$C/\text{eight-byte buffer} = (MD2 + MD3)\ DLEP$$

The clock signal is made true when DLEP comes true, and when DLEP goes false, the character that is at that time on the DAxR lines is clocked into memory at the address established by the WORD ADDRESS EXP lines.

4-90 EIGHT-BYTE BUFFER OPERATION (WRITE), DATA TRANSFER OUT. Control signals are generated in the memory mode select and timing circuits that cause the eight-byte buffer to store and to read out data alternately. When the write operation is initially started, the buffer stores four bytes. Depending on the timing and control signals, the

buffer either stores four more bytes or has one byte read out and then stores four more bytes. MD4 is one of the control signals. MD4 is AND-gated with MRA, MRB, and MRC from the read address counter to establish the WORD ADDRESS EXP outputs at the address inputs to the eight-byte buffer. The WORD ADDRESS EXP lines establish the address of the bytes that are read from memory.

The FMx lines of the buffer contain the byte that is selected by the address lines. It is not necessary to clock the buffer to read data from the FMx lines. The FMx lines are connected to the data bus and are connected from the data bus to the write register and to the other registers. (See figure 4-45 for the input/output configuration.) The read address counter now selects the condition of MRA, MRB, and MRC, and the address of the next byte to be read from memory is determined.

4-91 WRITE REGISTER OPERATION, WRITE. Refer to paragraph 4-68 for a basic explanation of the write register operation. During a write process, the input character is presented to the write register on the RxB lines from the eight-byte buffer and the data bus (see figure 4-46). The RxB lines are AND-gated with WRITE SEL which is true. The output of this gate is inverted to NWIx.

WRITE SEL = NREAD SEL

NWIx = NWRITE SEL + NRxB + ...

NWIx is AND-gated with NCCRC LRC CL which is true and with the set output of that particular stage in the register. NWIx is inverted to WIx and is AND-gated with NCCRC LRC CL and with the reset output of that particular stage in the register. The outputs of these two gates are gated in an exclusive OR configuration with (CCRC NCRCC) WRx-1 which is false. The output of this gate is connected to the set input of that particular stage in the register.

NCCRC LRC CL = NCCRC NCCRC/CRCC CLOCK

The clock signal, WRITE REG CLOCK, comes true when DLEP and MD4 come true.

WRITE REG CLOCK = DLEP MD4 + ...

When MD4 goes false, WRITE REG CLOCK goes false and clocks each of the flip-flops in the register. The register then contains the character that was on the RxB input lines.

Each flip-flop in the write register is dc-reset by WR RESET when WRCW comes true.

WR RESET = WRCW (NMCZ + FIN)

Thus, each time that a one bit appears on an RxB input line, that particular flip-flop in the write register sets at the next clock time, and resets when WRCW comes true. The output lines of the write register (WRx) are connected to buffer amplifiers and cable drivers, and are routed to the write data register in the station as WRxCD. WRxCD is gated with the write deskew switches and the 2-MHz clock signal, and is connected to the clock input of the flip-flops in the write data register.

4-92 WRITE REGISTER OPERATION, LRC CHARACTER GENERATION. The write register is also used to generate the LRC character. WLRC is AND-gated with MC1 and generates signal RW3. RW3 is connected to each of the data bus inputs, and WLRC is AND-gated with RW3 at bits P, 0, 1, 2, 4, and 5. The output of each gate is RxB, the data bus output, which is true. RxB is connected to the input gating circuits of each flip-flop in the write register, making each set input true.

4-93 MEMORY CHARACTER COUNTER OPERATION (WRITE), CRC CHARACTER TIMING. Refer to paragraphs 4-62 through 4-64 for a basic explanation of the memory character counter operation. When the last character has been read from the eight-byte buffer, NMCZ resets and MCZ comes true. MCZ causes WCRC CLOCK to come true.

WCRC CLOCK = MCZ NWCRC WRITE2 (FIN + WTM)

WCRC CLOCK causes (CRC/LRC FORCE) to come true.

(CRC/LRC FORCE = WCRC CLOCK + WLRC CLOCK

WCRC CLOCK is connected to the dc-set inputs of MCA and MCB and immediately sets them. MCA and MCB, which are now true, are AND-gated with WCRC CLOCK at the dc-set input of NMCZ and cause it to set. When NMCZ comes true, it causes WCRC CLOCK and (CRC/LRC FORCE) to go false.

The counter now contains a count of four. This count provides for the three blank spaces required after the last character in the record and for the one space for the CRC character to be recorded.

BMT now comes true, and the counter counts down until MC1 comes true (at the fourth count).

BMT = NMCC WRITE1 + ...

MC1 = NMCA NMCB NMCC NMCZ

The CRC character now is recorded in the fourth space following the last character in the record.

4-94 MEMORY CHARACTER COUNTER OPERATION (WRITE), LRC CHARACTER TIMING. Immediately after the CRC character has been recorded, NMCZ resets and MCZ comes true. MCZ causes WLRC CLOCK to come true.

WLRC CLOCK = NCZ NWLRC WCRC

WLRC CLOCK causes (CRC/LRC FORCE) to come true.

(CRC/LRC FORCE) = WLRC CLOCK + WCRC CLOCK

The operation of the counter is now the same as previously explained for the timing of the CRC character. The counter counts three blank spaces after the CRC character has been recorded; the LRC character is recorded on the fourth space.

4-95 MEMORY CHARACTER COUNTER OPERATION (WRITE), END OF FILE TIMING. If the operation to be performed is an end of file character, the timing sequence is as follows: The CRC and LRC character timing is performed as previously explained. NMCZ resets immediately after the TM character has been recorded, and MCZ comes true. MCZ causes WLRC CLOCK to come true, and WLRC CLOCK causes (CRC/LRC FORCE) to come true. MCA, MCB, and NMCZ are dc-set as in the CRC and LRC timing, and WLRC CLOCK is AND-gated with WTM at the dc-set input of MCC causing it to set. The counter now contains a count of eight. This count provides for the seven blank spaces required after the LRC character and for the one space for the tape mark character to be recorded. The counter counts down until MC1 comes true at the eighth count. The LRC is then recorded in the eighth space following the TM character.

4-96 CRC CHARACTER GENERATION, WRITE. Refer to paragraph 4-71 for a basic explanation of the CRC register operation. The CRC register does not function in a read after write operation, since it is being used to generate the CRC character to be written on tape.

In a write operation, the input characters are presented to the CRC register on the xB lines from the eight-byte buffer and the data bus (see figure 4-50). The xB lines are AND-gated with FWD SEL which is true, and the output of this gate is inverted to NDBx.

NDBx = NFWD SEL + NxB + . . .

NDBx is inverted to DBx, and DBx is AND-gated with NCx and SHIFT. The output of DBx NCx SHIFT is gated with NDBx Cx SHIFT in an exclusive OR configuration. SHIFT is the output of the SHIFT flip-flop.

S/SHIFT = NSHIFT

C/SHIFT = SHIFT CLOCK

SHIFT CLOCK = SHIFT DLEP + DLAP (NWCRC SRRWA NPRI)

E/SHIFT = RES2

The output of the exclusive OR gate is OR-gated with NSHIFT and the set output of the preceding stage in the register (Cx-1).

A unique method is used to shift data through the register. Any time that the ninth stage in the register (C9) is set, C4, C5, C6, and C7 set to the complement of their preceding stage at the next clock time, and C1 sets. The other stages in the register operate as a normal shift register, with the output of C9 connected back to the input of C1. If C9 is not set, however, the register shifts at the next clock time without complementing.

Assume that the first byte in a record has been sent to the write register and to the CRC register. CWR is false and SRRWA and MD4 are true. SHIFT CLOCK comes true when DLAP and MD4 come true and makes the clock input of the SHIFT flip-flop and CRC REG CLOCK true. See figure 4-67 for a timing diagram of the exclusive OR and shift functions.

When DLAP goes false, it causes SHIFT CLOCK and CRC REG CLOCK to go false. The flip-flops in the CRC register are clocked, and SHIFT is set. In this example, this is the first character in a record, and all flip-flops in the register were initially reset. Thus, all their set inputs are held false by SHIFT and Cx-1, and they remain reset at the clock signal.

DLEP now comes true and causes SHIFT CLOCK and CRC REG CLOCK to come true. When DLEP goes false, the flip-flops in the register are clocked, and SHIFT is reset. Since SHIFT was true when the flip-flops were clocked, the input data on the xB lines is gated into the register in an exclusive OR; the register assumes the state of the character.

When the second character arrives, the process is repeated. DLAP and SRRWA cause the SHIFT CLOCK and the CRC REG CLOCK to come true, and, when DLAP goes false, the register flip-flops are clocked and SHIFT is set. This time, however, data is present in the register. This data is shifted by one stage. If C9 was set, stages C4, C5, C6, and C7 set to the complement of their preceding stage, C1 sets, and the other stages in the register shift. DLEP comes true and causes SHIFT CLOCK and CRC REG CLOCK to come true. When DLEP goes false, the flip-flops in the register are clocked, and SHIFT is reset. The data that the flip-flops contain before they are clocked is gated at their set inputs with the new input data in an exclusive OR, and the flip-flops set accordingly when they are clocked. See table 4-10 for a typical representation of input data and the states of the CRC register.

This process is repeated until the last byte in the record is stored in the CRC register. The CRC character would be 111011110, after complementing all bits except those in positions 4 and 6. The LRC character recorded on tape would be 100100001.

The character parity of the CRC character is even if the number of data bytes in the record is odd and even if the number of data bytes in the record is odd. Note that the record shown in table 4-10 contains 16 data bytes, and that the parity of the CRC character in the WR register is odd.

Table 4-10. CRC Generation During a Write Operation

W-REGISTER									CRC REGISTER									FUNCTION IN CRC		WDR, WRITE DATA REGISTER								
P	0	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	SH	⊕	P	0	1	2	3	4	5	6	7
0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	X		1	1	1	1	1	1	1	1	1
									0	0	1	0	1	0	0	0	1		X	1	1	0	1	0	1	1	1	0
1	1	1	1	1	1	0	0	1	1	0	0	0	1	0	1	0	0	X										
									0	1	1	1	0	1	1	0	1		X	0	0	1	0	1	0	1	1	1
1	1	0	0	1	1	1	1	1	1	0	1	0	0	1	0	1	0	X										
									0	1	1	0	1	0	1	0	1		X	1	1	1	0	0	1	0	0	0
0	0	0	0	1	0	1	0	1	1	0	1	0	1	1	0			X										
									1	0	1	0	0	0	0	1	1		X	1	1	1	0	1	1	1	0	1
1	1	0	1	0	0	0	1	1	1	1	0	0	1	1	1	0	1	X										
									0	0	0	1	1	1	1	1	0		X	0	0	1	1	1	1	1	1	0
1	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	1	X										
									1	1	1	1	0	1	1	0	0		X	1	1	0	0	0	1	1	0	1
0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	1	1	0	X										
									0	1	1	0	0	1	0	0	0		X	1	1	0	1	1	0	0	1	1
0	0	0	1	0	1	0	1	0	0	0	1	1	0	0	1	0	0	X										
									0	0	1	0	0	1	1	1	0		X	1	1	0	0	1	1	0	0	1
1	0	1	0	0	0	1	1	1	0	0	0	1	0	0	1	1	1	X										
									1	0	1	1	0	0	0	0	0		X	0	1	1	0	1	1	1	1	0
0	1	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	X										
									0	0	1	0	1	0	1	1	0		X	0	0	0	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	1	0	1	0	1	1	X										
									0	0	1	0	1	0	1	1	1		X	0	0	1	0	0	0	1	0	0
1	0	1	0	1	0	1	0	1	1	0	0	0	1	0	1	1	1	X										
									0	0	1	0	0	0	0	1	0		X	1	0	0	0	1	0	0	0	1
0	1	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	1	X										
									0	1	0	1	0	1	1	1	0		X	1	1	0	0	1	1	1	1	0
1	1	1	0	0	1	1	0	0	0	0	1	0	1	0	1	1	1	X										
									1	1	0	0	1	1	0	1	1		X	0	0	1	0	1	0	0	1	0
0	1	1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	1	X										
									1	0	0	0	0	1	0	0	1		X	0	1	0	1	0	1	0	1	0
1	1	0	1	0	1	0	1	0	1	1	0	1	1	1	0	0	0	X										
									0	0	0	0	1	0	0	1	0		X	1	0	0	0	0	0	0	0	0
CRC CHARACTER									0	0	0	0	0	1	0	0	1	X										
1	1	1	0	1	1	1	1	0	←											0	1	1	0	1	1	1	1	0
FOR LRC CHARACTER																												
1	1	1	1	1	1	1	1	1												1	1	1	1	1	1	1	1	1

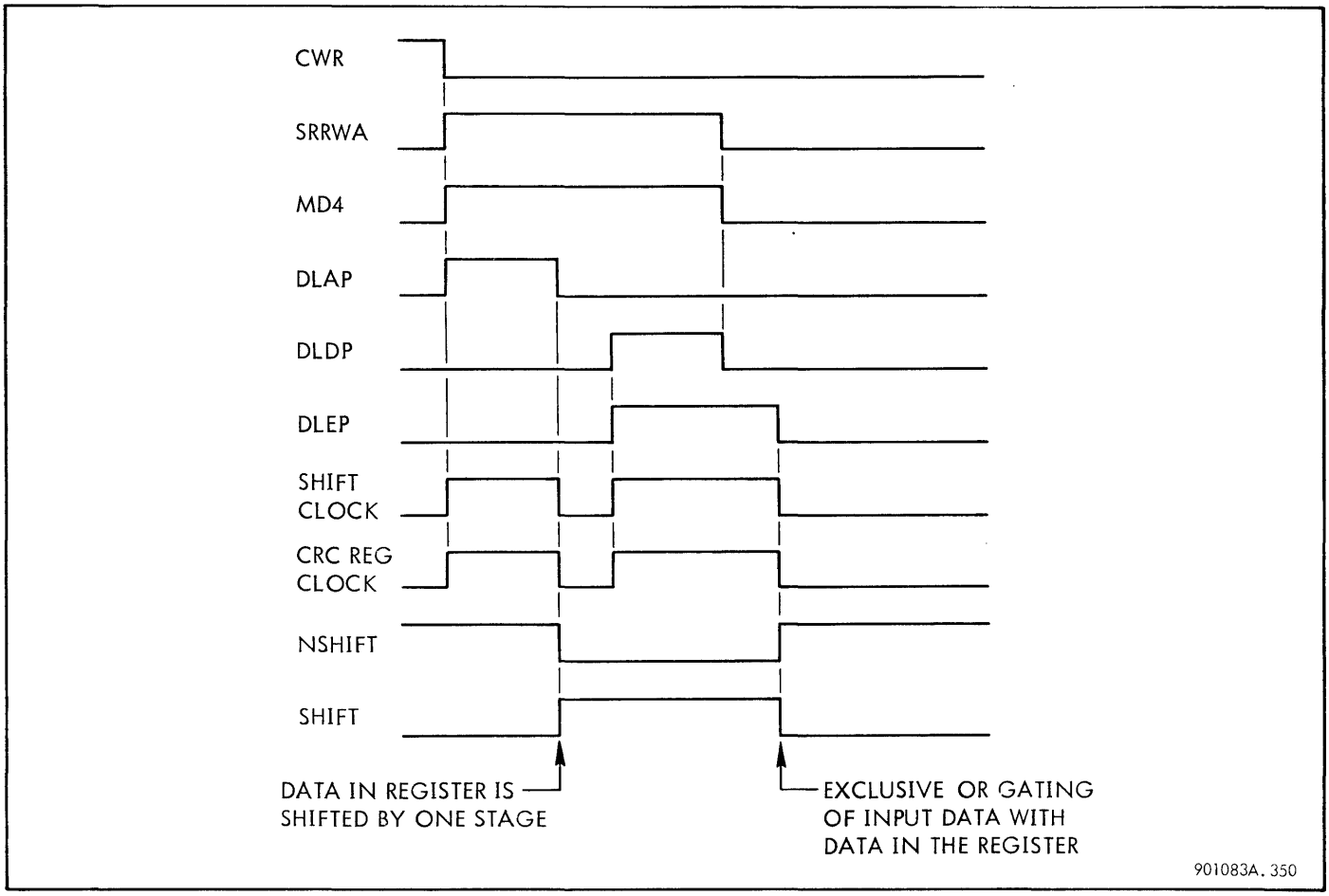


Figure 4-67. CRC Register (Write), Exclusive OR and Shift, Timing Diagram

After the last byte in the record is transferred from the IOP to the buffer, the IOP raises DA1R (count done) and ESR. This makes the set input for flip-flop FIN true.

S/FIN = SRWFIN

SRWFIN = ESR (DA1R + DA3R) + . . .

The FIN flip-flop is clocked by RSD and is set. FIN is AND-gated with WRITE2, MCZ, and NWCRC, and is used to generate signal WCRC CLOCK. See figure 4-68 for a timing diagram of the CRC (write) generation.

WCRC CLOCK = (FIN + . . .) WRITE2 MCZ NWCRC

MCZ (memory character counter at zero count) comes true when the last byte has been read from the buffer memory. MCZ causes WCRC CLOCK to come true. WCRC CLOCK causes CRC REG CLOCK and (CRC/LRC FORCE) to come true.

(CRC/LRC FORCE) = WCRC CLOCK + . . .

(CRC/LRC FORCE) is connected to the dc-set inputs of MCA and MCB in the memory character counter and sets them, giving a count of three in the counter. MCA and MCB are AND-gated with (CRC/LRC FORCE) at the dc-set input of NMCZ, causing it to set. The counter now contains a count of four. When NMCZ sets, MCZ goes false and causes WCRC CLOCK to go false. WCRC CLOCK causes CRC REG CLOCK to go false, and the CRC register is clocked. The data that is in the register is then given one final shift (see table 4-10). WCRC CLOCK also clocks flip-flop WCRC and it sets.

The write deskew counter now passes through zero count. WRCW comes true, causing WR RESET to come true and resetting each of the flip-flops in the write register.

RW RESET = (NMCZ + FIN) WRCW

When WRCW goes false, it initiates a data transfer cycle. The memory character counter counts down by one, and the write register records a blank space on the tape. The CRC register does not change the pattern of the character that is in the register, however, since the buffer memory contains

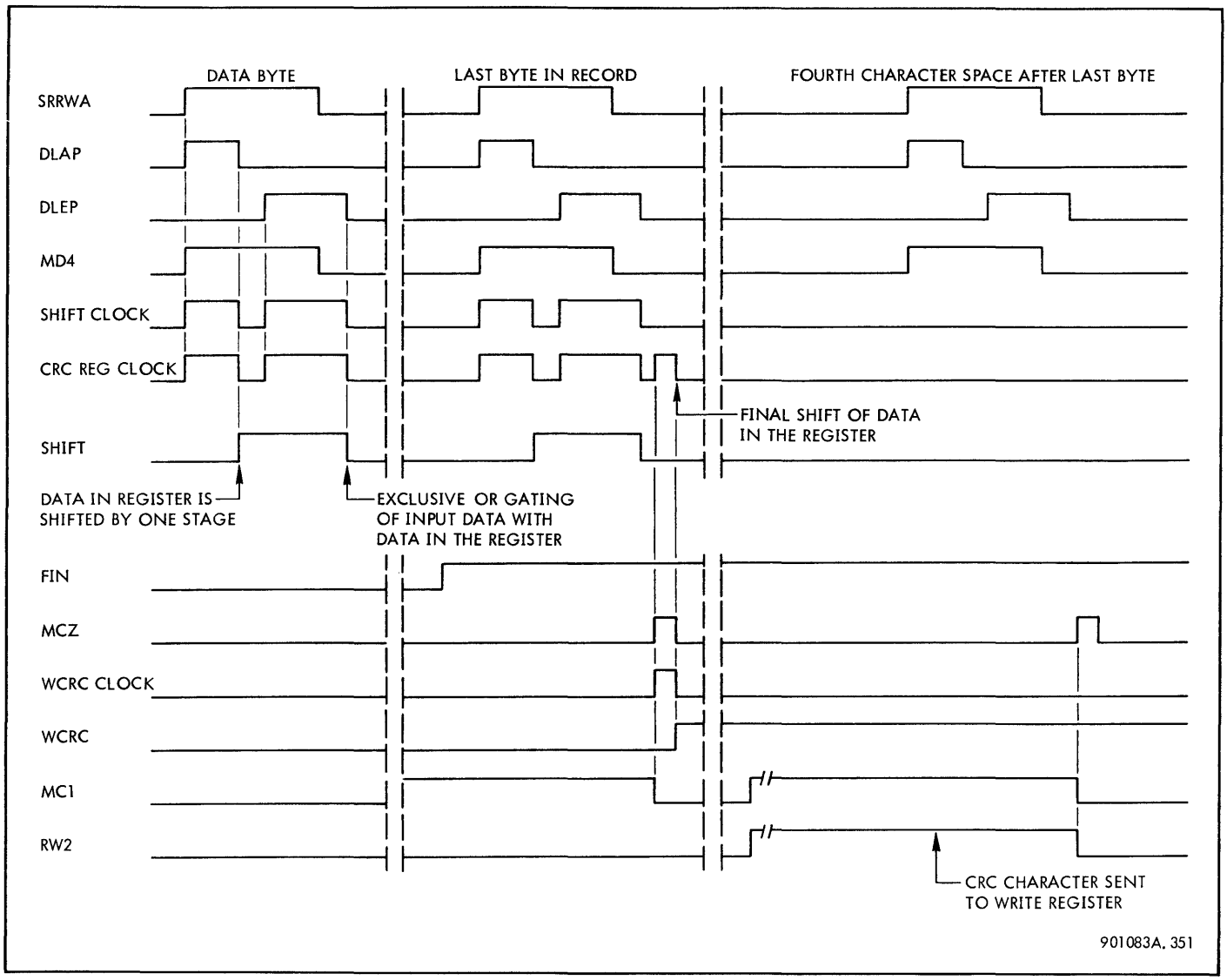


Figure 4-68. CRC Character Generation (Write), Timing Diagram

no data. Also, the SHIFT flip-flop is not clocked since its clock is held false by NWCRC.

SHIFT CLOCK = DLAP (NWCRC NPRI SRRWA)

When the memory character counter reaches a count of one (the fourth character space), MC1 comes true.

MC1 = NMCA NMCB NMCC NMCZ

MC1 is used to generate signal RW2.

RW2 = WCRC NWLRC MC1 NWTM

RW2 is AND-gated in the data bus with the set outputs of flip-flops C4 and C6, and with the reset outputs of flip-flops C1, C2, C3, C5, C7, C8, and C9 from the CRC register. Thus, each output of the CRC register is complemented with the exception of bits four and six.

The outputs of the data bus (RxB) are connected to the write register inputs. The write register is clocked, and the CRC character is loaded into the register. The write register is reset by WR RESET, and the CRC character is sent to the write data register in the station and is recorded in the fourth space after the last byte in the record. The character parity of the CRC character that is recorded on tape is odd if the number of bytes in the record is even; it is even if the number of bytes is odd. This ensures that the character parity of the LRC character that follows is odd.

4-97 LRC CHARACTER GENERATION, WRITE (See figure 4-69). MCZ comes true immediately after the CRC

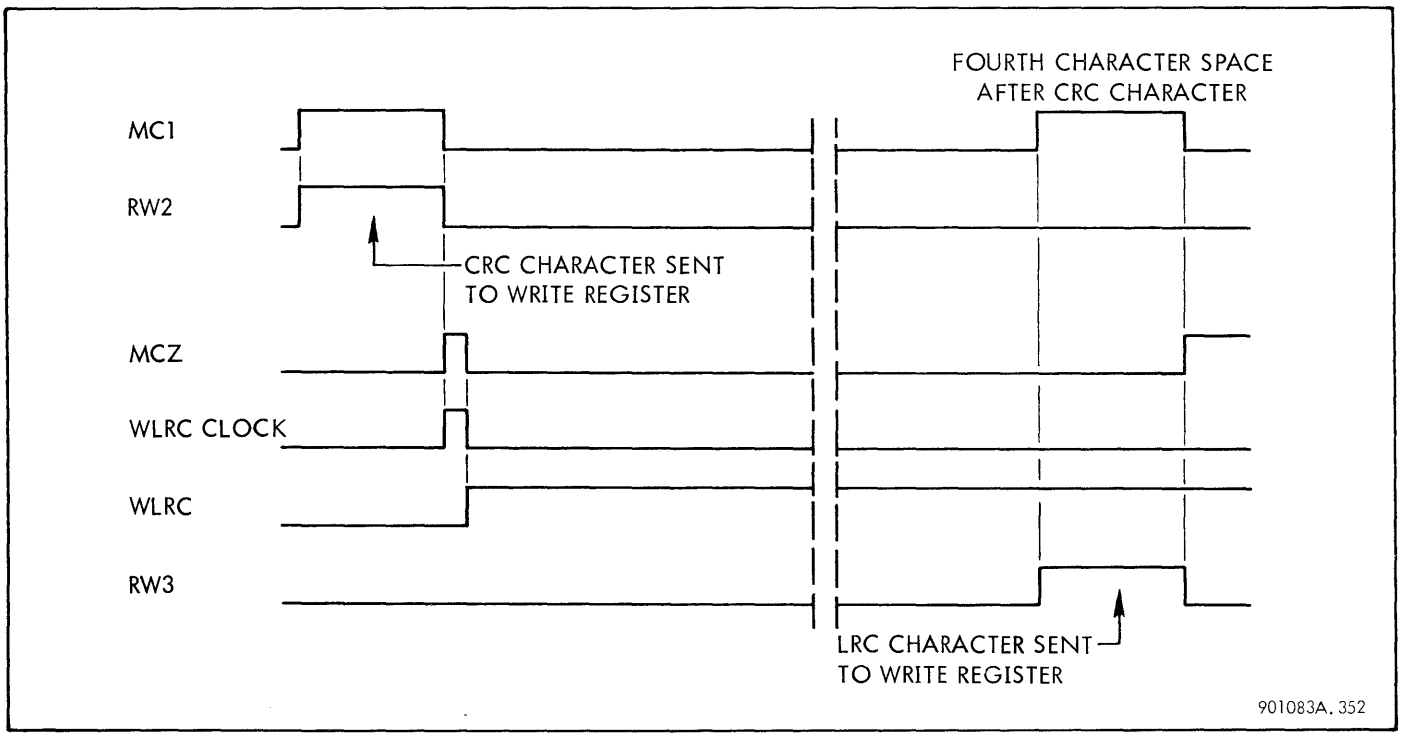


Figure 4-69. LRC Character Generation (Write), Timing Diagram

character is sent to the write register. WLRC CLOCK is made true at that time.

WLRC CLOCK = MCZ WCRC NWLRC

WLRC CLOCK is connected to the clock input of flip-flop WLRC and to the input gate of (CRC/LRC FORCE).

(CRC/LRC FORCE) = WLRC CLOCK + . . .

(CRC/LRC FORCE) is connected to the dc-set inputs of MCA and MCB in the memory character counter. MCA and MCB are dc-set, giving a count of three in the counter. MCA and MCB are AND-gated with (CRC/LRC FORCE) at the dc-set input of NMCZ, causing it to set. The counter now contains a count of four. When NMCZ sets, MCZ goes false, causing WLRC CLOCK to go false and setting WLRC.

The write deskew counter now passes through zero count, and WRCW comes true. WRCW causes RW RESET to come true and to reset each of the flip-flops in the write register. When WRCW goes false, it initiates a data transfer cycle. The memory character counter counts down by one, and the write register records a blank space on the tape.

MC1 comes true when the memory character counter reaches a count of one. MC1 is AND-gated with WLRC and generates signal RW3.

RW3 = MC1 (WLRC + . . .)

RW3 is connected to each of the data bus inputs, and WLRC is AND-gated with RW3 at bits P, 0, 1, 2, 4, and 5. The output of each gate is RxB, the data bus output, which is true. The outputs of the data bus (RxB) are connected to the input gating circuits of each flip-flop in the write register, making each set input true. The write register is clocked, and each flip-flop in the write register is clocked and set. The output of each flip-flop in the write register is connected to a clock input of a flip-flop in the write data register in the station. Figure 4-70 gives a CRC and an LRC timing sequence.

4-98 TAPE MARK RECORD, WRITE. A file of information consists of a group of one or more consecutive records. Adjacent files are separated by a tape mark record that consists of a tape mark character (equivalent to a hexadecimal 13) followed by an LRC character, which is also hexadecimal 13 (000010011). See figure 4-71 for a representation of the tape mark record.

After a file of information is recorded, the tape mark order (WTM) is issued by the IOP by raising the appropriate DAxR lines. WTM is the inverted output of NWTM.

NWTM = NF01345 + N03F + . . .

WTM = F01345 03F + . . .

The tape transport is at rest following the last LRC character in the file of information. When the WTM order is issued, the transport starts and an initial gap of 3-1/2 inches is generated.

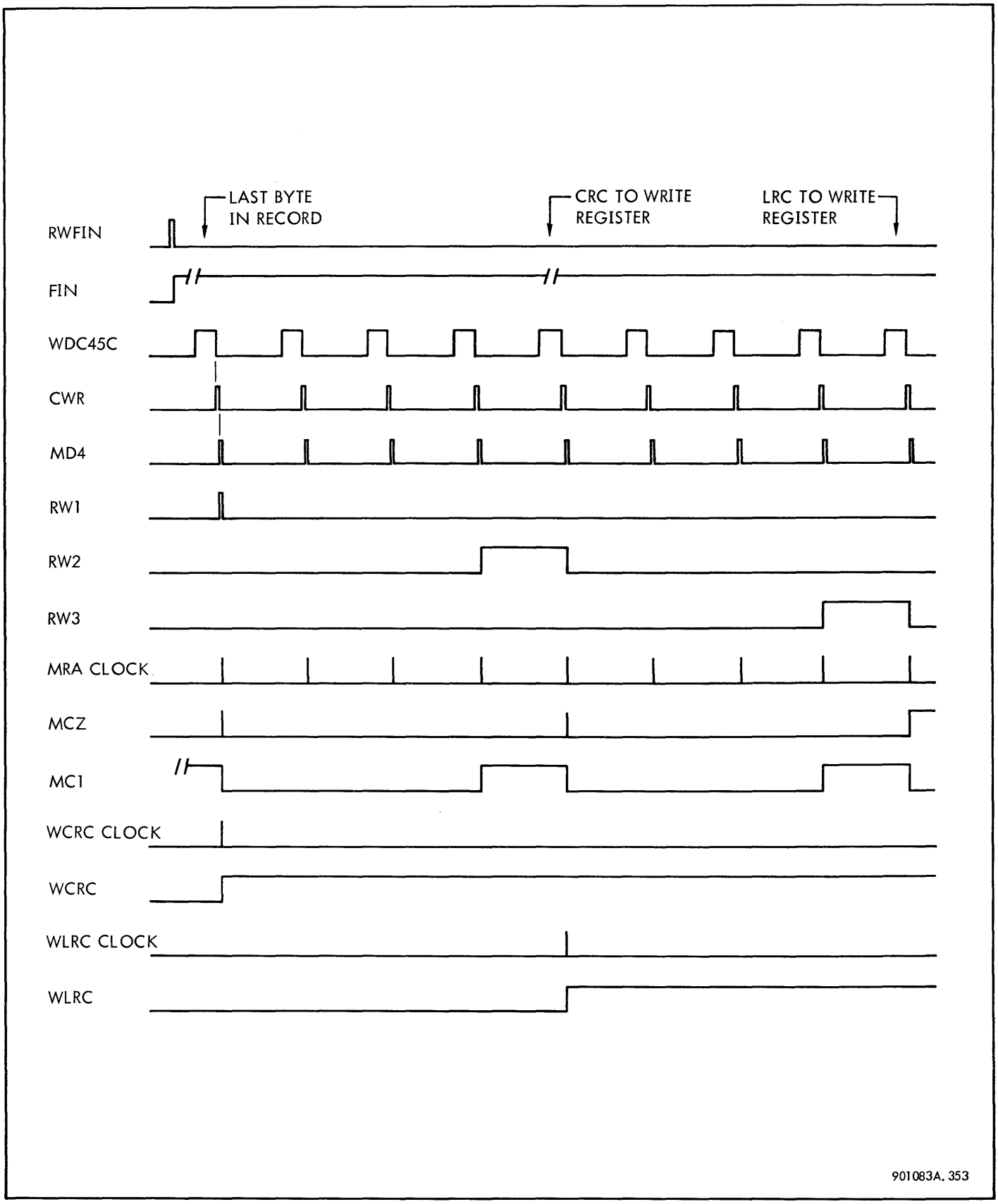


Figure 4-70. CRC and LRC (Write), Timing Diagram

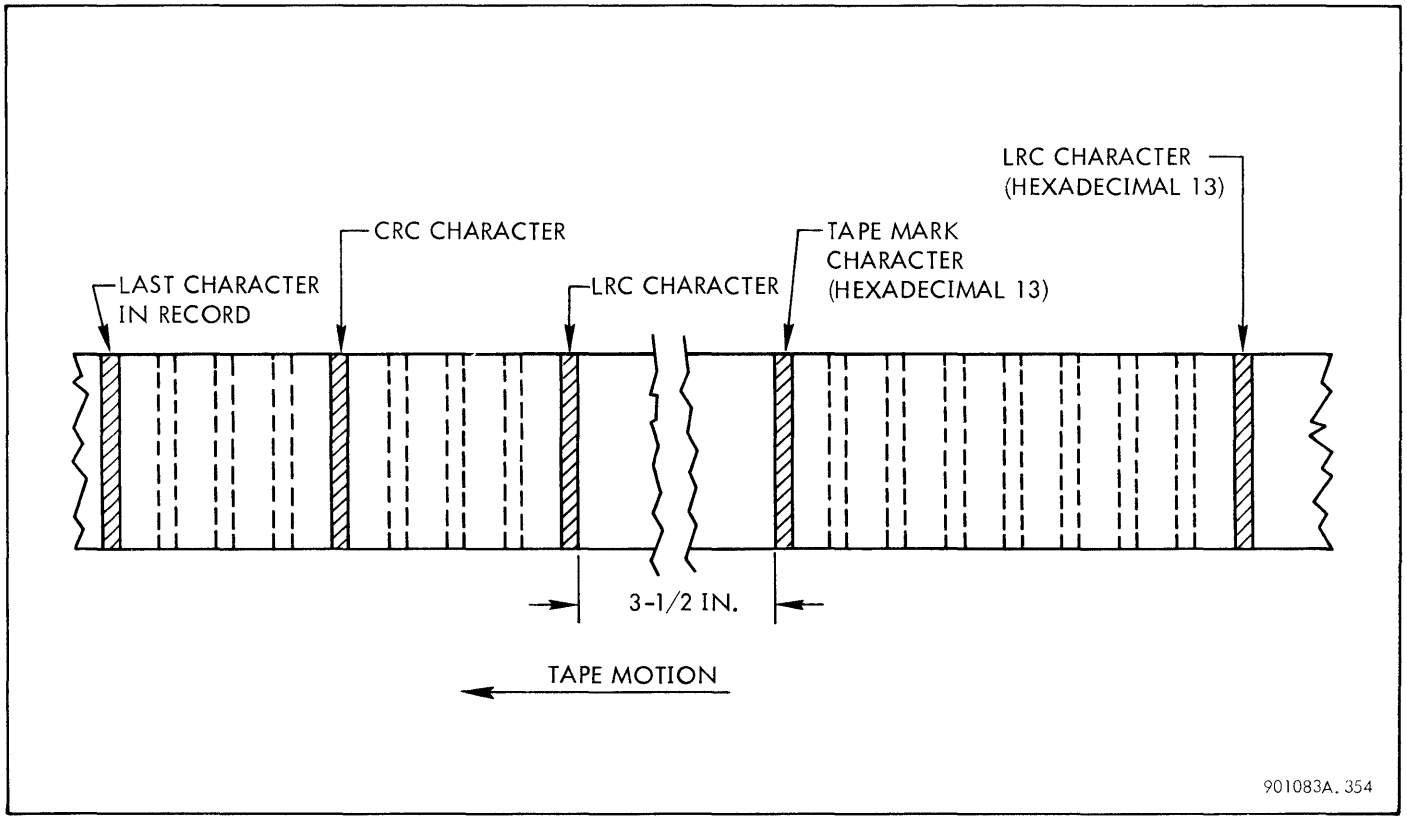


Figure 4-71. Tape Mark Record (Write), Physical Spacing

When WTM is issued, WCRC CLOCK comes true.

$$\text{WCRC CLOCK} = (\text{WTM} + \ldots)\ \text{WRITE2}\ \text{MCZ}\ \overline{\text{NWCRC}}$$

WCRC CLOCK causes (CRC/LRC FORCE) to come true. (CRC/LRC FORCE) is connected to the dc-set inputs of MCA and MCB in the memory character counter and sets them, giving a count of three in the counter. MCA and MCB are AND-gated with (CRC/LRC FORCE) at the dc-set input of NMCZ, causing it to set. The counter now contains a count of four. When NMCZ sets, MCZ goes false, causing WCRC CLOCK to go false and setting WCRC. See figure 4-72 for a timing diagram of this operation.

When the write deskew counter passes through zero count, the memory character counter starts to count down. MC1 comes true when the memory character counter reaches a count of one. MC1 is AND-gated with WCRC and WTM and generates signal RW3.

$$\text{RW3} = \text{MC1}\ (\text{WCRC}\ \text{WTM} + \ldots)$$

RW3 is connected to each of the data bus inputs, and WLRC is AND-gated with RW3 at bits P, 0, 1, 2, 4, and 5. The output of each gate is RxB and, since WLRC is false, the only true outputs from the data bus are bits 3, 6, and 7.

This gives the required hexadecimal 13 (000010011) configuration (P, 0, 1, 2, 3, 4, 5, 6, 7) for the tape mark character.

The outputs of the data bus are connected to the input gating circuits of each flip-flop in the write register, making the set inputs for flip-flops 3, 6, and 7 true. When MD4 goes false, WRITE REG CLOCK goes false and sets flip-flops 3, 6, and 7. Also, MC1 goes false, and MCZ and WLRC CLOCK go true.

$$\text{WLRC CLOCK} = \text{NWLRC}\ \text{WCRC}\ \text{MCZ}$$

WLRC CLOCK causes (CRC/LRC FORCE) to come true, and dc-sets MCA and MCB in the memory character counter. WLRC CLOCK is also AND-gated at the dc-set input of MCC with WTM, and MCC sets. MCA and MCB are AND-gated with (CRC/LRC FORCE) at the dc-set input of NMCZ and cause it to set. When NMCZ sets, the counter contains a count of eight. Also, when NMCZ goes true, MCZ and WLRC CLOCK go false. WLRC CLOCK clocks and sets flip-flop WLRC.

When the write deskew counter passes through zero count, the memory character counter starts to count down. MC1 comes true when the memory character counter reaches a count of one. MC1 is AND-gated with WCRC and generates signal RW3. RW3 is connected to each of the data bus

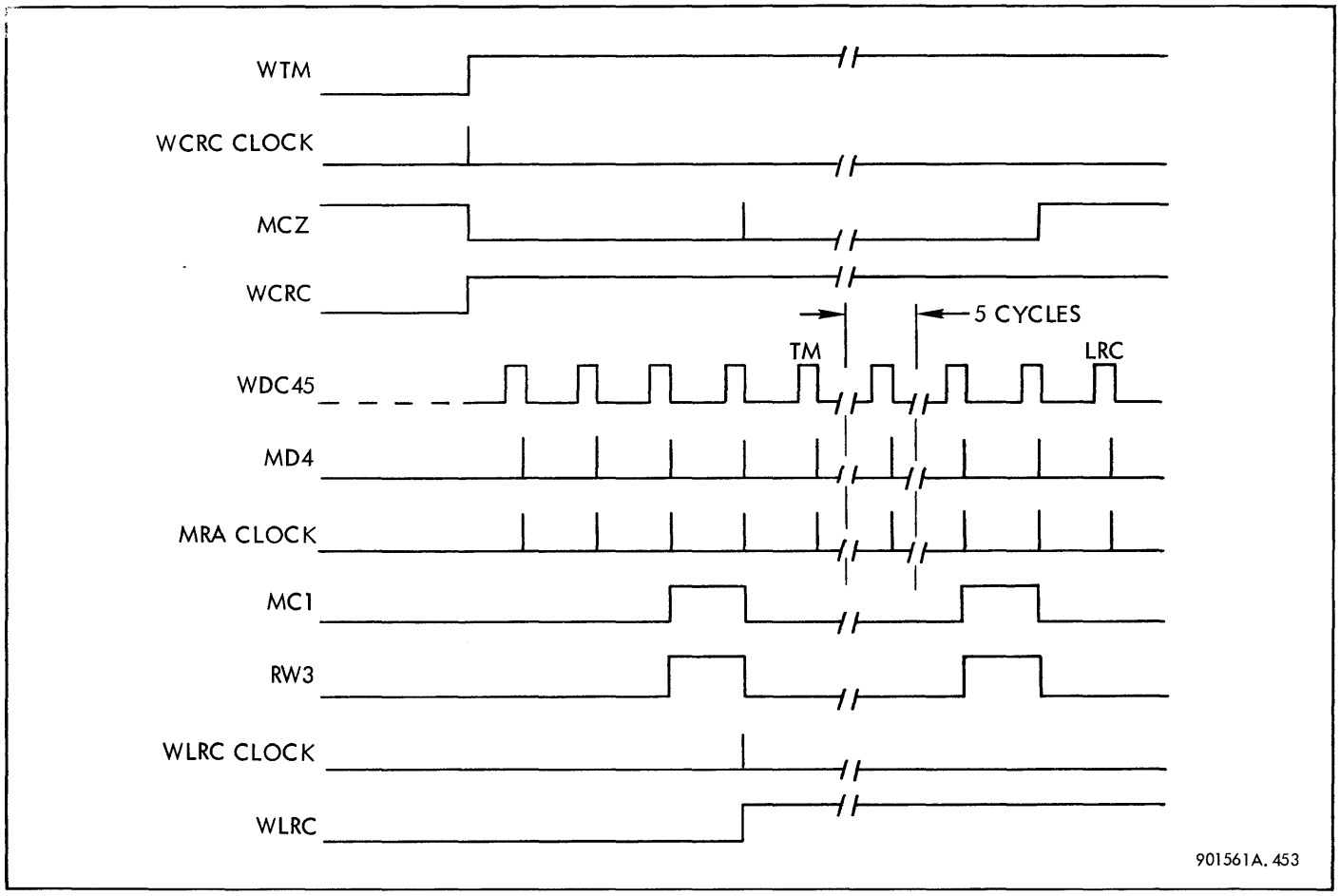


Figure 4-72. Tape Mark (Write), Timing Diagram

inputs, and WLRC is AND-gated with RW3 at bits P, 0, 1, 2, 4, and 5. The output of each gate is RxB (the data bus output) which is true. RxB is connected to the input gating circuits of each flip-flop in the write register, making each set input true. When MD4 goes false, the write register is clocked, and each flip-flop in the register is set.

4-99 WRITE ORDER TERMINATION. RSD goes false when the IOP signals count done by raising DA1R. When RSD goes false, it clocks and sets FIN, and clocks and resets FU1.

C/FIN = FU1 FU3 WRITE RSD + . . .

S/FIN = DA1R + . . .

C/FU1 = FU1 FU3 WRITE RSD

See figure 4-65 for the timing sequence. The controller has now advanced to state 03F01U (order finalization) from the write phase.

If the IOP raises the halt command (DA3R), RSD goes false. When RSD goes false, it clocks and sets FUN and FIN, and clocks and resets FU1.

S/FUN = DA3R ESR NDST FU3 + . . .

C/FUN = FU3 NFUN NDST RSD1 + . . .

S/FIN = DA3R ESR + . . .

The controller now advances to state 03F01U from an unusual end condition.

After the tape mark has been written, LG comes true. LG makes the set input of FIN true. See figure 4-73 for a timing diagram of the tape mark sequence.

S/FIN = LG + . . .

The clock input of FIN is made true by WTM (write tape mark), FU1, and 03F.

C/FIN = WTM FU1 03F

When CLK goes false, FU1 is clocked and resets, and when FU1 goes false, it clocks and sets FIN.

C/FU1 = CLK WTM LG + . . .

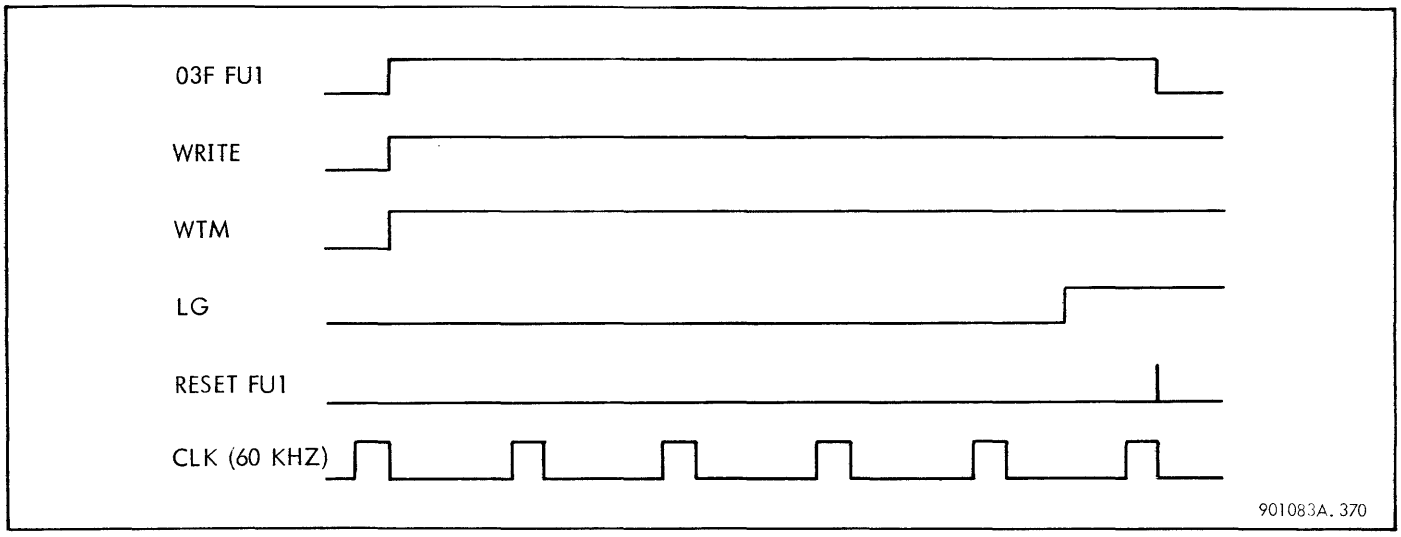


Figure 4-73. 03F03U-02U State (Write Tape Mark), Timing Diagram

The controller now advances to state 03F01U from a write tape mark order.

4-100 Error Circuits WAIT and RATE

In a write operation, the WAIT flip-flop sets before any data is transferred from the IOP and resets when the first byte from the IOP has been stored in memory. WAIT is set so that a transmission error is not indicated in this situation.

S/WAIT = NWAIT

C/WAIT = NWRITE + NMWA WAIT

E/WAIT = RES2

NWRITE is the inverted output of WRITE1. When WRITE1 comes true at the beginning of a write operation, NWRITE goes false and clocks and sets the WAIT flip-flop. When the first byte is stored in the eight-byte buffer, NMWA goes false and clocks and resets WAIT.

NWAIT is AND-gated with MCZ. The output of this gate is AND-gated with WRITE2. The output of this gate develops the set signal for the rate error flip-flop, RATE.

S/RATE = WRITE2 (MCZ NWAIT) + MC8 READ2

C/RATE = WRCW NWCRC + NRATE (READ NRCP) NFIN

WRCW = WRITE3 CWR

M/RATE = RTER

RTER = EAP1 RCP READ2

E/RATE = RESIN

If MCZ comes true during a write operation, the set input of RATE is made true. The next time that CWR goes false, RATE is clocked and set, indicating a transmission error (trying to write data with none in the memory).

When a read operation is in process and MC8 comes true (buffer memory is full), the next time that NRCP goes false, RATE is clocked and sets, indicating a transmission error (trying to store data in memory when it is full).

When NFIN goes false during a read operation (at the end of the data transfer for that record), it prevents the RATE flip-flop from being clocked. This is done so that the final bytes can be transferred without a transmission error indication.

If, during a read operation, the buffer memory does not get serviced in time and EAP1 and RCP come true, the RATE flip-flop is dc-set.

M/RATE = EAP1 RCP READ2

4-101 DATE (Data Error) Circuits

The DATE flip-flop is set whenever an error condition is detected in the data that is either being read from or being recorded on tape.

M/DATE = PE NFCR + CLKE

C/DATE = LRC ERROR

LRC ERROR = (CLOCK CRC/CRCC CLOCK + SG TM) (WRITE + NWRZ) + (NWRITE2 + NERZ)

The set input for the DATE flip-flop is floating. The flip-flop sets with a clock input or with a dc-set condition.

In a write operation, LRC ERROR is used to clock the flip-flop. The clock input normally is held false by CLOCK and ERZ. If an error is detected, ERZ remains false after the LRC character has been read (read after write), and LRC ERROR tries to come true. LRC ERROR is held false, however, by CLOCK. When CLOCK goes true, the clock input of DATE is made true, and when CLOCK goes false, the flip-flop is clocked and sets, indicating a data error.

In a read operation, WRZ normally is false. If an error condition exists after the LRC character has been read, WRZ remains false, and LRC ERROR tries to come true. LRC ERROR is held false, however, by CLOCK. When CLOCK goes true, the clock input of DATE is made true, and when CLOCK goes false, the flip-flop is clocked and sets, indicating a data error.

If a parity error or CRC error occurs during a read operation, the DATE flip-flop is dc-set, indicating a data error.

M/DATE = PE NFCR + CLKE

CLKE = CRCCMPRCLK (CCRC NCRCC) READ2 NWR7 (NCMPR + CMPR WRZ)

If a data byte parity error is detected during a read or a read after write operation, PE NFCR comes true and dc-sets the DATE flip-flop. If a CRC character parity error is detected during a read or a read after write operation, PE NFCR comes true and dc-sets the flip-flop.

4-102 03F01U (TERMINATION)

The 03F01U state is the order finalization (or terminate process) state. In this state the read, write, and space record or space file orders from the preceding state (03F03U-02U) are terminated.

4-103 LG Detection on Read or Read-After-Write Orders

When the controller enters this phase, if LG comes true, END comes true. See figure 4-74 for a flow diagram of the 03F01U state.

END = 03F NFU1 ENDX

ENDX = ENDY + BAND29

ENDY = LG + (F02 BOT)

BAND29 = NDCA NTTSH NAIOR NAUT

END is connected to the clock input of FU2.

C/FU2 = CLK END 03F + . . .

See figure 4-75 for a timing diagram of the terminate process operation.

When CLK comes true, it makes the clock input of FU2 true; when it goes false, it clocks and resets FU2, advancing the controller to state 03F00U (station finalization).

4-104 Beginning of Tape Detection on Reverse Tape Motion

If F02 (reverse order) is true when the controller enters this state and BOT (beginning of tape) comes true, END comes true.

END = 03F NFU1 (F02 BOT + . . .)

END is connected to the clock input of FU2.

C/FU2 = CLK END 03F + . . .

When CLK comes true, it makes the clock input of FU2 true; when it goes false, it clocks and resets FU2, advancing the controller to state 03F00U.

4-105 Unusual End (Station Not in AUTO)

If, when the controller advances to state 03F01U, DCA and TTSH, AIOR and AUTO are false, FUN sets and END comes true.

C/FUN = 03F BAND29 CLK + . . .

BAND29 = NDCA NTTSH NAIOR NAUT

END = 03F NFU1 BAND29

When CLK comes true, it makes the clock input of FUN true, and when it goes false, it clocks and sets FUN.

END is connected to the clock input of FU2.

C/FU2 = CLK END 03F

When CLK comes true, it makes the clock input of FU2 true. When CLK goes false, it clocks and resets FU2, advancing the controller to state 03F00U.

4-106 Read and Write Orders

When the controller enters the 03F01U state on a read or write order, that order remains true.

If FUN is true on a read order, however, it dc-sets FLE (length error flip-flop), indicating a long record.

M/FLE = READ 01U FUN + . . .

If FUN is false and BMT is true on a read order, FLE is dc-set by BMT.

M/FLE = READ 01U BMT + . . .

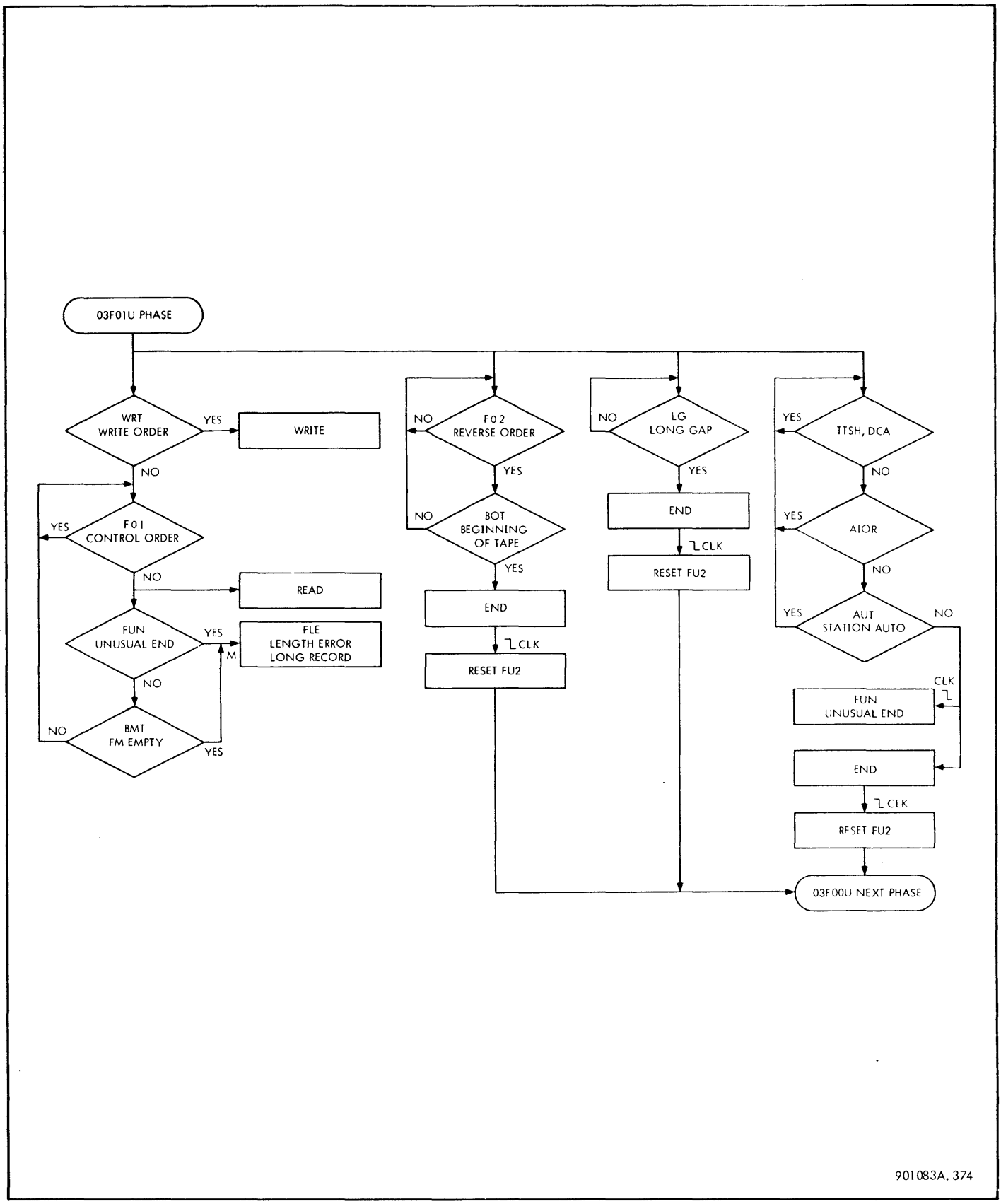


Figure 4-74. 03F01U State (Order Finalization), Flow Diagram

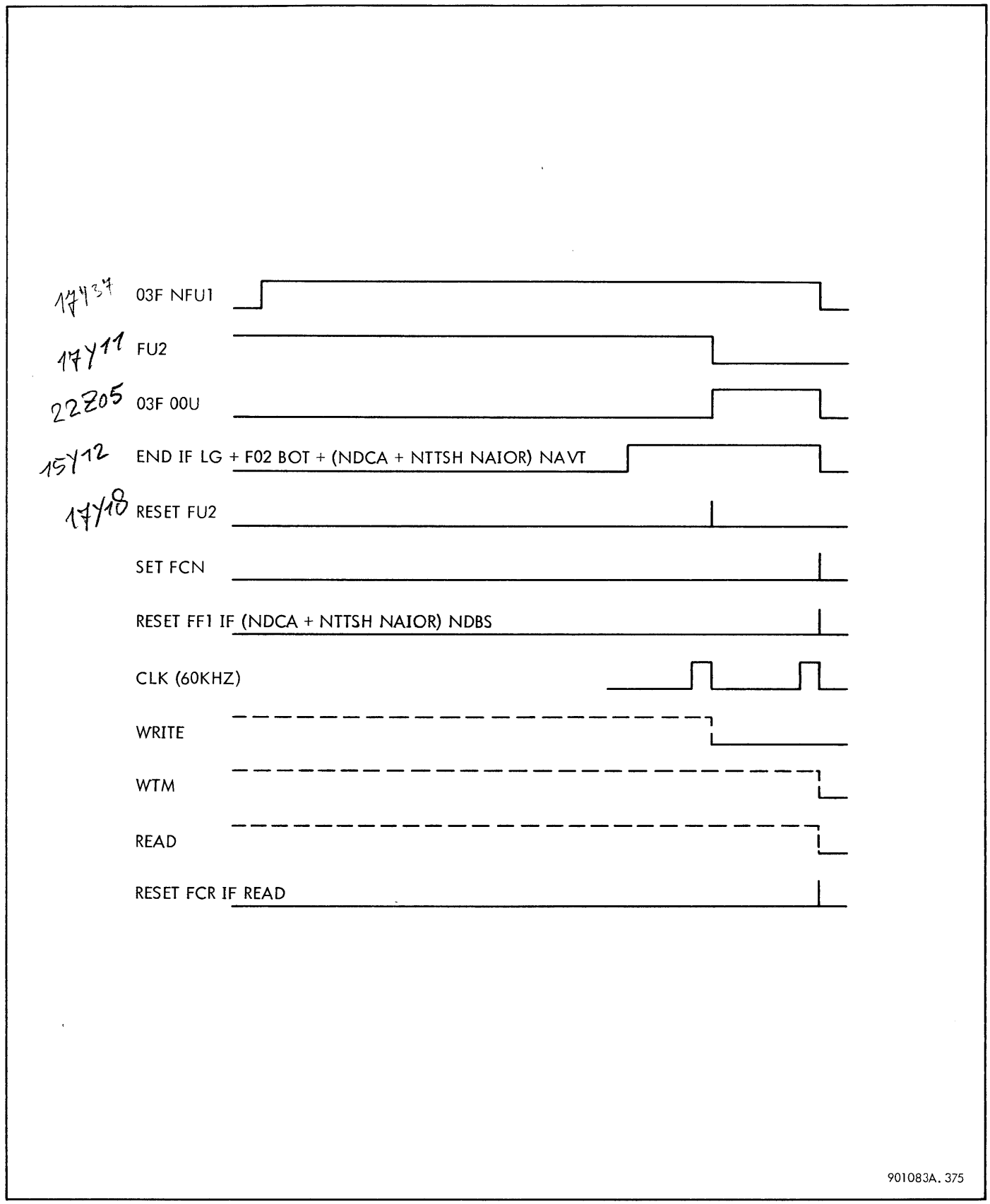


Figure 4-75. 03F03U-02U State (Termination), Timing Diagram

4-107 03F00U (STATION FINALIZATION)

The 03F00U state is the station finalization state. In this state, the read and the write orders are either carried through to the next state (order input) in a command chaining operation or the operation is terminated on an unusual end condition. See figure 4-76 for a flow diagram of the 03F00U state.

When the controller enters this state, if DCA and TTSH, AIOR, and DBS (device busy) are false and if AUTO is true, the following events occur: FF1 and FCR (set correction flip-flop) are reset, FCN (channel end) is set, and the controller advances to the 01F00U state (order input).

C/FF1 = 03F00U NOR01 NDBS CLK

NOR01 = NDCA NTSSH NAIOR

C/FCR = CLK 03F00U + . . .

C/FCN = CLK 03F00U + . . .

When CLK comes true, it makes the clock inputs of FF1, FCR, and FCN true; when it goes false, it clocks and resets FF1 and FCR and sets FCN. The controller then advances to state 01F00U (order input).

If, however, AUTO is false, FUN (unusual end) is set.

C/FUN = BAND29 CLK 03F

BAND29 = NDCA NTTSH NAIOR NAUT

When CLK comes true, it makes the clock input of FUN true; when it goes false, it clocks and sets FUN.

4-108 01F00U (ORDER INPUT)

The 01F00U state is the order input state. The controller enters this state when it must report errors, unusual end, and channel end to the IOP, or when the command chaining order must be sent to the state 01F01U.

The IOP receives its input orders from this state on the following DA lines:

Input Order	DA Line
Unusual End	DA4
Channel End	DA3
Chaining Modifier	DA2
Incorrect Length	DA1
Transmission Error	DA0

See figure 4-77 for a flow diagram of the 01F00U state.

4-109 Unusual End

When the controller enters this state, it generates a service call by raising CSL.

CSL = 01F00U + . . .

See figure 4-78 for a timing diagram of the 01F00U state. The IOP acknowledges the service call request by raising ASC, FS, and AVI. The set and clock inputs of service connect flip-flop FSC are made true, and when FSR (from the IOP) goes false, FSC is clocked and sets. At the same time, RSD, DORD, EDD, and DA4 are made true.

The IOP accepts the data and raises ESR and RSA (request strobe acknowledge). RSAR (receiver output from RSA) then comes true. When RSAR comes true, RSD goes false. When RSD goes false, it clocks and resets FSC. If there is no terminal order, RSD also resets FF2, and if there is an unusual end condition, it resets FCR (correction flip-flop) and FRS (erase flip-flop).

C/FCR = RSD 01F00U FCR + . . .

C/FRS = RSD 01F00U FRS + . . .

C/FF2 = RSD 01F00U + . . .

The controller now advances to state 00F00U (ready state), where it waits for the next order.

4-110 Channel End and Interrupt

The controller raises CSL, and the IOP acknowledges the service call request by raising ASC, FS, and AVI. FSC is clocked and set by FSR (from the IOP), and RSD, DORD, EDD, and DA3 are made true.

The IOP accepts the data and raises RSA, and RSAR comes true. When RSAR comes true, RSD goes false. When RSD goes false, it clocks and sets FU3.

C/FU3 = RSD FSCC

The IOP drops RSA, and RSD comes true again. The IOP raises ESR and RSA, and RSD goes false. When RSD goes false, it clocks and resets FSC and FU3.

If DA0 (interrupt) is true, FIO is set when RSD goes false.

C/FIO = (FU3 NDST) ESR RSD1 NFIO + . . .

If DA3 is true, FUN is set, and FCR, FRS, and FF2 are reset when RSD goes false.

The controller has advanced to state 00F00U (ready) on a channel end or interrupt.

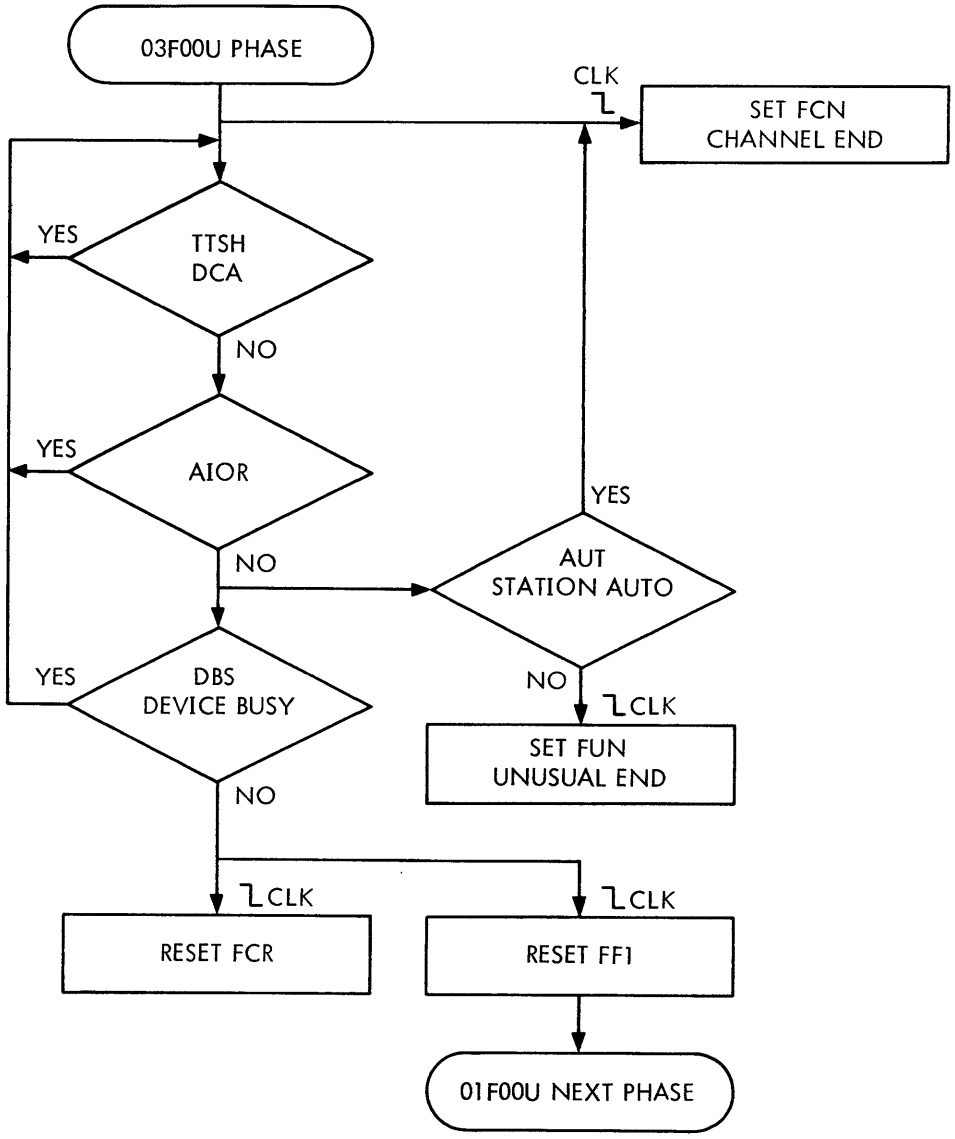


Figure 4-76. 03F00U State (Station Finalization), Flow Diagram

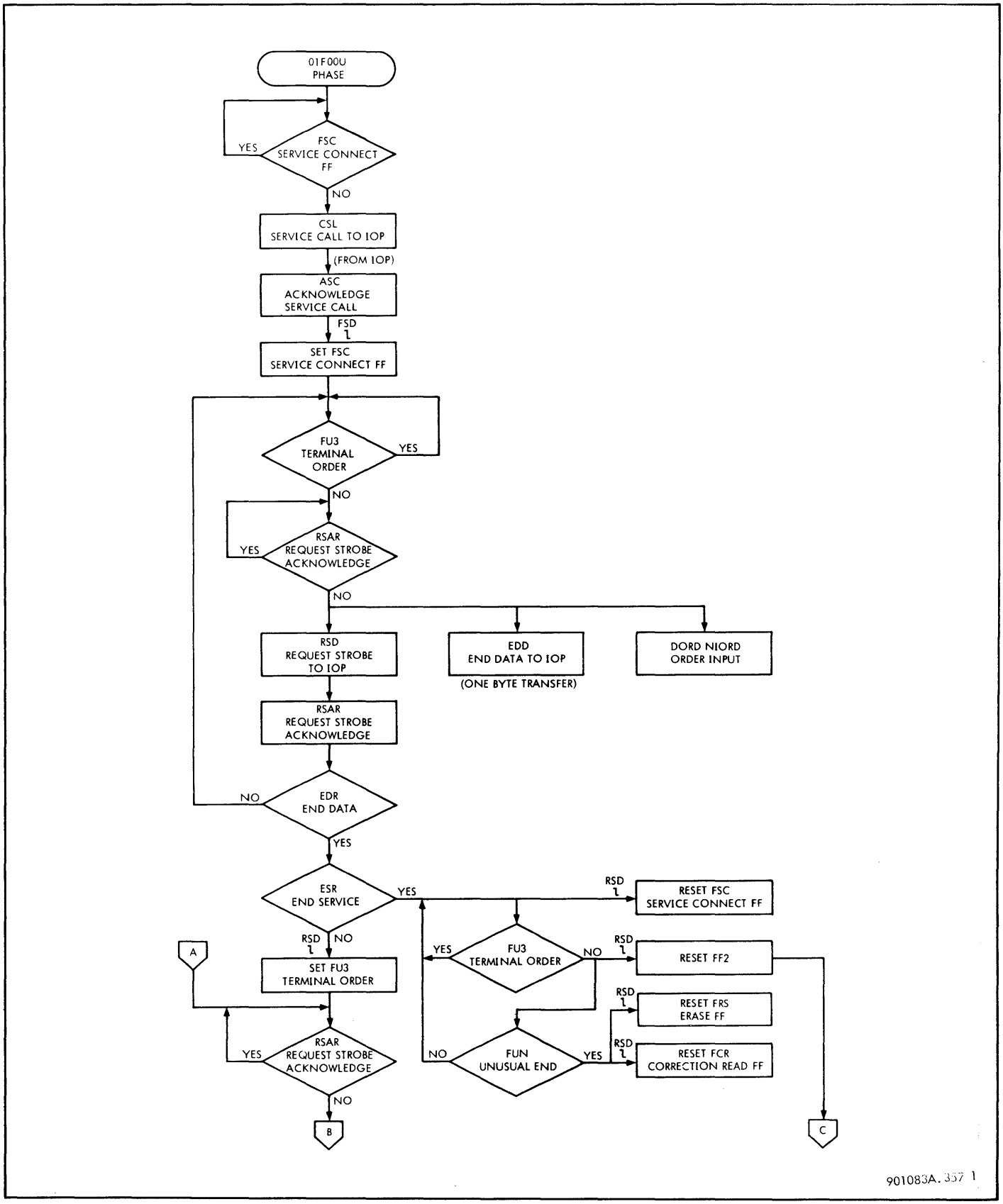


Figure 4-77. 01F00U State (Order Input), Flow Diagram (Sheet 1 of 2)

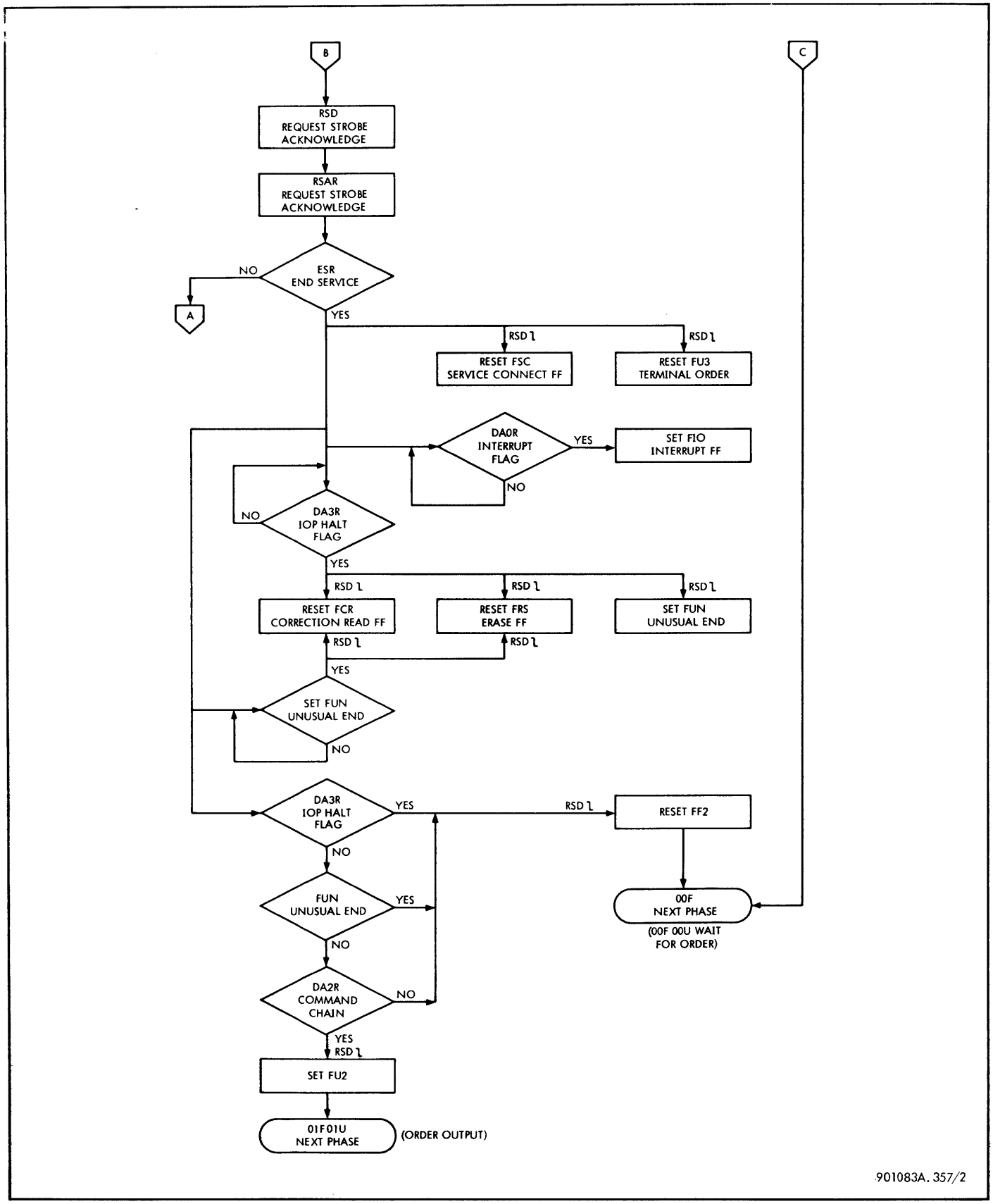


Figure 4-77. 01F00U State (Order Input), Flow Diagram (Sheet 2 of 2)

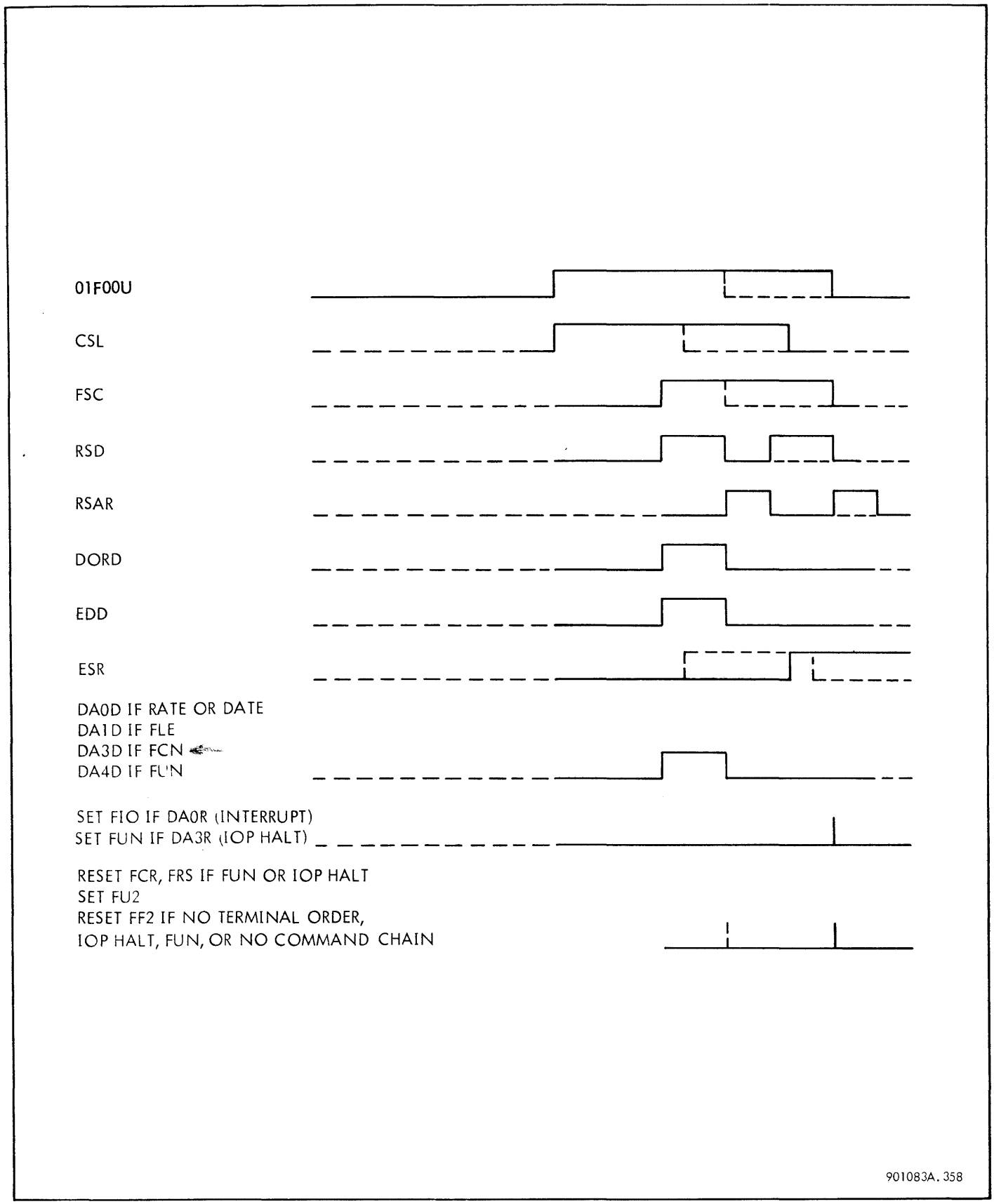


Figure 4-78. 01F00U State (Order Input), Timing Diagram

4-111 Command Chaining

The controller raises CSL, and the IOP acknowledges the service call request by raising ASC, FS, and AVI. FSC is clocked and set by FSR (from the IOP), and RSD, DORD, EDD, and DA2 are made true.

The IOP accepts the data and raises RSA, and RSAR comes true. When RSAR comes true, RSD goes false. When RSD goes false, it clocks and sets FU3. The IOP drops RSA, and RSD comes true again. The IOP raises ESR and RSA, and RSD goes false. When RSD goes false, it clocks and resets FSC and FU3.

If IOP halt (DA3) and FUN are false and command chain (DA2) is true, FU2 sets when RSD goes false, and the controller advances to state 01F01U (order output) on a command chain order.

C/FU2 = 01F00U RSD

The controller now continues with the order that it contains (read and write) until the particular operation is completed.

SECTION V
LOGIC EQUATIONS AND GLOSSARY

5-1 INTRODUCTION

This section includes information concerning the logic equations, dictionary, pin list, and pin index which reflect the implementation and the wiring of the Magnetic Tape Controller Model 7320. A complete logic glossary of terms is also included in this section.

5-2 DESIGN AUTOMATION DOCUMENTS

The following design automation documents must be available on site for maintenance purposes. The documents are separated into two groups: those which reflect the implementation and the wiring of the V, W, and X chassis and those which reflect the Y and Z chassis.

5-3 V, W, and X Chassis

Document No.	Description
145610-001	Logic Equations
145610-100	Logic Dictionary
145608-001	Pin List
145608-960	Pin Index

5-4 Y and Z Chassis

Document No.	Description
145612-001	Logic Equations
145612-100	Logic Dictionary
145606-001	Pin List
145606-968	Pin Index

5-5 SIGNAL GLOSSARY

Tables 5-1 and 5-2 contain a glossary of signals for the magnetic tape controller, chassis Y and Z and chassis V, W, and X, respectively.

Table 5-1. Glossary of Terms, Chassis Y-Z

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
00F	State counter, phase 0
01F	State counter, phase 1
03F	State counter, phase 2
00U	State counter, subphase 0
01U	State counter, subphase 1
02U	State counter, subphase 2
03U	State counter, subphase 3
01F00U	Order input state
01F01U	Order output state
01F02U	Receive and decode order state

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
01F03U	Waiting for device proceed signal state
03F00U	Order finalization state
ABD	Data enable, read mode, chassis Y, Z to IOP
AIOR	Acknowledge interrupt signal from IOP
AIOC	Acknowledge interrupt signal from IOP (Interrupt status call)
AIOM	Acknowledge interrupt signal from IOP (Ready for AVIR)
ARG	Address recognized by station
ASCR	Acknowledge service call receiver output signal from IOP's ASC
ASCM	Acknowledge service call, priority determinant
ASCB	Acknowledge service call, buffer
AIO	Signal that initiates HIO operation in chassis Y, Z
AUT	Indication that station is in automatic mode (from station)
AVI	Available signal input (from preceding controller)
AVO	Available signal output (to succeeding controller)
AVIR	Available signal receiver output (for controller use)
AVOD	Available signal driver (from chassis Y, Z logic)
BAND XX	Miscellaneous logic terms (XX = 01-32)
BOR XX	Beginning of record (XX = 02-07)
BFSD	Buffered FSD signal (for status transfer logic)
BMT	Indication that buffer memory needs service (from chassis V, W, X)
BOT	Addressed station at beginning of tape (from station)
BSYC	Indication that controller is busy processing an order
BTS	Beginning of tape response signal for TDV (from station)
CFCNX	Clock signal for FCN
CFCNY	Clock signal for FCN
CFCR	Clock signal for FCR
CFCRX	Clock signal for FCR
CFCRY	Clock signal for FCR

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
	Note When x appears as part of the term, it represents a number 0 through 7, except as indicated
CFF1	Clock signal for FF1
CFF1X	Clock signal for FF1
CFF2	Clock signal for FF2
CFIO	Clock signal for FIO
CFLE	Clock signal for FLE
CFPET5	Clock signal for FPET5
CFRS	Clock signal for FRS
CFRSX	Clock signal for FRS
CFSCM	Clock signal for FSC
CFU1	Clock signal for FU1
CFU1X	Clock signal for FU1
CFU1Y	Clock signal for FU1
CFU1Z	Clock signal for FU1
CFU1ZA	Clock signal for FU1
CFUZ	Clock signal for FUZ
CFUZX	Clock signal for FUZ
CFUZY	Clock signal for FUZ
CFU3	Clock signal for FU3
CFUNX	Clock signal for FUN
CIH	High priority interrupt signal
CIL	Low priority interrupt signal
CLB	Clock signal for FRSD and FRSDD
CLK	General clock signal (from chassis V, W, X)
CORE	Indication of correctable read error
CRD	Indication controller that is ready for order
CSH	High priority service request
CSL	Low priority service request
CSLI	Service request inhibit signal (100 ns delay of NFSC)
CTL	Indication that control order is being interpreted
DAx	Data line x to IOP from controller, or to controller from IOP
DAxD	Data line x driver

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, if represents a number 0 through 7, except as indicated	
DAxR	Data line x receiver
DAxC	Data line x to station from chassis Y, Z or from chassis Y, Z to station
DACFD	Indication that device address compares with device in operation
DACFDD	Indication that addressed device be given halt signal
DAI5	Device number transferring to FR lines
DAI6	Device number transferring to FR lines
DAI7	Device number transferring to FR lines
DATE	Data error signal (from chassis V, W, X)
DBS	Device busy signal (from station)
DCA	Indication that controller address is recognized
DCA47	Indication that station address is recognized
DDAx	Data lines having octal code of x
DFDx	Device in operation has octal x
DOR	Data order line to or from IOP
DORD	Data order line driver
DORR	Data order line receiver
DPR	Device proceed signal (from station)
DPRNWN	Order for device proceed, not rewind
DRD	Device ready signal (from station)
DSG	Device selected signal (from station)
DSL	Device selected clock signal (to station)
DSS	Device selection gate enable (to station)
DST	Device selection time enable (to station)
DVxC	Device address line (to or from station)
DVxD	Device address line driver input
DVxR	Device address line receiver output
ED	End data line (to IOP from or to controller from IOP)
EDB	Indication of less than four bytes in buffer memory at long gap
EDD	End data line driver
EDR	End data line receiver

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
END	End operation signal to station
ENT	End of tape signal from station
ES	End service line (to IOP from controller or to controller from IOP)
ESR	End service signal
ESRC	Receiver output of IOP's end service line
EXTCLK	Clock signal (to PET panel)
EXTRES	Reset signal (to PET panel)
FCD	Indication that AIO is being processed
FCN	Channel end signal storage
FCR	Set correction signal storage
FD1	Device in operation address, bit 1
FD2	Device in operation address, bit 2
FD3	Device in operation address, bit 3
FF1	Phase counter, stage 1
FF2	Phase counter, stage 2
FIO	Interrupt signal storage
FLE	Incorrect length error storage
F01	Order code register, stage 1 (MSB)
F02	Order code register, stage 2
F03	Order code register, stage 3
F04	Order code register, stage 4
F05	Order code register, stage 5 (LSB)
FPE	Indication that write protect violation occurred
FPET1	PET panel record counter, stage 1 (MSB)
FPET2	PET panel record counter, stage 2
FPET3	PET panel record counter, stage 3
FPET4	PET panel record counter, stage 4
FPET5	PET panel record counter, stage 5 (LSB)
FPET6	PET panel byte counter, stage 6 (MSB)
FPET7	PET panel byte counter, stage 7
FPET8	PET panel byte counter, stage 8 (LSB)

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
FPETIE	Error insertion signal in PET operations
FRx	Function response line (to IOP)
FRxD	Function response line driver
FRS	Indication of erase order storage
FRSD	Request strobe delay 1
FRSDD	Request strobe delay 2
FS	Function strobe (from IOP)
FSC	Indication that controller is connected to IOP for service
FSCC	Indication that controller is connected for service
FSCL	Signal to extend RSAR until FSC is reset
FSCM	Indication that controller is connected to PET for service
FSD	Function strobe driver
FSL	Function strobe leading signal (to IOP)
FSLD	Function strobe leading signal driver
FSR	Function strobe receiver output from IOP's FS
FU1	Subphase counter, stage 1
FU2	Subphase counter, stage 2
FU3	Subphase counter, terminal order next
FUN	Unusual end storage
GBS	Indication that controller is now busy
GDA	Signal to gate data lines to device lines during FS
GFD	Signal to gate active device number to device lines except during an FS response
GFI	Signal to gate interrupt latch to highest priority device during AIO
GNT	Signal to gate all interrupting devices to interrupt latch in AIO
HIO	HIO signal line (from IOP)
HLTD	Halt signal to be sent to station
HPI	Indication of high priority interrupt level
HPS	Indication of high priority service level

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
IC	Interrupt line (to IOP)
ICD	Interrupt line driver
INx	Interrupt latch, station x
INC	Initialize controller signal
INI	Initialize controller signal
INP	Indication that addressed device has interrupt pending
INT	Indication that device interrupt is occurring
IOR	Input/output signal line (to or from IOP)
IORD	Input/output signal line driver
IVO	Indication that invalid order has been received in controller
LD	Signal to send read with correction byte to chassis V, W, X
LG	Indication that long gap has been detected by chassis V, W, X
LIH	High priority interrupt latch
LIL	Low priority interrupt latch
LIRS	Signal to inhibit new request strobe until disconnected
LSH	High priority service request latch
LSL	Low priority service request latch
MAN	Indication that controller is operating from PET
MANA	Selector toggle switch output
PC	Indication to check parity on transmitted byte
PCD	Cable driver signal from ABD
PETCx	PET counter, stage x
PETCD	PET counter reset
PETCMPA	Strobe signal to compare data in PET
PETCTR	PET data control signal
PETDA1	PET data control signal
PETDA2	PET data control signal
PETDA3	PET data control signal
PETFB	Signal to write or read a fixed number of records

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
PETIE	Error insertion signal of records
PETPATA	PET data control signal
PETPATA B	PET data control signal
PETPATA C	PET data control signal
PETPATA D	PET data control signal
PETPATA X	PET data control signal
PETREV	Reverse control signal from PET
PETRP	Repeat cycle signal from PET
PETST	Start signal from PET
PETW	Write signal from PET
PETWDx	Data signal for comparison in PET
PETWDP	Parity signal from PET
PETWDPA	Parity signal for comparison in PET
PETWTM	Write tape mark signal from PET
RxB	Data line x from chassis V, W, X
RxBA	Data line x to PET
RASC	Erase signal (to station)
RATE	Rate error signal (from chassis V, W, X)
RDP	Indication to read data on the line (from chassis V, W, X)
READ	Indication that read operation is in progress
RES	General reset signal (to chassis V, W, X)
RESIN	SIO reset signal (to chassis V, W, X)
REV	Indication that reverse operation is in progress
RS	Request strobe line (to IOP)
RSDA	Request strobe line driver
RSA	Request strobe acknowledge line (from IOP)
RSARC	Request strobe acknowledge line receiver
RSAR	Request strobe acknowledge signal
RSD	Request strobe signal
RSDD	Request strobe signal delayed
RST	Controller reset signal

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears as part of the term, it represents a number 0 through 7, except as indicated	
RVRC	Indication that reverse operation is in progress (to station)
RWFIN	Indication that read/write data transmission has been completed
SC	Service call line (to IOP)
SCD	Service call line driver
SCO	Indication that set correction operation is in progress
SCOD	Indication that set correction code is in order code register
SEP	Indication that sense track in error byte is in operation
SEPD	Indication that sense track in error byte code is in register
SIO	Start input/output line (from IOP)
SIOR	Start input/output line receiver
SPF	Indication that space file operation is in progress
SPR	Indication that space record operation is in progress
SRIP	Memory access request (to chassis V, W, X)
SRIPA	Memory access request acknowledge (from chassis V, W, X)
SWAx	Address code switch x
TDV	Test device line (from IOP)
TDVR	Test device line receiver output
TIO	Test controller line (from IOP)
TIOR	Test controller line receiver
TM	Indication that tape mark has been detected (from chassis V, W, X)
TSH	Indication that TIO, SIO or HIO operation is in progress
TTSH	Indication that TDV, TIO, SIO or HIO operation is in progress
WDC45C	Clock signal from chassis V, W, X
WN1	Rewind and interrupt code in order register
WN1C	Rewind and interrupt code to station, 1
WN2	Rewind offline code in order register
WN2C	Rewind and interrupt code to station, 2
WN102	Rewind code in order register
WND	Indication that station is in rewind

(Continued)

Table 5-1. Glossary of Terms, Chassis Y-Z (Cont.)

Term	Definition
Note When x appears on part of the term, it represents a number 0 through 7, except as indicated	
WPM	Indication that station in write protect mode
WRITE	Indication that write operation is in progress (to chassis V, W, X)
WRT	Indication that write operation is in progress
WRTA	Indication of a write operation by controller
WRTB	Indication of a write operation by PET
WRTC	Write operation signal (to station)
WTM	Indication that write tape mark operation is in progress

Table 5-2. Glossary of Terms, Chassis V-W-X

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
xB	Indication of bit x data bus input to bit x of CRC register (x = 1 - 9)
1BCCx	Read deskew counter, channel x, stage 1
2BCCx	Read deskew counter, channel x, stage 2
3BCCx	Read deskew counter, channel x, stage 3
1PKDETx	Peak detector counter, channel x, stage 1 (LSB)
2PKDETx	Peak detector counter, channel x, stage 2
3PKDETx	Peak detector counter, channel x, stage 3
4PKDETx	Peak detector counter, channel x, stage 4
5PKDETx	Peak detector counter, channel x, stage 5
6PKDETx	Peak detector counter, channel x, stage 6 (MSB)
1RDSCx	Bit crowding counter, channel x, stage 1 (LSB)
2RDSCx	Bit crowding counter, channel x, stage 2
3RDSCx	Bit crowding counter, channel x, stage 3
4RDSCx	Bit crowding counter, channel x, stage 4
5RDSCx	Bit crowding counter, channel x, stage 5 (MSB)
ACSG	Indication that all zero CRC character has occurred

(Continued)

Table 5-2. Glossary of Terms, Chassis V-W-X (Cont.)

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
APC1	Assembly period counter, stage 1 (LSB)
APC2	Assembly period counter, stage 2
APC3	Assembly period counter, stage 3
APC4	Assembly period counter, stage 4
APC5	Assembly period counter, stage 5
APC6	Assembly period counter, stage 6
APC7	Assembly period counter, stage 7
APC8	Assembly period counter, stage 8 (MSB)
APCC1	First bit detector for assembly period counter
APCC2	Assembly period counter enable
APCZ	Assembly period counter reset
ARx	Assembly register, bit x
BMT	Memory service signal (to chassis Y, Z)
BOTC	Beginning of tape signal (to chassis Y, Z)
Cx	CRC register, bit x (x = 1-9)
CCRC	CRC verify interval
CD1	Master oscillator divider, stage 1
CD2	Master oscillator divider, stage 2
CLx	Channel x clock signal
CLA	Buffered clock signal
CLB	Buffered master oscillator
CLKE	CRC compare error signal
CLOCK	Clock signal
CMPR	Indication that CRC character compares
CNCRE	CRE clock signal
CORE	Correctable read error
CORRx	Bit crowding correction enable signal, channel x
CRBx	Bit crowding correction output, channel x
CRCC	Indication that CRC character checks correctly
CRCT	Read CRC interval

(Continued)

Table 5-2. Glossary of Terms, Chassis V-W-X (Cont.)

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
CRE	Correctable read error signal
CROK	Indication that CRC contains good CRC comparison
CRWFIN	Indication that controller has finished clock
CWR	Write deskew counter reset
DAxR	Data line bit x (from chassis Y, Z)
DATE	Data error (to chassis Y, Z)
DBx	Input to CRC register, channel x
DCx	Input to error pattern register, channel x
DJAxC	Channel x deskew jumper, stage A
DJBxC	Channel x deskew jumper, stage B
DSCxC	Channel x deskew jumper, stage C
DLA	Delay signal in memory access logic
DLA1	Delay signal in memory access logic
DLA2	Delay signal in memory access logic
DLAP	Delay signal in memory access logic
DLBP	Delay signal in memory access logic
DLCP	Delay signal in memory access logic
DLDP	Delay signal in memory access logic
DLEP	Delay signal in memory access logic
DLEPA	Delay signal in memory access logic
DPR	Device proceed signal (from station to chassis Y, Z)
Ex	Error pattern register, bit x (x = 1-9)
EAP	Indication of end of assembly period
EAP1	Indication of end of assembly period, first clock
EAP2	Indication of assembly period, second clock
EDB	End data signal (to chassis Y, Z)
EPRL	Error pattern register load signal
EPRS	Error pattern register shift signal
ERZ	Error pattern register reset

(Continued)

Table 5-2. Glossary of Terms, Chassis V-W-X (Cont.)

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
FCR	Indication that read with correction is in progress (from chassis Y, Z)
FIN	Data finished signal (from chassis Y, Z)
FMx	Memory output signal, bit x
LD	Indication to load read with correction byte into EP register
LG	Long gap
LRCT	Read LRC interval
LTMD	Tape mark detect latch
MC1	Indication of one character in memory
MC8	Indication of eight characters in memory
MCA	Memory character counter (LSB)
MCB	Memory character counter
MCC	Memory character counter
MCZ	Memory character counter (MSB)
MD1	Memory access cycle (memory to chassis Y, Z)
MD2	Memory access cycle (memory to chassis V, W, X)
MD3	Memory access cycle (chassis Y, Z to memory)
MD4	Memory access cycle (chassis V, W, X to memory)
MRA	Read memory character address
MRB	Read memory character address
MRC	Read memory character address
MWA	Write memory character address
MWB	Write memory character address
MWC	Write memory character address
ODD	Indication that parity of CRC character is odd
OSCLO	Master oscillator clock (3840 kHz)
PC1	Indication that parity of bits 0, 1, 2 of read register is odd
PC2	Indication that parity of bits 3, 4, 5 of read register is odd

(Continued)

Table 5-2. Glossary of Terms, Chassis V-W-X (Cont.)

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
PC3	Indication that parity of bits 6, 7, P of read register is odd
PC4	Indication that parity of all bits of read register is odd
PDQx	Peak detector qualify signal, channel x
PDTx	Peak detector transfer signal, channel x
PE	Parity error
PES	Character parity error
PR1	Indication that controller has access to memory
RxB	Transfer of data bus input to write register, bit x
RATE	Rate error (to chassis Y, Z)
RBDx	Deskew read data output, channel x
RCP	Signal that read register is ready for transfer to memory
RCPR	Signal that read register transfer is completed
RDAMPxC	Read amplifier signal, channel x
RDP	Read data present signal (to chassis Y, Z)
RDSCCx	Deskew counter reset, channel x
READ	Read mode (from chassis Y, Z)
REAPC	Reset assembly period counter
REPKDETx	Reset peak detector counter, channel x
RES	General reset signal (from chassis Y, Z)
RESIN	SIO reset signal (from chassis Y, Z)
REV	Indication that reverse operation is in progress
RRx	Read register, stage x
RRCx	Read with correction data bus, channel x
RTER	Indication that rate error is set, read mode
RW1	Memory output to data bus transfer
RW2	CRC character to write register transfer
RW3	Tape mark or LRC to write register transfer
RW4	Track-in-error signal transfer (to chassis Y, Z)
RWFIN	Data finished signal (from chassis Y, Z)

(Continued)

Table 5-2. Glossary of Terms, Chassis V-W-X (Cont.)

Term	Definition
Note In the following terms, x refers to channels 0, 1, 2, 3, 4, 5, 6, 7, P except as indicated	
SEP	Sense signal from chassis Y, Z to determine track-in-error
SG	Short gap signal
SHIFT	Shift signal for CRC register
SRA	Memory access cycle enable signal
SRAI	Memory access cycle enable input
SRIP	Memory access request, chassis Y, Z
SRIPA	Memory access request, chassis Y, Z acknowledge
SRIPDLY	Memory access request, chassis Y, Z delay
SRIPL	Signal in memory access logic
SRRW	Memory access request, chassis V, W, X
SRRWA	Memory access request for storage, chassis V, W, X
TM	Indication of tape mark detected storage
TMC	Indication of tape mark character in read register
TMD	Indication that tape mark character is detected
WAIT	Indication that chassis Y, Z have not sent first byte, write mode
WCRC	Write CRC character
WDC1C	Write deskew counter, stage 1 (LSB)
WDC2C	Write deskew counter, stage 2
WDC3C	Write deskew counter, stage 3
WDC4C	Write deskew counter, stage 4
WDC5C	Write deskew counter, stage 5 (MSB)
WIx	Write register input data, stage x
WLRC	Write LRC character signal
WRx	Write register, stage x
WRxCD	Write data cable driver (to station)
WRCW	Memory access request for data write mode
WRITE	Write register reset signal
WRZ	Write register reset signal
WTM	Writer tape mark mode

SECTION VI
DRAWINGS

6-1 ENGINEERING DRAWINGS

All engineering drawings which are released by manufacturing and pertain to the installation and the maintenance of the Magnetic Tape Controller Model 7320 are either referenced or included within other sections of this manual.

SECTION VII
SPECIFICATIONS AND INSTALLATION DATA

7-1 INTRODUCTION

This section contains information relative to the specifications and the installation of the Magnetic Tape Controller Model 7320.

7-2 SPECIFICATIONS

The specifications are shown in table 7-1 for the Model 7320/7322 Magnetic Tape System and for the Model 7320/7323 Magnetic Tape System, since the Model 7320 Magnetic Tape Controller is never operated without either a Model 7322 Magnetic Tape Station or a Model 7323 Magnetic Tape Station.

7-3 INSTALLATION

The Magnetic Tape Controller Model 7320 is normally installed within the cabinet before shipment, eliminating the need for such installation in the field. As part of a Model 7320/7322 Magnetic Tape System or a Model 7320/7323 Magnetic Tape System installation, refer to the Installation Material List (IML) and to the Installation Cable List (ICL) which accompany the equipment for cabling instructions. For clarification, refer to figure 7-1 which shows typical cabling information.

7-4 Installation in Cabinet

The following procedure should be followed when installing the controller in a magnetic tape system cabinet.

a. Remove the busbar from the PT16 and PT18 power supplies.

b. Remove the PT16 and PT18 power supplies from the frame.

c. Mount the magnetic tape controller chassis using 8-32 NC-2B x 3/8-inch long screws, flat washers, and locknuts (20 of each are required).

d. Install five cable routing channels and one door latch bracket using the hardware provided with the controller.

e. Replace the PT16 and PT18 power supplies.

f. Replace the busbar assembly on the PT16 and the PT18 power supplies.

g. Add a jumper wire from chassis U, connector J27, pin 42, to chassis Y, connector J23, pin 5. Route the jumper wire along the right side of the chassis and dress with the cable clamps and the screws in the mounting kit.

7-5 Cabling

Refer to the Installation Cable List (ICL) supplied with the installation documents for the proper cabling, using figure 7-1 as a guide.

7-6 Module Location Chart

The module location chart for the controller is shown in figure 7-2.

Table 7-1. Magnetic Tape System Specifications

Characteristics	Specifications	
General Characteristics:	Model 7320/7322	Model 7320/7323
Tape speed (read/write)	75 ips	150 ips
Data Transfer Rate	60 KB/s	120 KB/s
Tape speed (fast forward, rewind)	250 ips	450 ips
Start time	5.0 ms max	3.0 ms max
Start distance	0.175 (±0.025) in.	0.190 (±0.020) in.
Stop time	5.0 ms max	3.4 ms max

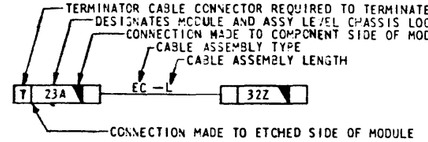
(Continued)

Table 7-1. Magnetic Tape System Specifications (Cont.)

Characteristics	Specifications	
General Characteristics: (Cont.)	Model 7320/7322	Model 7320/7323
Stop distance	0.175 (±0.075) in.	0.240 (±0.025) in.
Rewind time (2400-ft reel)	2 minutes	1 minute
Recording format	9 channel, NRZI (nonreturn to zero, change to ones), IBM 2400 series compatible	9 channel, NRZI (nonreturn to zero, change to ones), IBM 2400 series compatible
Interrecord gap	0.60 (-0.10, +0.15) in.	0.60 (-0.10, +0.15) in.
Instantaneous speed variation (ISV)	±4%	±3%
Variation of average speed	±4%	±2.5%
Dynamic skew (peak-to-peak)	225 µin.	225 µin.
Static skew (peak)	225 µin.	225 µin.
Tape Characteristics:		
Recording density	800 bits per inch	800 bits per inch
Width	1/2 in.	1/2 in.
Type	1.5 mm base Mylar	1.5 mm base Mylar
Reel size	10.5 in. diameter, 3.688 in. hub	10.5 in. diameter, 3.688 in. hub
Logic Levels:		
True level	+4.0 (±0.4)V	+4.0 (±0.4)V
False level	+0.25 (±0.25)V	+0.25 (±0.25)V
Power Requirements:		
Primary power	108 to 127 Vac, 48 to 62 Hz, Single-phase, 25A	108-127 Vac, 56-66 Hz, 3-phase, 25A/phase
Secondary power	±8.0 (±10%)V, ±25.0 (±10%)V +4.0 (±10%)V	+8.0 (±10%)V, +25.0 (±10%)V +4.0 (±10%)V
Physical Characteristics:		
Height	63-1/2 in.	63-1/2 in.
Depth	35 in.	35 in.
Width	29-1/4 in.	29-1/4 in.
Weight, master unit	950 lb	950 lb
Weight, slave unit	850 lb	850 lb
Environmental:		
Temperature range	10° to 50°C (50° to 122°F)	10° to 50°C (50° to 122°F)
Humidity range	5% to 80%	5% to 80%

LEGEND,

A.



B.

CODE	SDS PART NO.
EC-L	127314-CHARTED
T	127315 TERMINATOR

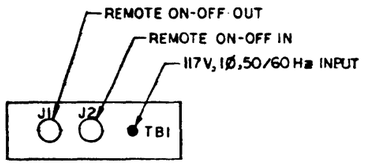
C. SYMBOL E DENOTES ETCHED SIDE OF MODULE AS VIEWED FROM MODULE SERVICE END

D. SYMBOL C DENOTES COMPONENT SIDE OF MODULE AS VIEWED FROM MODULE SERVICE END

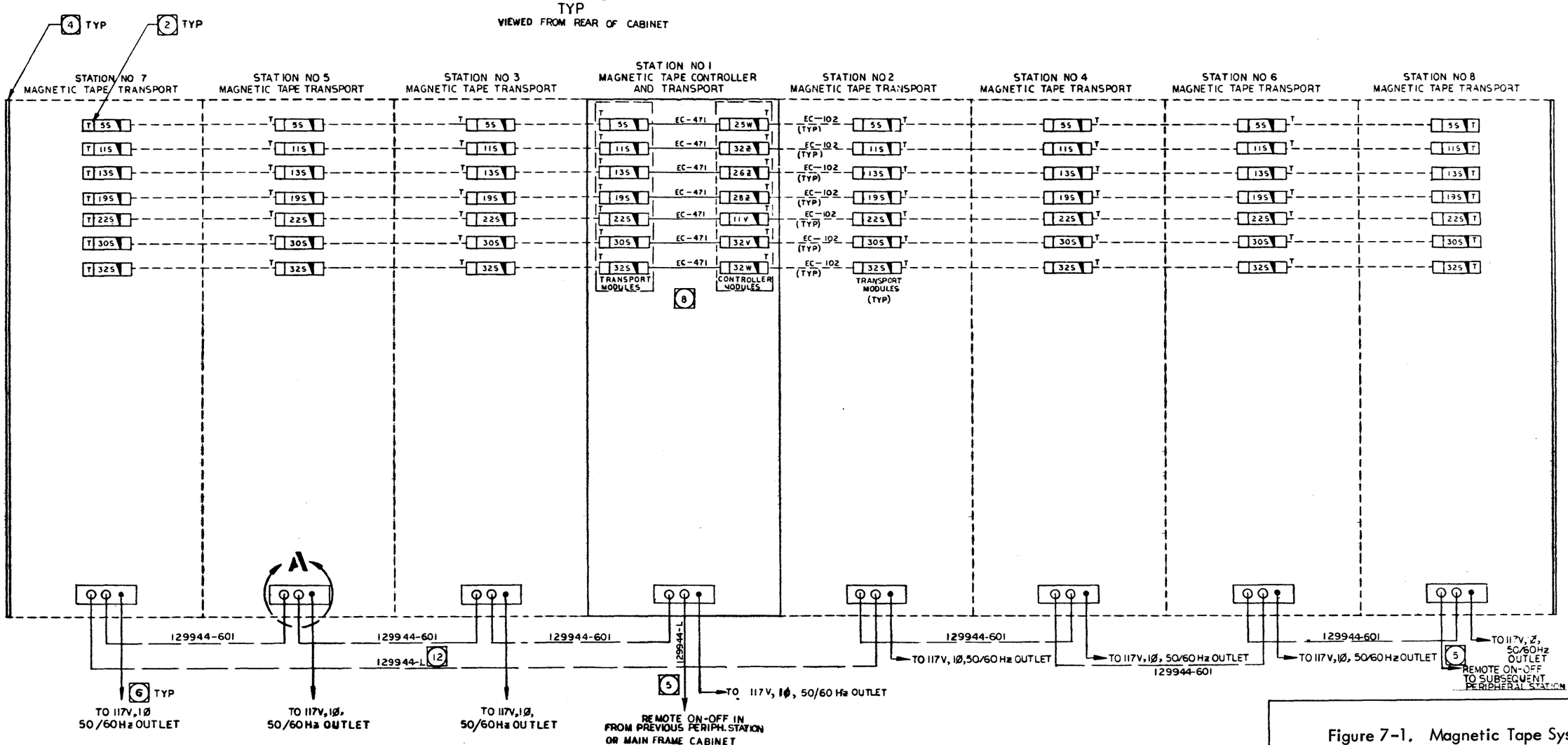
E. DENOTES TERMINATION IF CABLE CHAIN ENDS AT THIS POINT

F. DENOTES HARDWIRED CABLE

NOTE: REFERENCE SDS DWG: 137114-1,2B



DETAIL A
TYP
VIEWED FROM REAR OF CABINET



MAGNETIC TAPE SUBSYSTEM

Figure 7-1. Magnetic Tape System Interconnections (Sheet 1 of 2)

901561A. 454/1

NOTES, UNLESS OTHERWISE SPECIFIED

1. REFERENCE DRAWINGS:
   - A. INSTALLATION DRAWING, MAGNETIC TAPE STATION - 128336
   - B. INSTALLATION DRAWING, MAGNETIC TAPE CONTROLLER
   - C. INSTALLATION DRAWING, PANEL, SIDE - 131411
   - D. INSTALLATION DRAWING, SIGMA SYSTEM POWER INTERCONNECTIONS -133273

2. THE STATION INTERCONNECT CABLE CHAIN MUST BE TERMINATED AT EXTREME ENDS WITH CABLE TERMINATOR 127315

3. A MAGNETIC TAPE SUBSYSTEM MAY BE EXPANDED BY ADDING TRANSPORTS (STATIONS) TO EITHER SIDE OF THE STATION CONTAINING THE CONTROLLER. A NUMERICAL BALANCE OF ADDED STATIONS MUST BE MAINTAINED TO THE LEFT AND RIGHT OF THE CABINET CONTAINING THE CONTROLLER. NO MORE THAN FOUR TRANSPORTS CAN BE CHAINED ON EITHER SIDE OF THE CONTROLLER, I.E. FOUR MAXIMUM PER SIDE.

4. INSTALL RESPECTIVE SIDE PANELS AT THE EXTREME ENDS OF THE MAGNETIC TAPE SUBSYSTEM.

5. INCOMING REMOTE ON-OFF POWER CABLE CONNECTS TO THE REMOTE ON-OFF POWER "IN" PLUG (J2) OF THE STATION HOUSING OF THE CONTROLLER. THE REMOTE ON-OFF SIGNAL FROM THE LAST STATION IS AVAILABLE FOR CONNECTION TO SUBSEQUENT PERIPHERAL STATIONS IN AN INSTALLATION

6. INDEPENDENT POWER CABLE FOR EACH STATION CONNECTS TO A SOURCE OF 117V ± 10%, 50/60 HZ (PROVIDED BY CUSTOMER). A 30 AMP WALL CIRCUIT BREAKER MUST BE PROVIDED FOR EACH OUTLET. A POWER CABLE IS PROVIDED WITH EACH CABINET (12 FT.)

7. FOR CONNECTIONS OF 8 BIT (1BYTE) INTERFACE, CONTROL AND PRIORITY CABLES TO IOP INTERFACE OR OTHER DEVICE CONTROLLERS SEE INSTALLATION DWG, SIGMA I/O INTERFACE SUBSYSTEM - 137113

8. SEE SHEET 2 FOR THE INTERCONNECT CABLE ROUTING FROM CONTROLLER TO TRANSPORT CHASSIS OF STATION NO. 1 (CONTAINED WITHIN SAME CABINET)

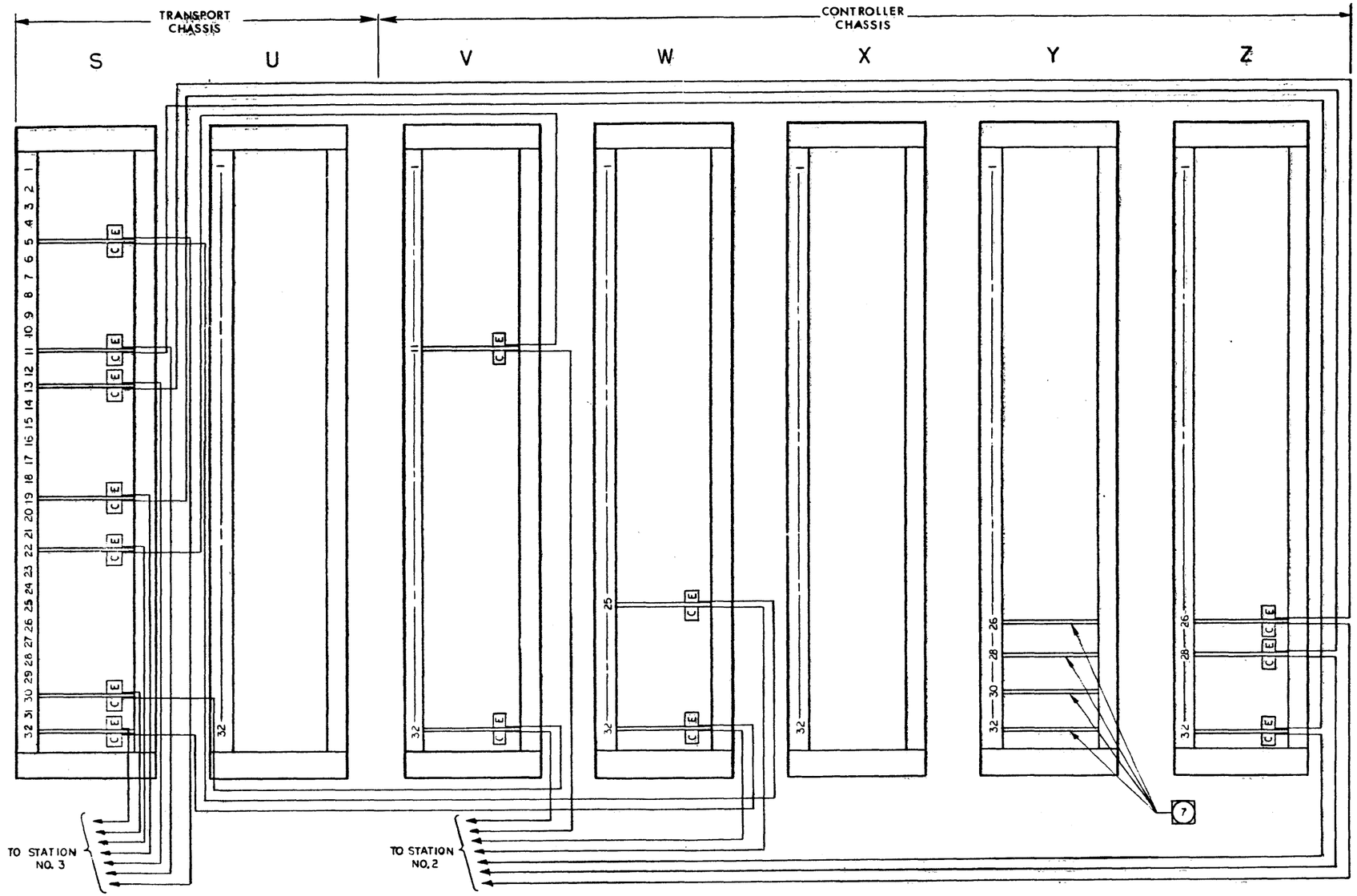
9. ROUTE STATION TO STATION SIGNAL CABLES FROM THE HINGE SIDE OF SWING FRAMES AND ALONG THE CABLE TROUGHS PROVIDED AT THE TOP OF CABINETS

10. ROUTE AND HARNESS ALL CABLES AS REQUIRED USING CABLE STRAP 124712

11.

12. THE REMOTE CABLE SHOULD BE CHAINED SEQUENTIALLY FROM STATION NO. 1 (CONTROLLER) TO LEFT (I.E. TO STATION NO. 3, NO. 5 ETC.) THEN FROM LEFT TO RIGHT (I.E. TO STATION NO. 2, NO. 4 ECT). THE LENGTH (L) OF THE REMOTE CABLE TO STATION NO. 2 WILL VARY DEPENDING ON THE NUMBER OF STATIONS IN THE SYSTEM. ALLOW AN ADDITIONAL 6 FT. FOR EACH STATION BETWEEN EXTREME LEFT STATION AND STATION NO. 2 (L = 24 FT. FOR A QUANTITY OF 7 OR 8 STATIONS)

13. REFERENCE SDS DWG: 137114-1,3B



HINGE SIDE

INTERCONNECT CABLE ROUTING

VIEWED FROM SERVICE END OF MODULES (STATION NO.1)

Figure 7-1. Magnetic Tape System Interconnections (Sheet 2 of 2)

901561A, 454/2

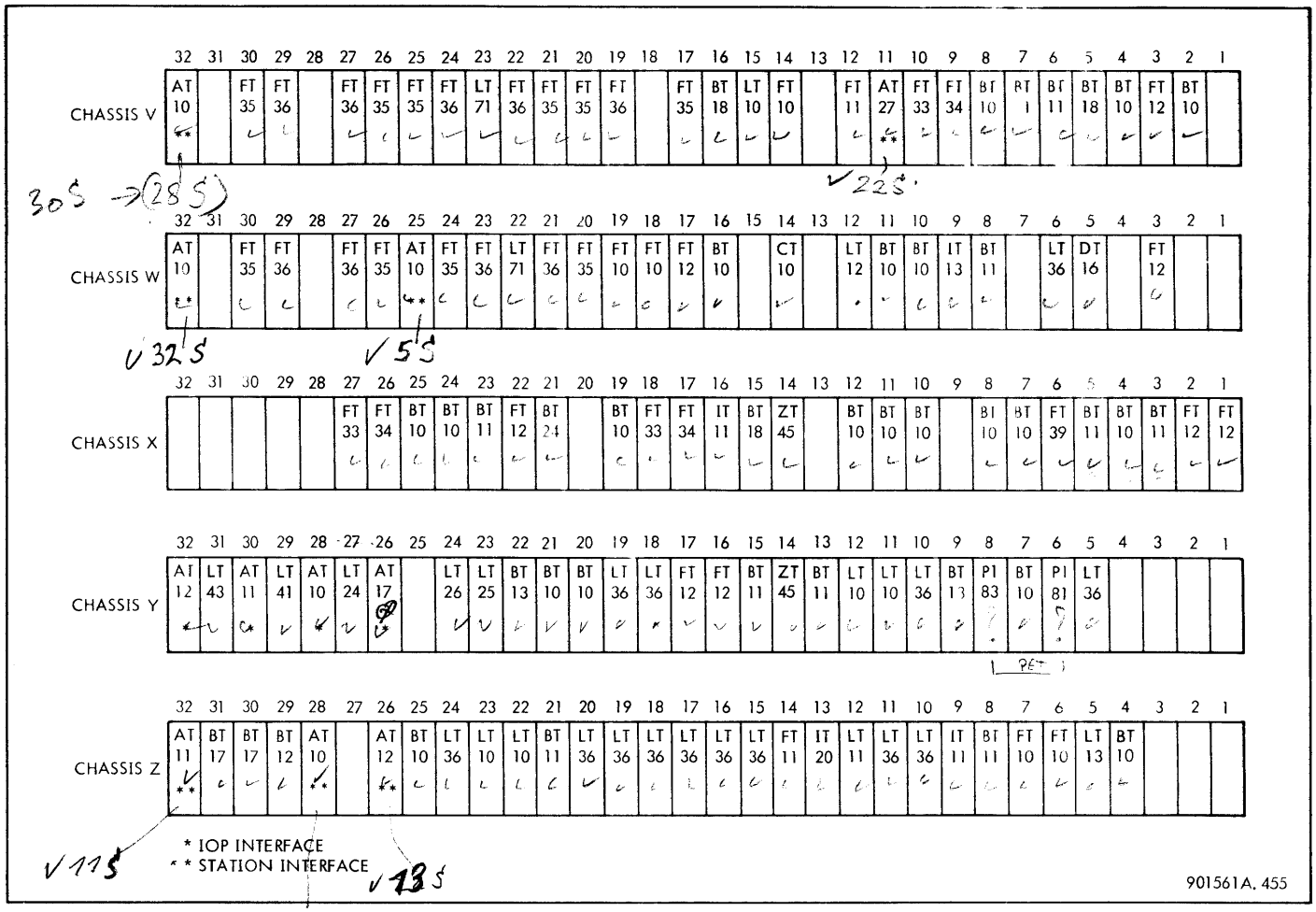


Figure 7-2. Magnetic Tape Controller, Module Locations

SECTION VIII
MAINTENANCE

8-1 INTRODUCTION

This section contains information relevant to the maintenance of the Magnetic Tape Controller Model 7320 including preventive and corrective maintenance procedures, adjustments, and applicable diagnostic programs. Maintenance information for the stations is found in the technical manuals in the List of Related Publications at the front of this manual.

8-2 PREVENTIVE MAINTENANCE

No preventive maintenance on the controller is required other than keeping all surfaces clean and free of maintenance jumpers or other "kludged" wiring.

8-3 DIAGNOSTIC PROGRAMS

The following diagnostic programs apply to the testing of the controller.

8-4 SIGMA 2 SYSTEMS

8-5 Sigma 2 Magnetic Tape Test

This diagnostic tests the magnetic tape system's capability for performing all of its designed operations. It diagnoses the system for possible malfunctions and reports such conditions. This program uses the standard diagnostic control program executive routine for control line entry.

Document	Description
704021-83	Diagnostic Program on absolute binary paper tape
704021-84	Diagnostic Program on absolute binary cards
901166	Sigma 2 Magnetic Tape Test (9 Channel, 75/150 Ips), Diagnostic Program Manual, with listing

8-6 Sigma 2 Diagnostic Control Program (DCP)

This program is the executive routine which is part of program 704021. It is not necessary to have this routine as a separate program. The listing and manual, however, should be available.

Document	Description
900839	Diagnostic Control Program for Sigma 2 Computer Peripheral Devices, Reference Manual, with listing

8-7 Sigma 2 System Test Monitor (SEVA)

This diagnostic program is made up of an executive routine which controls a series of individual tests. The purpose of the diagnostic is to allow the maintenance person to run a number of peripherals simultaneously if interference between the peripherals is suspected. This system-level diagnostic is not meant to isolate failures within a peripheral, but to isolate failures within a system.

Document	Description
704700-83	Diagnostic Program on absolute binary paper tape
704700-84	Diagnostic Program on absolute binary cards
900841	Diagnostic Control Program Sigma 2 System Test Monitor, Diagnostic Program Manual, with listing

8-8 Sigma 2 9-Channel Magnetic Tape System Test

This program is part of the Sigma 2 System Test. It has to be loaded with the Sigma 2 System Test Monitor, described in paragraph 8-7. The program includes a number of routines which exercise the magnetic tape system.

Document	Description
704701-23	Diagnostic Program on relocatable binary paper tape
704701-24	Diagnostic Program on relocatable binary cards
901149	Sigma 2 9-Channel Magnetic Tape System Test, Diagnostic Program Manual, with listing

8-9 SIGMA 5 AND 7 SYSTEMS

8-10 Sigma 5 and 7 Magnetic Tape Test

This diagnostic tests the magnetic tape system's capability for performing all of its designed operations. It diagnoses

the system for possible malfunctions and reports such conditions. This program uses the standard diagnostic control program executive routine for control line entry.

Document	Description
704066-83	Diagnostic Program on absolute binary paper tape
704066-84	Diagnostic Program on absolute binary cards
901129	Sigma 5 and 7 9-Channel Magnetic Tape Test, Diagnostic Program Manual, with listing

8-11 Sigma 5 and 7 Diagnostic Control Program (DCP)

This program is the executive routine which is part of program 704066. It is not necessary to have this routine as a separate program. The listing and manual, however, should be available.

Document	Description
900712	Diagnostic Control Program for Sigma 5 and 7 Computer Peripheral Devices, Reference Manual, with listing

8-12 Sigma 5 and 7 System Test Monitor (SEVA)

This diagnostic program is made up of an executive routine which controls a series of individual tests. The purpose of the diagnostic is to allow the maintenance person to run a number of peripherals simultaneously if interference between the peripherals is suspected. This system-level diagnostic is not meant to isolate failures within a peripheral, but to isolate failures within a system.

Document	Description
704138-83	Diagnostic Program on absolute binary paper tape
704138-84	Diagnostic Program on absolute binary cards
901076	Sigma 5 and 7 Systems Tape Monitor, Diagnostic Program Manual, with listing

8-13 Sigma 5 and 7 System Test, Magnetic Tape (9 Channels)

This program is part of the Sigma 5 and 7 System Test. It has to be loaded with the Sigma 5 and 7 System Tape Monitor, described in paragraph 8-12. The program includes a number of routines which exercise the magnetic tape system.

Document	Description
704352-23	Diagnostic Program on relocatable binary paper tape
704352-24	Diagnostic program on relocatable binary cards
901119	Sigma 5 and 7 9-Channel Magnetic Tape System Test, Diagnostic Program Manual, with listing

8-14 CORRECTIVE MAINTENANCE

8-15 SPECIAL TOOLS AND TEST EQUIPMENT

Table 8-1 lists the special tools and the test equipment required to maintain the Magnetic Tape Controller Model 7320.

Table 8-1. Special Tools and Test Equipment

Description	Manufacturer	Model or Part No.
Oscilloscope or	Tektronix	453, 545 or equivalent
Oscilloscope	Hewlett-Packard	180A or equivalent
Peripheral Equipment Tester	SDS	Model 7901
Multimeter	Triplett	Model 630NA or equivalent
Module Extender	SDS	Model ZT10 (SDS Part No. 117037)

8-16 OSCILLATOR ADJUSTMENT

The following procedure should be followed when the CT10 oscillator is suspected of being out of adjustment.

a. Turn off power.

b. Remove the CT10 module from location 14W. Insert a ZT10 module extender in this location and insert the CT10 into the ZT10.

c. Turn on power.

d. Verify that the crystal mounted on the CT10 is rated at 3.840 mc (SDS Part No. 128131-06).

e. Adjust coil L1 on the CT10 for maximum amplitude at test point A.

f. Adjust the oscillator output at 14W34 using R16 to obtain a positive pulse width of 130 nanoseconds.

g. Verify that the cycle time at 14W34 is 258 nanoseconds.

h. Verify that the WDC45 signal at 15V09 has a cycle time of 16.7 microseconds and a positive pulse width of 4.2 microseconds.

i. Turn off power; remove extender module; and insert CT10 module in 14W.

8-17 OPERATOR CONTROL PANEL INDICATOR CHECK

The following operator control panel indicators should light under the conditions indicated:

a. POWER. When primary power is applied to station.

b. LOAD. When the BOT tab is over the photosense head.

c. REWIND. When the transport is rewinding.

d. ATTENTION. When the indicator is pressed and the machine is in the manual mode and is operating.

e. FILE PROTECT. When the file protect ring is absent.

f. AUTO. When the station is in the automatic mode. (The AUTO indicator illuminates the START button.)

g. BUSY. When the station is in the automatic mode, and the controller is using the machine.

h. READY. When all power is on, all power supplies are functioning, vacuum and pressure pumps are operating, tape is properly loaded, and all interlocks are closed. Pressing the READY indicator activates a lamp test switch, which lights the lamps in all of the indicators except RESET and POWER. (The RESET switch has no lamp.)

8-18 PERIPHERAL EQUIPMENT TESTER OPERATION

The Peripheral Equipment Tester (PET) Model 7901 is used to verify the operation of approximately 90 percent of the controller logic.

8-19 Preparation and Connection

To perform the tests described in this section, connect the PET to the controller as follows:

a. Connect the power plug to 115 Vac single phase. Make sure that the POWER switch is OFF.

b. Connect the P180-P181 cable from the PET to controller slot 6Y.

c. Connect the P182-P183 cable from the PET to controller slot 8Y.

Note

If the PET is a portable unit, P180 and P182 plug in wiring side up. If the PET is a rack mounted unit, P180 and P182 plug in wiring side down.

Check PET connection and voltages as follows:

a. Turn on the power to the PET and the tape system.

b. Check the logic supply voltages in the PET and the tape system. Settings in the shaded area monitor the voltages generated in the PET, while the rest of the settings monitor the voltages in the controller. Be sure all voltages are normal at this time. There should be no indication in positions TV01 and TV02.

CAUTION

Do not use the PET voltmeter to adjust any power supplies.

8-20 Overlay

Figure 8-1 is a full-size drawing of the magnetic tape No. 12 overlay. This overlay should be used for the tests in this manual. In the event that the overlay is unavailable, figure 8-1 may be removed from this book and used for the overlay.

8-21 Online/Offline Switch

The online/offline switch on the LT25 module located in slot 23Y must be placed in the OFFLINE position for the PET to operate.

8-22 Write Operation

The write operation is set up on the PET as follows:

a. Set the PET ON-OFF/RESET switch to the OFF/RESET position.

b. Select the desired station with the DEV ADDR switches.

c. Set the command switches to 00001 (01).

d. Select the number of bytes per record using the RECORD LENGTH switches. All switches in the off (down) position cause eight bytes per record to be written, switch 2^0 causes 16 bytes to be written, and so forth.

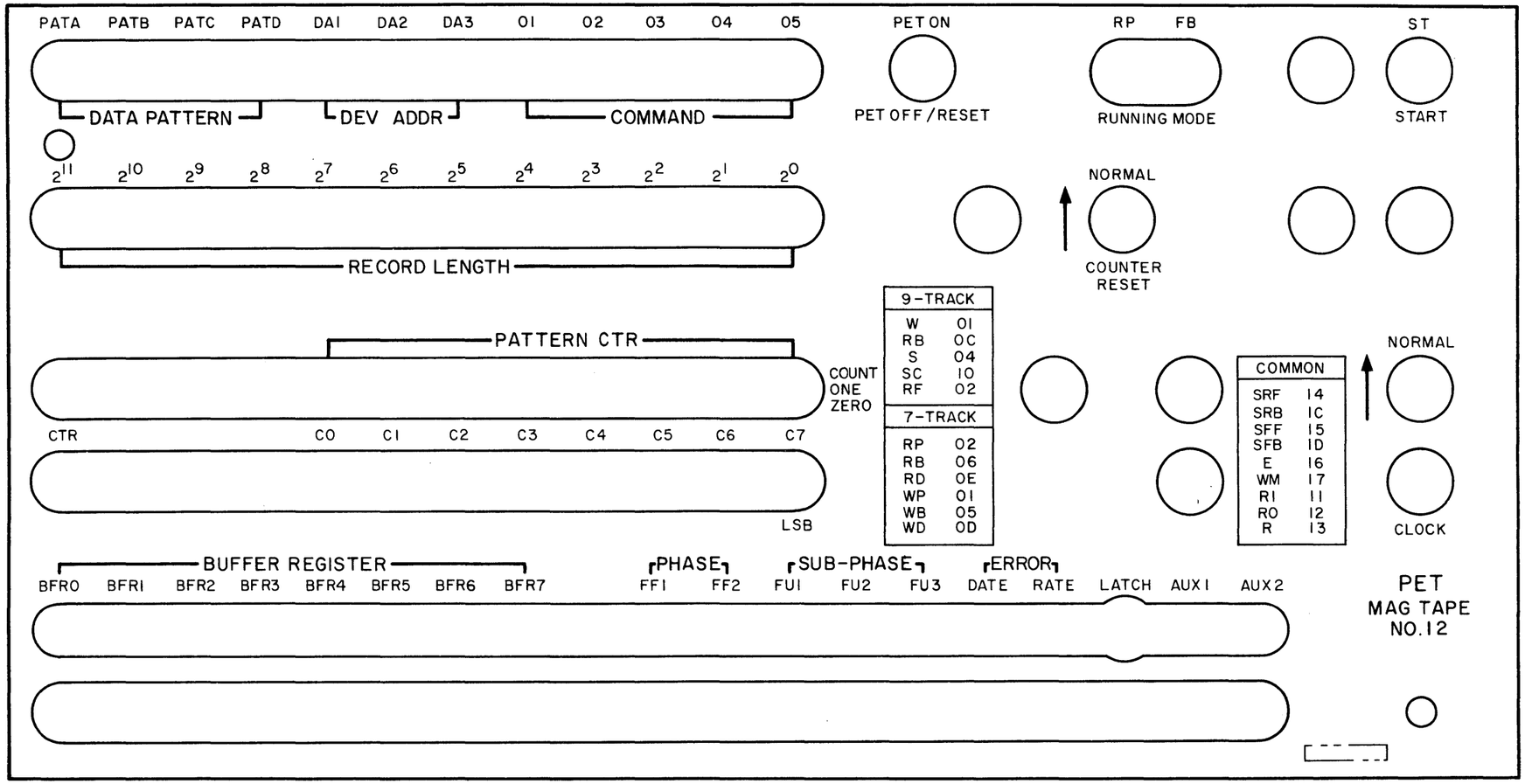


Figure 8-1. PET Magnetic Tape Overlay No. 12

901561A.401

e. Select the data pattern as follows:

1. All zero pattern. Place all PATTERN CTR switches in the ZERO position. Place CTR and all DATA PATTERN switches in the on (up) position.

2. All one pattern. Place all PATTERN CTR switches in the ONE position. Place all DATA PATTERN switches in the off position and CTR in the on (up) position.

3. Binary pattern. Data for each channel corresponds to the counter output. Place all PATTERN CTR switches in the COUNT position. Place CTR switch and RECORD LENGTH 2^7 switches in the on (up) position. Set all DATA PATTERN switches to the off position.

4. Patterns A through D. The combination of DATA PATTERN switches determines the data pattern for each channel with parity generated automatically. The switch positions for the available pattern are shown in table 8-2. Place CTR in off position and all PATTERN CTR switches to ZERO.

f. Select the mode of operation from table 8-3.

Table 8-2. PET Data Patterns

SWITCH UP	BYTE NUMBER															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
PATA	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
PATB	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
PATA and PATB	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
PATC	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
PATD	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

g. If monitoring of data channels is desired, patch from the desired monitor jack (BFR0 to BFR7) to either AUX1 or AUX2 Place LATCH/ONE-SHOT in the up position. (See table 8-4.)

h. To stop the operation, momentarily move PET ON-OFF/RESET to the OFF/RESET position.

8-23 Read Forward Operation

The read forward operation is set up on the PET as follows:

a. Set the PET ON-OFF/RESET switch to OFF/RESET.

b. Set the COMMAND switches to the 00010 (02) position.

c. Select the mode of operation from table 8-3.

d. Move the PET ON-OFF/RESET switch to the ON position.

e. Press and release the START pushbutton to initiate an operation.

f. To stop the operation, momentarily move the PET ON-OFF/RESET switch to OFF/RESET.

8-24 Read Reverse Operation

The space reverse operation is set up on the PET as follows:

a. Set the PET ON-OFF/RESET switch to OFF/RESET.

b. Set the COMMAND switches to the 01100 (0C) position.

c. Select the mode of operation from table 8-3.

d. Move the PET ON-OFF/RESET switch to on.

e. Press and release the START pushbutton to begin operation.

f. To stop the operation, momentarily move the PET ON-OFF/RESET switch to the PET OFF position.

Table 8-3. PET Functions

Overlay Designation	Panel Designation	Function
COMMAND 01 02 03 04 05	DATA CONTROL 7 8 9 10 11	Functions indicated on PET overlay and the chart below: Abbreviation — Command — Hexadecimal Representation — Switch Position* R — Read — 02 — 00010 RB — Read backward — 0C — 01100 S — Sense — 04 — 00100 W — Write — 01 — 00001 SC — Set correction — 10 — 10000 SRF — Space record forward — 14 — 10100 SRB — Space record backward — 1C — 11100 SFF — Space file forward — 15 — 10101 SFB — Space file backward — 1D — 11101 E — Set erase — 16 — 10110 WM — Write tape mark — 17 — 10111 RI — Rewind and interrupt — 11 — 10001 RO — Rewind offline — 12 — 10010 R — Rewind online — 13 — 10011 *1 = Up; 0 = Down
CTR	COUNTER OUTPUT 0	Must be in up position for PATTERN CTR switches to be effective. If CTR is in down position, data pattern is controlled by DATA PATTERN switches
DEV ADDR DA1 DA2 DA3	DATA CONTROL 4 5 6	Selects station address. DA1 is most significant switch, and DA3 is least significant switch. Station address is selected by the binary setting of the DEV ADDR switches and the UNIT SELECT switch on the station operator control panel
DATA PATTERN PATA PATB PATC PATD	DATA CONTROL 0 1 2 3	Determines data patterns for all station channels. See table 8-2 for switch combinations
PATTERN CTR C0 C1 C2 C3 C4 C5 C6 C7	COUNTER OUTPUT 4 5 6 7 8 9 10 11	Determines data for channels 0-7. If switch setting is ONE or ZERO, sets up data. If switch setting is COUNT, data to channel corresponds to counter output
RECORD LENGTH 2^{11} ↑ ↓ 2^{0}	COUNTER RESET 0 ↑ ↓ 11	Determines record length for a write operation. Switch 2^{0} is the least significant, and switch 2^{11} is the most significant switch. With all switches down, eight bytes are written on tape. Write count can be incremented in multiples of eight bytes

(Continued)

Table 8-3. PET Functions (Cont.)

Overlay Designation	Panel Designation	Function
PET ON-PET OFF/ RESET	SELECTOR	In OFF position, resets controller. In ON position, enables PET operations
RUNNING MODE RP FB	DEBOUNCED 1 2	Controls tape motion as follows: RP ON, FB OFF. Continuous mode. Selected operation is repeated until RP is turned OFF or until tape runs out. Tape does not stop at EOT marker RP OFF, FB ON. Single cycle mode. Selected operation is repeated 16 times; tape spaces backward 16 times; and unit returns to idle state RP ON, FB ON. Continuous cycle mode. Same as single cycle mode except unit repeats operation continuously rather than going into idle state RP OFF, FB OFF. Single order mode. One operation is executed and unit returns to idle state
ST/START	3	Initiates selected operation. Operation is started when switch is released
NORMAL/COUNTER RESET	RESET DEV/INT	Connects device reset signal to 12-stage binary counter. Must always be in UP position
NORMAL CLOCK	COUNTER CLOCK DEV/INT	Allows device clock to increment counter in PET. Must always be in UP position

Table 8-4. Overlay Indicators

Overlay	Pet	Description
DATE	14	Data error. A data parity error may consist of a lateral or a longitudinal parity error in both read and read after write, or of a cyclic redundancy check mismatch error during reading
FF1	9	Major phase determinate. It is used in conjunction with FF2 to determine major phase of controller
FF2	10	Major phase determinate. It is used in conjunction with FF1 to determine major phase of controller. The indications show the major phase the controller is in:

FF1	FF2	Major Phase
0	0	00F
0	1	01F
1	1	03F

(Continued)

Table 8-4. Overlay Indicators (Cont.)

Overlay	PET	Description
FU1, FU2, FU3	11, 12, 13	Subphase or minor phase determinate of controller. The various combinations below are possible: FU1 / FU2 / Minor Phase 0 / 0 / 00U 0 / 1 / 01U 1 / 0 / 02U 1 / 1 / 03U
RATE	15	Rate error. A rate error can occur during a read or a write operation
BFR0-BFR7	0-8	Read data. Used to indicate data pattern being read in each of nine channels. Because of frequency and pulse width, indicators do not turn on. It is necessary to patch into one-shot circuit AUX1 or AUX2 to get an indication of the data pattern

SECTION IX
ILLUSTRATED PARTS BREAKDOWN

9-1 GROUP ASSEMBLY PARTS LIST

The Group Assembly Parts List is a breakdown of all systems, assemblies, and subassemblies which can be disassembled, reassembled or replaced and which are contained in the end article. The Group Assembly Parts List consists of columnar listings of parts related to illustrations. Parts are listed in order of disassembly sequence, except in cases where sequence of disassembly cannot be maintained. Attaching parts are listed below the related assembly or subassembly. Items which are purchased in bulk form (for example, wire and insulating materials) are not listed.

Each parts list table is arranged in seven columns as follows:

a. The figure number of the part listed and the index number corresponding to the illustration reference

b. The SDS manufacturer's part number for the part

c. The vendor's part number for the part (if available)

d. A brief description of the part

e. The manufacturer's code for the part

f. The quantity of the part used per assembly

g. Usable on code column indicating that when a letter is used in the code column. The use of the coded part is restricted to the model identified by the code letter. (Where no letter symbol appears in this column, the part is used on all models of this configuration.)

9-2 NUMERICAL INDEX

This index is a listing of the items contained in the Group Assembly Parts List. The numerical order of the index (table 9-3) is determined by the SDS part number.

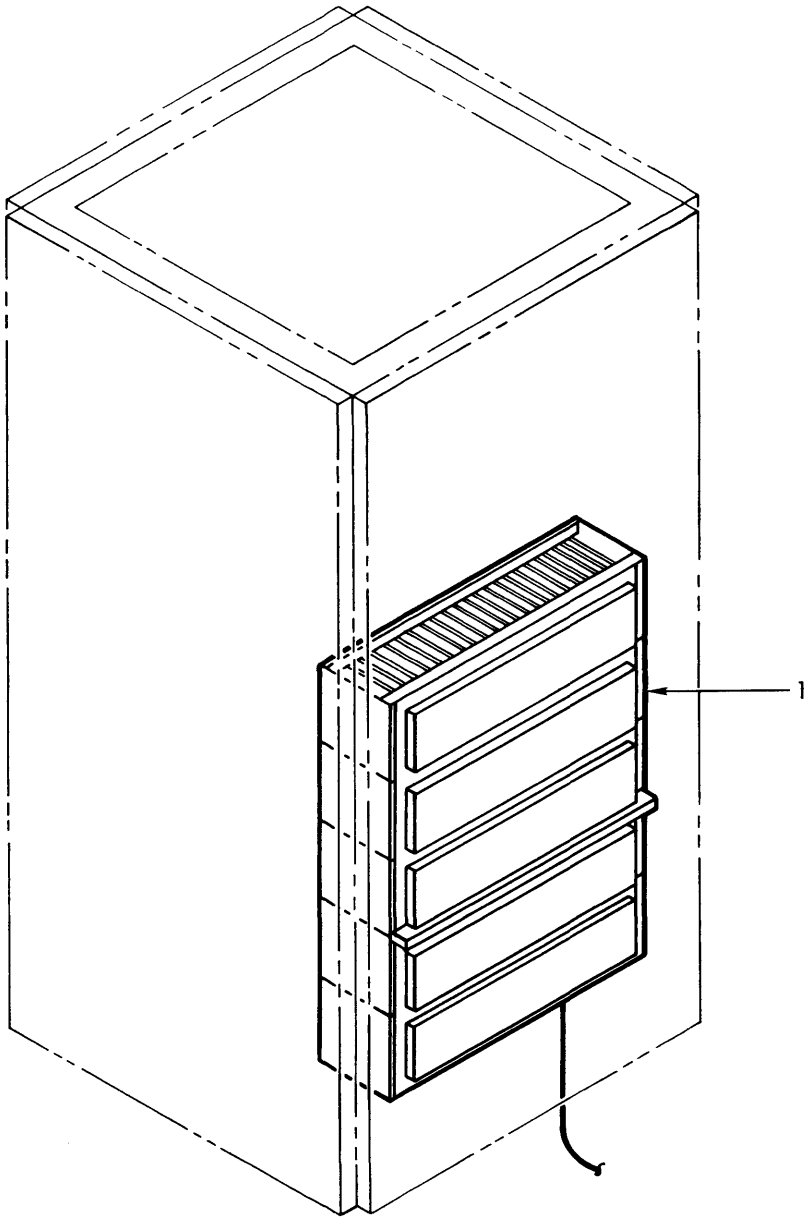


Figure 9-1. Magnetic Tape Controller Assembly

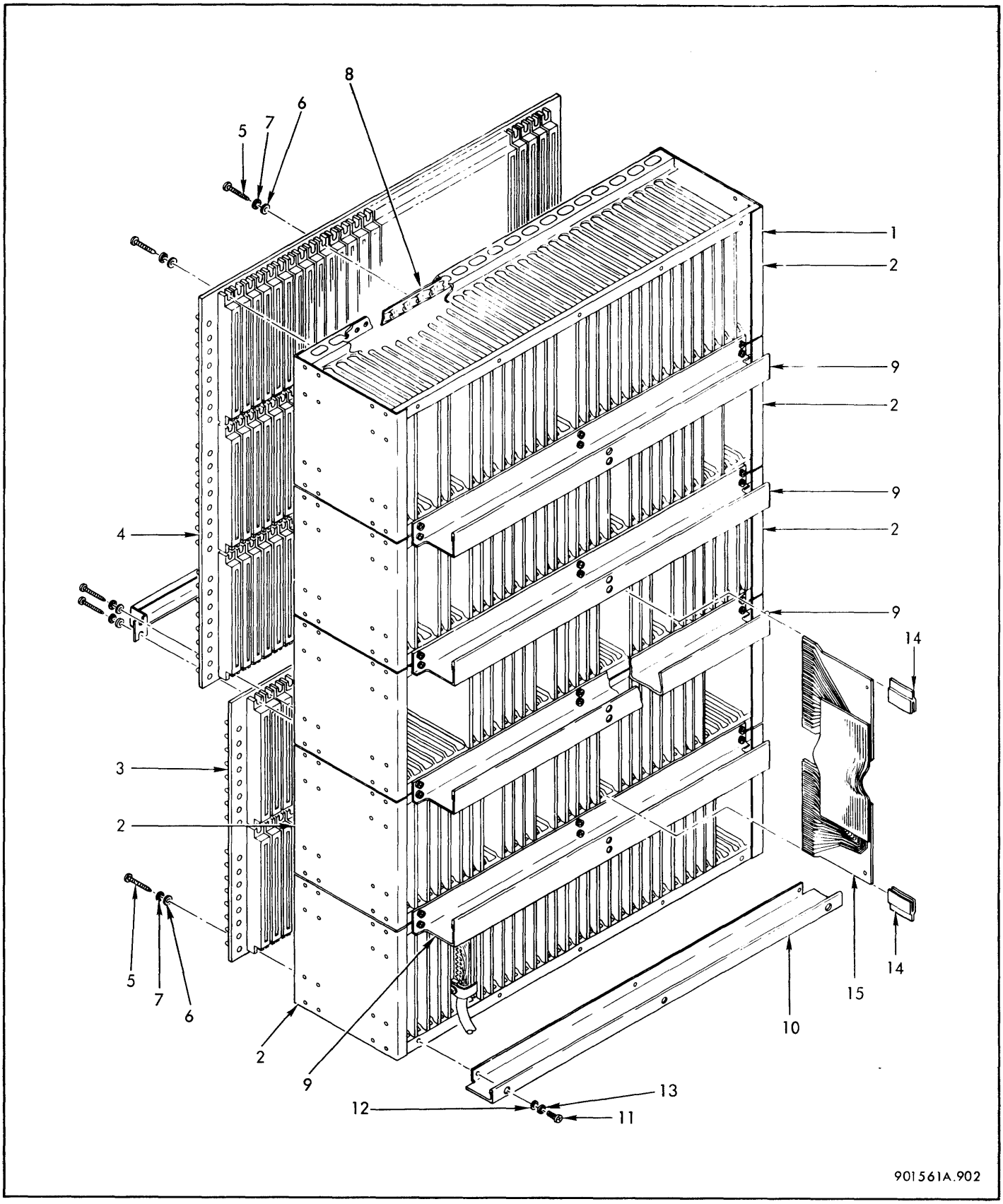


Figure 9-2. Controller Chassis Assembly

Table 9-1. Controller Chassis Assembly

Fig. & Index No.	SDS Part Number	Vendor Part Number	Description 1 2 3 4 5 6 7	Mfg. Code	Units Per Assy	Usable on Code
9-1-	145603		Magnetic Tape Controller Assy			
2-1	145604		. Controller Chassis Assy		1	
-2	116231		. . Chassis, 32 Module		5	
-3	145605		. . Backwiring Board Assy (Y, Z)		1	
-4	145607		. . Backwiring Board Assy (V, W, K)		1	
			(Attaching Parts)			
-5	114538-212		. . Screw, Sheet Metal Pan Head		90	
-6	100018-300		. . Washer, Flat		90	
-7	100024-300		. . Washer, Lock Int. Tooth		90	
-8	129567		. . Nut, Strip Speed		10	
			- - - * - - -			
-9	132197		. . Channel, Cable Routing		4	
-10	123940-001		. . Channel, Cable Routing		1	
			(Attaching Parts)			
-11	100012-204		. . Screw, Pan Head Rec.		30	
-12	100018-200		. . Washer, Flat		30	
-13	100024-200		. . Washer, Lock Int. Tooth		30	
			- - - * - - -			
-14	149850		. . Retainer		1	
-15	137481-171		. . Interframe Cable Assy (ZT45)		1	
-16	127314-221		. . Interconnecting Cable Assy		Ref	

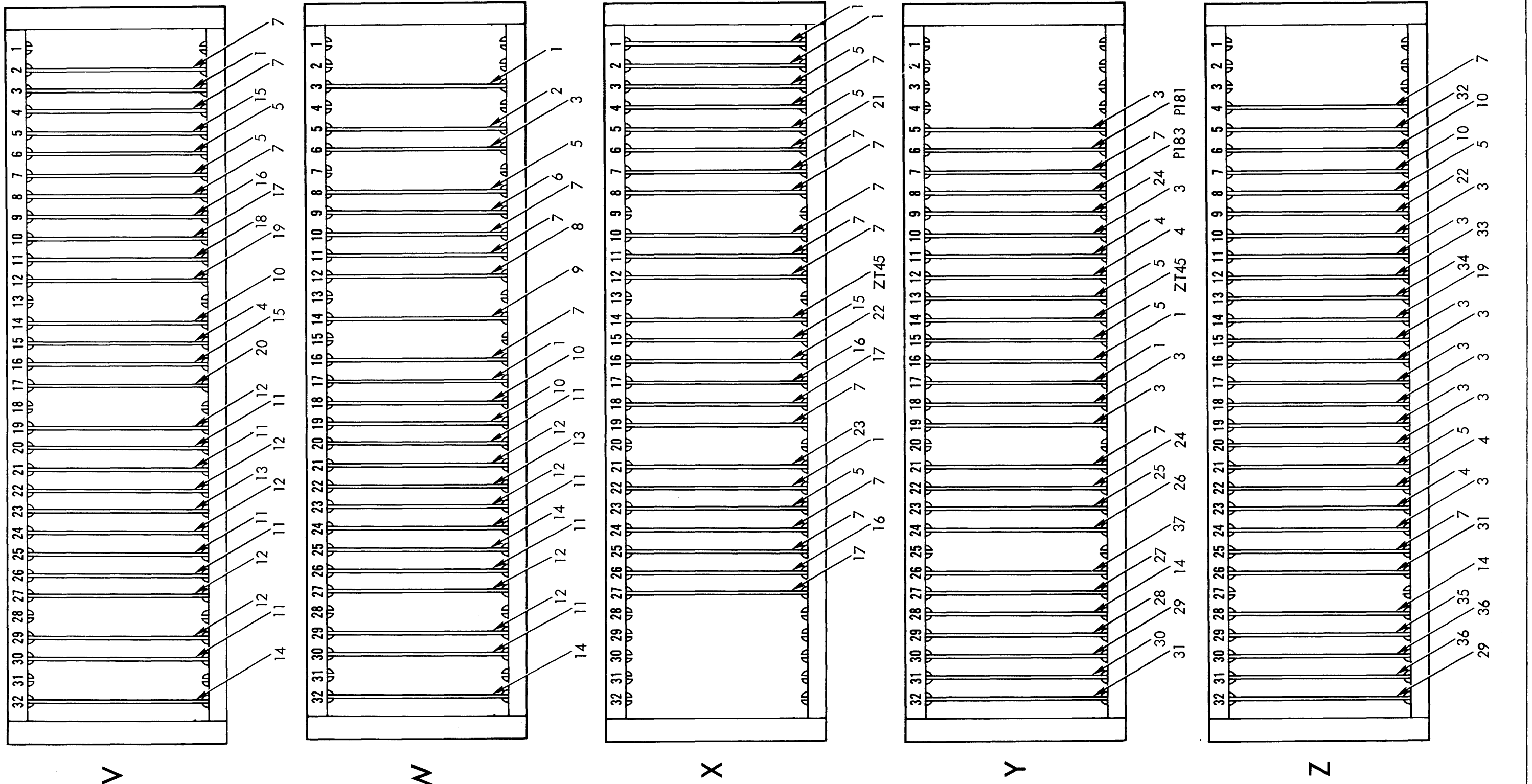


Figure 9-3. Module Locations

901561A.903

Table 9-2. Module Locations

Fig. & Index No.	SDS Part Number	Vendor Part Number	Description 1 2 3 4 5 6 7	Mfg. Code	Units Per Assy	Usable on Code
9-3-			. . (Module Locations)			
3-1	117028		. . Module Assy, FT12, Gated Flip-Flop		8	
-2	128172		. . Module Assy, DT16, Delay Element		1	
-3	131617		. . Module Assy, LT36, Logic Element		14	
-4	116017		. . Module Assy, LT10, Logic Element		5	
-5	116029		. . Module Assy, BT11, Band Gate		10	
-6	117000		. . Module Assy, IT13, Inverter Matrix		1	
-7	116056		. . Module Assy, BT10, Buffered And/Or		20	
-8	117382		. . Module Assy, LT12, Parity Gen.		1	
-9	123491		. . Module Assy, CT10, Clock Osc.		1	
-10	116380		. . Module Assy, FT10, Basic Flip-Flop		5	
-11	130178		. . Module Assy, FT35, Peak Detector		10	
-12	130187		. . Module Assy, FT36, Deskew Register		9	
-13	136547		. . Module Assy, LT71, Exclusive Or		2	
-14	123018		. . Module Assy, AT10, Cable Receiver		5	
-15	126613		. . Module Assy, BT18, Band Gate		3	
-16	130171		. . Module Assy, FT34, CRC Register #2		3	
-17	130118		. . Module Assy, FT33, CRC Register #1		3	
-18	129862		. . Module Assy, AT27, Buffered Cable Dr.		1	
-19	117021		. . Module Assy, FT11, High Speed Count		2	
-20	131072		. . Module Assy, FT39, Fast Access Mem.		1	
-21	116994		. . Module Assy, IT11, Nand Gate		2	
-22	130967		. . Module Assy, BT24, Buffered And/Or		1	
-23	116407		. . Module Assy, BT13, Buffered Matrix		2	
-24	126712		. . Module Assy, LT25, Logic Element		1	
-25	126982		. . Module Assy, LT26, SW Comparator		1	

Table 9-2. Module Locations (Cont.)

Fig. & Index No.	SDS Part Number	Vendor Part Number	Description 1 2 3 4 5 6 7	Mfg. Code	Units Per Assy	Usable on Code
9-3-						
-26	126710		. . Module Assy, LT24, Logic Element		1	
-27	133392		. . Module Assy, LT41, Logic Element		1	
-28	123019		. . Module Assy, AT11, Cable Driver/Rec.		1	
-29	133657		. . Module Assy, LT43, Logic Element		1	
-30	124629		. . Module Assy, AT12, Cable Driver		1	
-31	123016		. . Module Assy, LT13, Buff Inverter #1		1	
-32	116324		. . Module Assy, LT11, Logic Element		1	
-33	126747		. . Module Assy, IT20, Gated Inverter		1	
-34	115965		. . Module Assy, BT12, Decoder		1	
-35	126330		. . Module Assy, BT17, Gated Buffer		2	

Table 9-3. Numerical Index

Fig. & Index No.	SDS Part Number	Vendor Part Number	Description 1 2 3 4 5 6 7	Mfg. Code	Units Per Assy	Usable on Code
2-11	100012-204		Screw, Pan Head Rec.			
2-12	10018-200		Washer, Flat			
2-6	100018-300		Washer, Flat			
2-13	100024-200		Washer, Lock Int. Tooth			
2-7	100024-300		Washer, Lock Int. Tooth			
2-5	114538-212		Screw, Sheet Metal Pan Head			
3-35	115965		Module Assy, BT12, Decoder			
3-4	116017		Module Assy, LT10, Logic Element			
3-5	116029		Module Assy, BT11, Band Gate			
3-7	116056		Module Assy, BT10, Buffered And/Or			
2-2	116231		Chassis, 32 Module			
3-33	116324		Module Assy, LT11, Logic Element			
3-10	116380		Module Assy, FT10, Basic Flip-Flop			
3-24	116407		Module Assy, BT13, Buffered Matrix			
3-22	116994		Module Assy, IT11, Nand Gate			
3-6	117000		Module Assy, IT13, Inverter Matrix			
3-19	117021		Module Assy, FT11, High Speed Count			
3-1	117028		Module Assy, FT12, Gated Flip-Flop			
3-8	117382		Module Assy, LT12, Parity Gen.			
3-32	123016		Module Assy, LT13, Buff Inverter #1			
3-14	123018		Module Assy, AT10, Cable Receiver			
3-29	123019		Module Assy, AT11, Cable Driver/Rec.			
3-9	123491		Module Assy, CT10, Clock Osc.			
2-10	123940-001		Channel, Cable Routing			
3-31	124629		Module Assy, AT12, Cable Driver			
3-36	126330		Module Assy, BT17, Gated Buffer			
3-15	126613		Module Assy, BT18, Band Gate			

Table 9-3. Numerical Index (Cont.)

Fig. & Index No.	SDS Part Number	Vendor Part Number	Description 1 2 3 4 5 6 7	Mfg. Code	Units Per Assy	Usable on Code
3-27	126710		Module Assy, LT24, Logic Element			
3-25	126712		Module Assy, LT25, Logic Element			
3-34	126747		Module Assy, IT20, Gated Inverter			
3-26	126982		Module Assy, LT26, Switch Comparator			
2-16	127314-221		Interconnecting Cable Assembly			
3-2	128172		Module Assy, DT16, Delay Element			
2-8	129567		Nut, Strip Speed			
3-18	129862		Module Assy, AT27, Buffered Cable Drive			
3-17	130118		Module Assy, FT33, CRC Register #1			
3-16	130171		Module Assy, FT34, CRC Register #2			
3-20	130178		Module Assy, FT35, Peak Detector			
3-12	130187		Module Assy, FT36, Deskew Register			
3-23	130967		Module Assy, BT24, Buffered And/Or			
3-21	131072		Module Assy, FT39, Fast Access Memory			
3-3	131617		Module Assy, LT36, Logic Element			
2-9	132197		Channel, Cable Routing			
3-28	133392		Module Assy, LT41, Logic Element			
3-30	133657		Module Assy, LT43, Logic Element			
3-13	136547		Module Assy, LT71, Exclusive OR			
2-15	137481-171		Interframe Cable Assy (ZT45)			
1-	145603		Magnetic Tape Controller Assy			
2-1	145604		Controller Chassis Assy			
2-3	145605		Backwiring Board Assy (Y, Z)			
2-4	145607		Backwiring Board Assy (V, W, X)			
2-14	149850		Retainer			
3-11	153420		Module Assy, FT76, Peak Detector			

READER SURVEY

XDS
Xerox Data Systems

Use this postpaid form to communicate any constructive comments you may have regarding this publication. Please be specific.

PUBLICATION

NUMBER	DATE OF PUBLICATION	TITLE	REV.

HOW DID YOU USE THIS PUBLICATION

☐ LEARNING ☐ INSTALLING ☐ MAINTAINING ☐ OPERATING ☐ SALES

☐ OTHER _____

COMMENTS

YOUR NAME AND RETURN ADDRESS

THANK YOU FOR YOUR INTEREST!

STAPLE

STAPLE

FOLD

FIRST CLASS
PERMIT NO. 229
EL SEGUNDO, CALIF.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

Xerox Data Systems
701 South Aviation Blvd.
El Segundo, California 90245

ATTN: MARKETING PUBLICATIONS

FOLD

CUT ALONG LINE