J. H. HENKE
SAN JOSE #394
360 COMMON I/O
#9999

IBM Field Engineering

Theory of Operation

2803/2804 Model 1 Tape Control and

Tape Controls for 2403/2404 Models 1, 2, and 3

Tape Controls for 2403/2404 Models 1, 2, and 3

Y22-2833-1

IBM Field Engineering

Theory of Operation

Restricted Distribution

This manual is intended for internal use only and may not be used by other than IBM personnel without IBM's written permission.

2803/2804 Model 1 Tape Control and

Tape Controls for 2403/2404 Models 1, 2, and 3

Preface

This manual describes the operation of the tape controls for the IBM 2403/2404 Models 1, 2, and 3 and the IBM 2803/2804 Model 1. The reader's familiarity with IBM component circuits, I/O interface logic, and IBM 2400-series tape units is presumed.

In this manual, the terms "record" and "block" are used interchangeably to denote a continuous series of bytes recorded on magnetic tape.

Additional information on the 2400-series machines is contained in the following manuals:

Field Engineering Theory of Operation, *IBM Magnetic Tape Units: 2401, 2402, 2403 Models 1-6; 2404 Models 1-3*, Form Y22-2819.

Field Engineering Maintenance Manual, *IBM Magnetic Tape Units: 2401, 2402, 2403 Models 1-6; 2404 Models 1-3*, Form Y22-6631.

Field Engineering Maintenance Manual, *IBM Tape Controls: 2403 Models 1-6, 2404 Models 1-3, 2803/2804 Models 1 and 2*, Form Y22-6635.

Systems Reference Library, *IBM 2400 and 2816 Model 1 Component Description*, Form A22-6866.

Field Engineering Manual of Instruction, *IBM SLT Power Supplies*, Form 223-2799.

Field Engineering Diagram Manual, *IBM 2401, 2402, 2403 Models 1-3 Magnetic Tape Units; 2403 Models 1-3 and 2803 Model 1 Tape Controls*, Form Y22-2854.

In this manual, three-digit numbers refer to figures in the Field Engineering Diagram Manual.

FOURTH EDITION

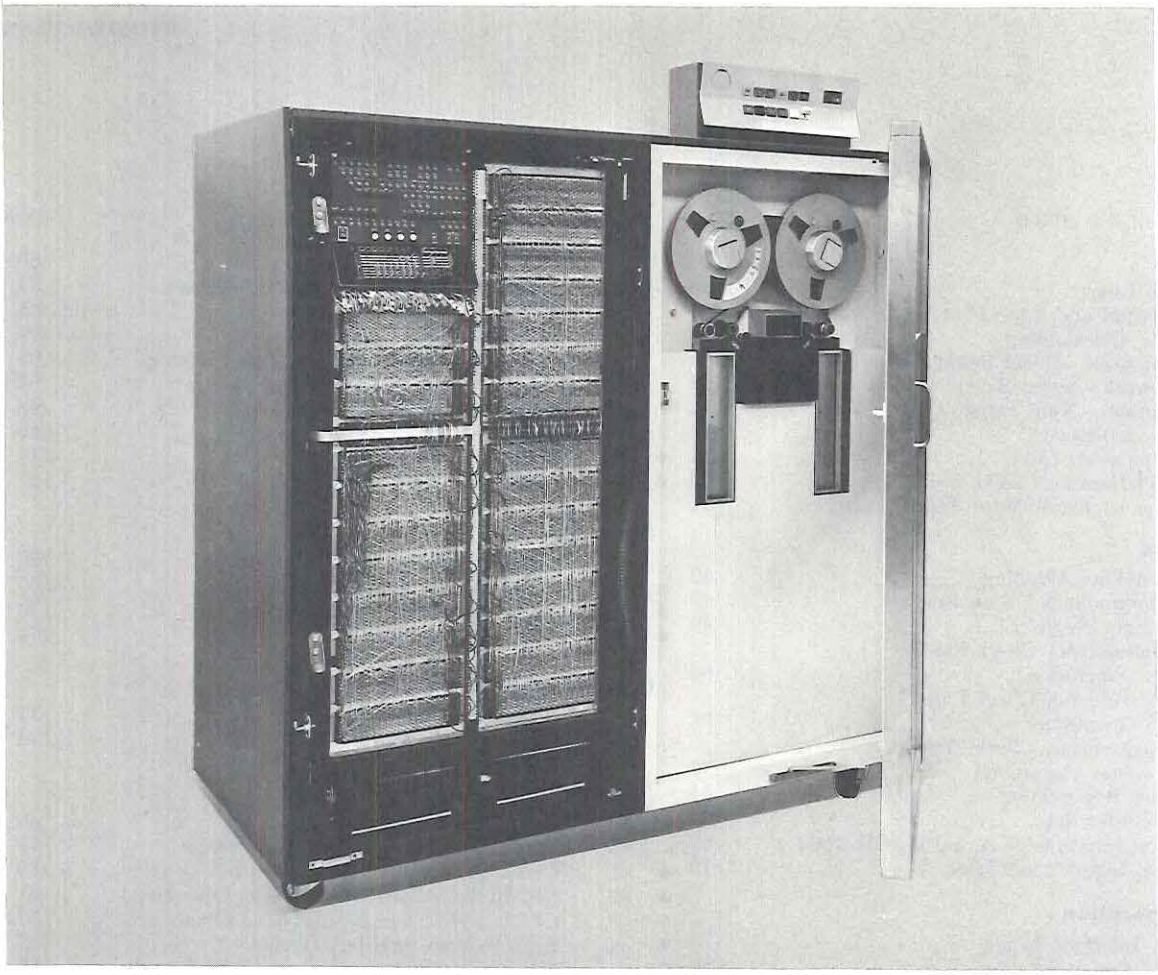
This is a major revision of, and obsoletes, Form Y22-2853-0. Significant additions to this manual are in the area of Cyclic Redundancy Check, Device End Scanner, and CE Panel Indicator Lights. Engineering changes through April 1967 are included.

Changes to the text are indicated by a vertical line to the left of the change; where a complete section has been revised, the symbol • appears to the left of the heading in text. Revised or new illustrations are denoted by the symbol • to the left of the caption.

Specifications contained herein are subject to change from time to time. Any such change will be reported in subsequent revisions or FE Supplements.

This manual has been prepared by the IBM Systems Development Division, Product Publications, Dept. B95, PO Box 390, Poughkeepsie, N. Y. 12602. A form is provided at the back of this publication for the reader's comments. If the form has been removed, comments may be sent to the above address.

Contents



IBM 2403 Magnetic Tape Unit and Control

The tape control units described in this manual are adapter units that connect tape units to IBM System/360 data channels. All data flow and operations of the tape units are controlled and checked by the tape control units.

Purpose of Tape Control Unit

- **Tape control units provide the means of communication between selected tape units and data channels.**
- **Tape control units adapt high-speed, electronic data channel circuits to operate mechanical tape units.**

The basic purpose of a tape control unit is to adapt high-speed, electronic data channel circuits to control a mechanical tape unit. To accomplish this, tape control units are logically divided into high-speed channel interface circuits and low-speed tape unit controlling circuits. Interface circuits include all of the functions needed to communicate with the channel, while the control section provides timed control signals to operate the tape unit.

Data transmitted from the channel to the tape control unit is handled at electronic speed on the channel interface lines (Figure 1). The control unit holds each byte of data until the tape unit is ready for it. When data is transmitted to the channel, each byte from the tape unit is held in the control unit until the channel interface accepts it.

As data is transmitted through the tape control, it is checked for errors. If an error occurs, the interface adapter circuits inform the data channel at the end of the operation.

Packaging

- **Tape control units are packaged either as independent control units or control unit/tape unit combinations.**
- **When a control unit and tape unit are packaged together, the model number indicates the type of tape unit.**

Tape control circuits described in this manual are packaged in one of four types of units:

2803 Model 1 (tape control)
2403 Models 1, 2, and 3 (tape control and tape unit)
2804 Model 1 (simultaneous tape control)
2404 Models 1, 2, and 3 (simultaneous tape control and tape unit)

Circuits in the 2803 Model 1 are identical to circuits in the tape control of the 2403 Models 1, 2, and 3. Circuits in the 2804 Model 1 are identical to circuits in the tape control of the 2404 Models 1, 2, and 3.

Tape units attached to the controls can be single units (called 2401 Models 1, 2, and 3) or double units (called 2402 Models 1, 2, and 3).

Configurations

- **2403 and 2803 tape control units connect up to eight tape units to one data channel.**
- **2404 and 2804 tape control units connect up to eight tape units to two data channels simultaneously.**
- **Features on the tape control units permit operation with both seven-track and nine-track tape units intermixed in the same group.**

The 2403 and 2803 tape control units communicate with one data channel and control operations of one group of tape units (Figures 1 and 2). Electrical power circuits and operating logic circuits in the tape control unit can accommodate up to eight tape units.

The 2404 and 2804 tape control units can communicate with two data channels simultaneously. Either data channel has access to any one of up to eight tape units. All tape units attached to a simultaneous read/write (R/W) control unit must have the simultaneous read/write feature (SIMS) installed.

If the tape control unit is equipped with the seven-track feature, seven-track tape units can be intermixed with standard nine-track tape units. Any combination of tape units, seven-track or nine-track, Models 1, 2, or 3, can be combined, up to a total of eight tape units on each tape control.

When the sixteen address feature (SXT) is installed, the 2403 or 2803 tape control can select up to sixteen tape unit addresses. The SXT feature is used when the tape control is attached to an IBM 2816 Switching Unit.

Physical Description

- **Each tape control unit contains two circuit card gates, labeled A and B.**
- **Simultaneous R/W control units have a third gate, labeled C.**

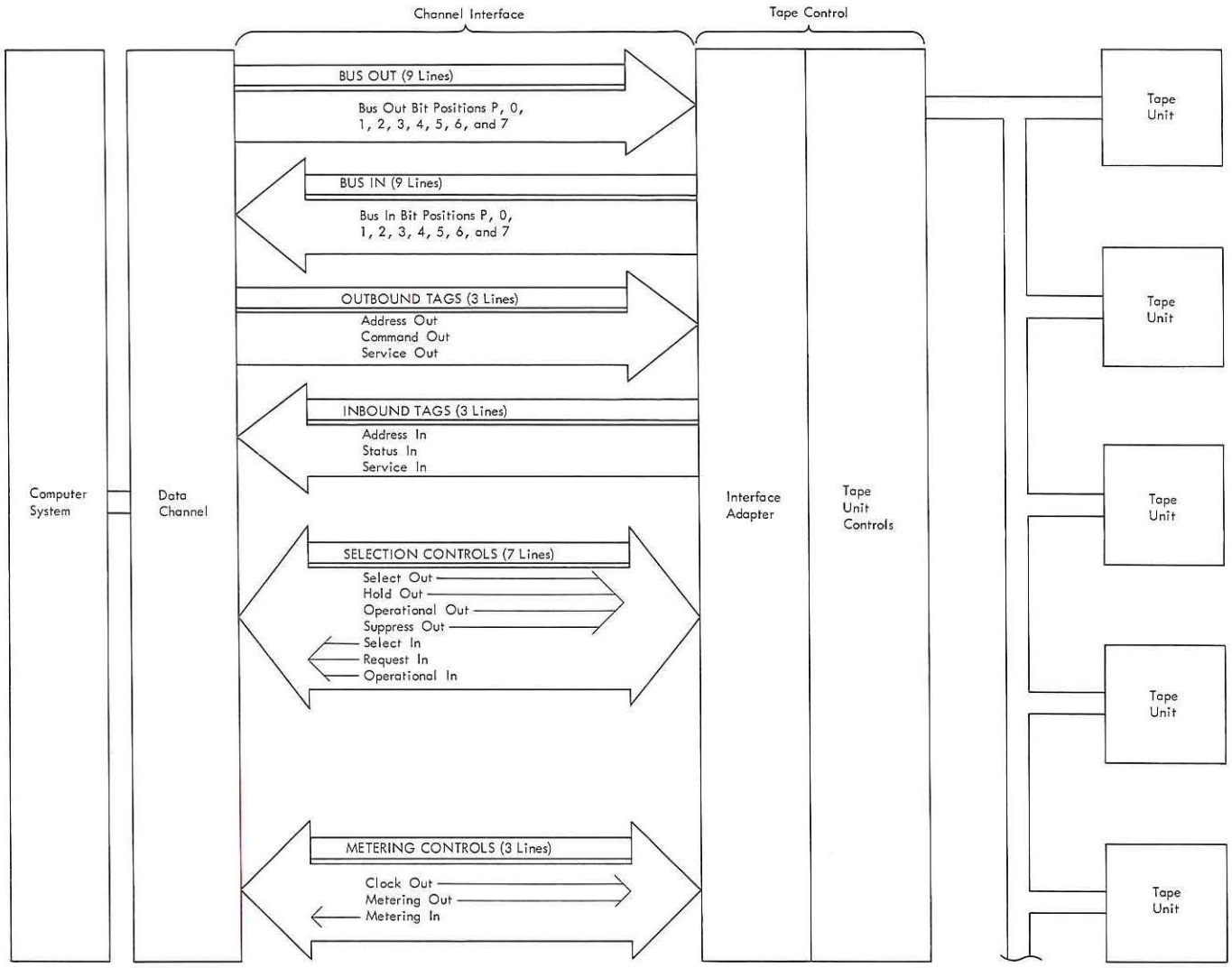


Figure 1. Interface Lines

- **The A gate is divided into five panels and the B and C gates into four panels each.**
- **There are three sets of cable connectors on each tape control unit:**
  1. **Tape unit signal cables.**
  2. **Input cables from the data channel or other control unit.**
  3. **Channel terminators or output cables to the next control unit.**

Figure 3 illustrates the physical placement of gates, panels, and cable connectors on a 2403 tape control unit; the same arrangement is used in the 2803 tape control units but without the tape unit and its A/B connectors.

Panel Layout

Two gates of circuit cards are included in each 2403 and 2803 tape control unit. Gate A is on hinges to swing out for access to gate B. As shown in Figure 3, the A gate is divided into five panels, one of which is the CE panel. The B gate is divided into four panels.

Simultaneous R/W control units contain the A and B gates, plus a third swinging gate labeled C. The C gate contains four panels of circuit cards and is located behind the A and B gates.

Circuit cards within each panel are numbered 1 to 20 from left to right and rows A to H from top to bottom. Circuit card numbering is from the card side of the panel.

Cable Connections

Control unit signal cable connectors are located on a small panel next to panel 3 of the B (fixed) gate. These connectors are designated A, B, C, and D in the top group and 1 to 8 in the bottom group. As shown in Figure 3, the 2403/2803 control units do not use the

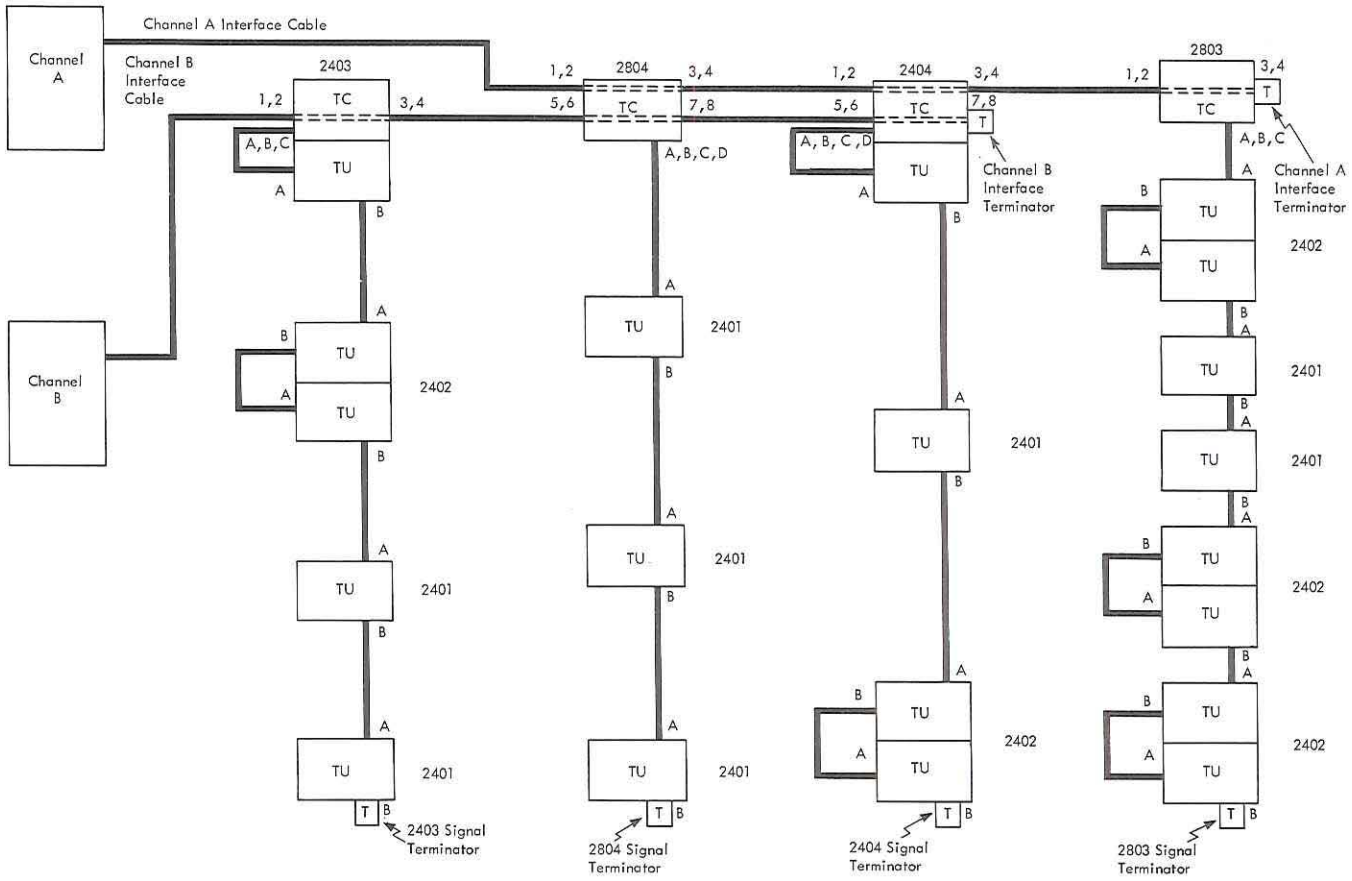


Figure 2. Tape Control and Tape Unit Signal Cabling

D or 5 to 8 positions; these positions are used on simultaneous R/W control units.

The A, B, and C connectors are cabled to the first tape unit of the attached group. On simultaneous R/W control units, the D connector is also cabled to the first tape unit.

Connectors 1 to 8 are the channel interface connectors—two in and two out for each channel. A 2403/2803 uses only four interface connectors (1 to 4), as shown in Figure 3. A simultaneous R/W control unit uses all eight connectors because it communicates with two data channels.

Connectors 1, 2, 5, and 6 are cabled to either the data channel or the previous control unit of a group. Connectors 3, 4, 7, and 8 are cabled to the next control unit of the group. If this control unit is the last of a group, connectors 3, 4, 7, and 8 contain terminators.

Data Format

The process of recording bits on magnetic tape is described in Field Engineering Theory of Operation, *IBM Magnetic Tape Units: 2401, 2402, 2403 Models 1-6; 2404 Models 1-3*, Form Y22-2819. The following discussion deals with the organization of bits into meaningful patterns.

Bytes

- **Each bit position can record only 1 or 0.**
- **Eight bits form a byte and represent one number, letter or symbol.**
- **Eight bits of a byte plus a parity bit are written in nine tracks on tape.**
- **Groups of bytes are called records or blocks.**

Any bit position (change in flux on tape) can record only two conditions: 1 or 0. Groups of bits can be coded to represent numbers, letters or symbols (Figure 4). IBM System/360 employs a standard grouping of eight bits (a byte) plus a parity bit (P bit) for error detection.

Skew

When a byte is recorded on tape, each bit position is written on a separate track of the tape. Each byte is written at right angle to the length of the tape.

Deviation from the ideal right angle line is called write skew. Tape units are designed with variable delay circuits to compensate for small variations in write head construction that could cause skew. The end result is that bits in a byte are written as nearly in a straight line as possible.

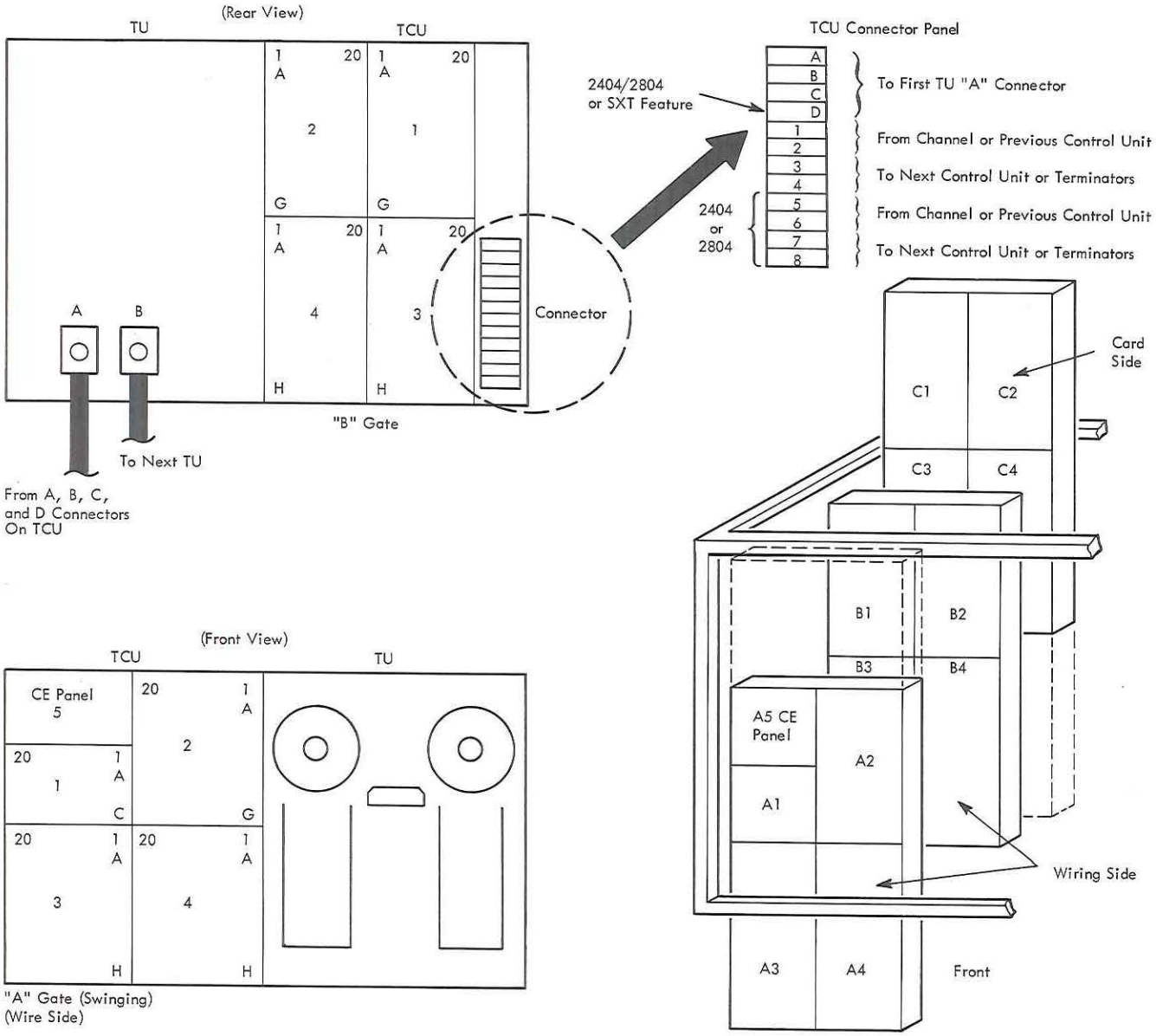


Figure 3. Locations and Cabling

When a byte is read from tape , all the bits may not be sensed at the same time. The time interval between the first and last bits of one byte is called read skew. Read skew can be caused by write skew in the byte on tape or by variation between tracks in the reading operation. To compensate for minor variations in read head construction, the read circuits have variable delays.

Block or Record Density

Groups of bytes along tape are called records or blocks. The computer program specifies the number of bytes to be written in a block.

Byte density, written on a tape, is determined by the speed of the tape and the speed at which the control unit sends bytes to be written. In nine-track operation, 800 bytes can be written on one inch of tape. Each of the three models of tape units operates at a different speed, so the tape control must be able to supply data at three different frequencies.

For seven-track operation, the control unit must be able to supply data for three densities: 200, 556, and 800 bytes per inch. To use all three speeds of tape units, the control unit must be able to write bytes at nine operating frequencies.

The physical spacing between bytes (or adjacent bits) in a track is:

800 BPI	0.00125 inch
556 BPI	0.0018 inch
200 BPI	0.005 inch

Collating Sequence	Graphics		Eight – Bit Code								BCD					
	8 Bit	BCD	0	1	2	3	4	5	6	7	B	A	8	4	2	1
00	blank	blank	0	1	0	0	0	0	0	0	0	0	0	0	0	0
01	.	.	0	1	0	0	1	0	1	1	1	1	1	0	1	1
02	←	¤)	0	1	0	0	1	1	0	0	1	1	1	1	0	0
03	(	[	0	1	0	0	1	1	0	1	1	1	1	1	0	1
04	+	<	0	1	0	0	1	1	1	0	1	1	1	1	1	0
05	GM	GM	0	1	0	0	1	1	1	1	1	1	1	1	1	1
06	&	& +	0	1	0	1	0	0	0	0	1	1	0	0	0	0
07	$	$	0	1	0	1	1	0	1	1	1	0	1	0	1	1
08	*	*	0	1	0	1	1	1	0	0	1	0	1	1	0	0
09	)	]	0	1	0	1	1	1	0	1	1	0	1	1	0	1
10	;	;	0	1	0	1	1	1	1	0	1	0	1	1	1	0
11	MC	MC	0	1	0	1	1	1	1	1	1	0	1	1	1	1
12	-	-	0	1	1	0	0	0	0	0	1	0	0	0	0	0
13	/	/	0	1	1	0	0	0	0	1	0	1	0	0	0	1
14	,	,	0	1	1	0	1	0	1	1	0	1	1	0	1	1
15	%	% (	0	1	1	0	1	1	0	0	0	1	1	1	0	0
16	WS	WS	0	1	1	0	1	1	0	1	0	1	1	1	0	1
17	↓	\	0	1	1	0	1	1	1	0	0	1	1	1	1	0
18	SM	SM	0	1	1	0	1	1	1	1	0	1	1	1	1	1
19	↑	↑	0	1	1	1	1	0	1	0	0	1	0	0	0	0
20	#	# =	0	1	1	1	1	0	1	1	0	0	1	0	1	1
21	@	@ '	0	1	1	1	1	1	0	0	0	0	1	1	0	0
22	▽	:	0	1	1	1	1	1	0	1	0	0	1	1	0	1
23	=	>	0	1	1	1	1	1	1	0	0	0	1	1	1	0
24	TM	TM	0	1	1	1	1	1	1	1	0	0	1	1	1	1
25	ẟ	ẟ	1	1	0	0	0	0	0	0	1	1	1	0	1	0
26	A	A	1	1	0	0	0	0	0	1	1	1	0	0	0	1
27	B	B	1	1	0	0	0	0	1	0	1	1	0	0	1	0
28	C	C	1	1	0	0	0	0	1	1	1	1	0	0	1	1
29	D	D	1	1	0	0	0	1	0	0	1	1	0	1	0	0
30	E	E	1	1	0	0	0	1	0	1	1	1	0	1	0	1
31	F	F	1	1	0	0	0	1	1	0	1	1	0	1	1	0
32	G	G	1	1	0	0	0	1	1	1	1	1	0	1	1	1
33	H	H	1	1	0	0	1	0	0	0	1	1	1	0	0	0
34	I	I	1	1	0	0	1	0	0	1	1	1	1	0	0	1
35	ō	ō	1	1	0	1	0	0	0	0	1	0	1	0	1	0
36	J	J	1	1	0	1	0	0	0	1	1	0	0	0	0	1
37	K	K	1	1	0	1	0	0	1	0	1	0	0	0	1	0
38	L	L	1	1	0	1	0	0	1	1	1	0	0	0	1	1
39	M	M	1	1	0	1	0	1	0	0	1	0	0	1	0	0
40	N	N	1	1	0	1	0	1	0	1	1	0	0	1	0	1
41	O	O	1	1	0	1	0	1	1	0	1	0	0	1	1	0
42	P	P	1	1	0	1	0	1	1	1	1	0	0	1	1	1
43	Q	Q	1	1	0	1	1	0	0	0	1	0	1	0	0	0
44	R	R	1	1	0	1	1	0	0	1	1	0	1	0	0	1
45	RM	RM	1	1	1	0	0	0	0	0	0	1	1	0	1	0
46	S	S	1	1	1	0	0	0	1	0	0	1	0	0	1	0
47	T	T	1	1	1	0	0	0	1	1	0	1	0	0	1	1
48	U	U	1	1	1	0	0	1	0	0	0	1	0	1	0	0
49	V	V	1	1	1	0	0	1	0	1	0	1	0	1	0	1
50	W	W	1	1	1	0	0	1	1	0	0	1	0	1	1	0
51	X	X	1	1	1	0	0	1	1	1	0	1	0	1	1	1
52	Y	Y	1	1	1	0	1	0	0	0	0	1	1	0	0	0
53	Z	Z	1	1	1	0	1	0	0	1	0	1	1	0	0	1
54	0	0	1	1	1	1	0	0	0	0	0	0	1	0	1	0
55	1	1	1	1	1	1	0	0	0	1	0	0	0	0	0	1
56	2	2	1	1	1	1	0	0	1	0	0	0	0	0	1	0
57	3	3	1	1	1	1	0	0	1	1	0	0	0	0	1	1
58	4	4	1	1	1	1	0	1	0	0	0	0	0	1	0	0
59	5	5	1	1	1	1	0	1	0	1	0	0	0	1	0	1
60	6	6	1	1	1	1	0	1	1	0	0	0	0	1	1	0
61	7	7	1	1	1	1	0	1	1	1	0	0	0	1	1	1
62	8	8	1	1	1	1	1	0	0	0	0	0	1	0	0	0
63	9	9	1	1	1	1	1	0	0	1	0	0	1	0	0	1

Figure 4. Eight Bit Code – BCD Relations

Seven/Nine-Track Check Characters

- **The standard nine-track format contains data followed by *two* check characters, a CRCC and an LRCC (Figure 6).**
- **The seven-track format contains data followed by *one* check character, an LRCC (Figure 5).**

The standard nine-track format (Figure 6) writes two check characters at the end of each block. The first check character is the cyclic redundancy check character (CRCC) and the second check character is a longitudinal redundancy check character (LRCC).

The CRCC is a character developed in the CRC register in the tape control. This character, during each nine-track write operation, represents an accumulation of all the bits in the block. Four byte spaces after the last data byte, the CRCC is written on tape.

The second check character (LRCC) is an odd/even parity count of all the bits in each track of a block. The total number of bits in any track of a block is made an even number by placing a 1 or 0 in the LRCC position. The second check character is written four spaces after the CRCC, or a total of eight spaces from the last data byte. The nine-track LRCC represents the same odd-even bit count as the LRCC used in seven-track operation. Since the CRCC is written before the LRCC, the CRCC bits are included in the odd/even LRCC count. Each track must have an even number of bits in each block. For seven-track operation only one check character (the LRCC) is written.

The LRCC in seven-track operation is written four character spaces after the last data character. The seven-track format meets the same specifications as the format used on other IBM tape systems, such as the IBM 729 Magnetic Tape Units.

Seven/Nine-Track Tape Mark

Groups of blocks can be separated by a "tape mark." A tape mark is a one byte block with a single check character (LRCC). The bit configurartion of a nine-track tape mark is bits 3, 6, and 7 in both the single byte block and the LRC character. A seven-track tape mark consists of bits 1, 2, 4, and 8 in both the single byte block and the LRC character. Both seven- and nine-track tape marks are preceded by 3¾ inches of blank tape (a load point skip). The computer program can initiate a write tape mark (WTM) command to write a tape mark. A tape mark detection circuit is built into the tape control unit to indicate when a tape mark is recognized during read operations.

Data Byte Parity

The parity bit, or P bit, of each byte is used to detect parity errors. The P position of a byte can contain a 1 or a 0, depending on the number of bits in the rest of the byte. A standard nine-track tape control uses odd parity; thus, the total number of bits in a byte, including the P bit, should be odd. If the data portion (bits 0-7) of a byte contains an odd number of bits, the P position should contain 0. If the data portion contains an even number of bits, the P position will be on, to make the total odd. All correct bytes have an odd number of bits. If a byte with an even number of bits is detected, an error has occurred.

With features installed, a System/360 tape control can process seven-track information compatible with other IBM tape systems. Seven-track information can have either odd or even parity. Parity checking circuits in the tape control can check for odd or even parity in data transmitted to or from seven-track tape units.

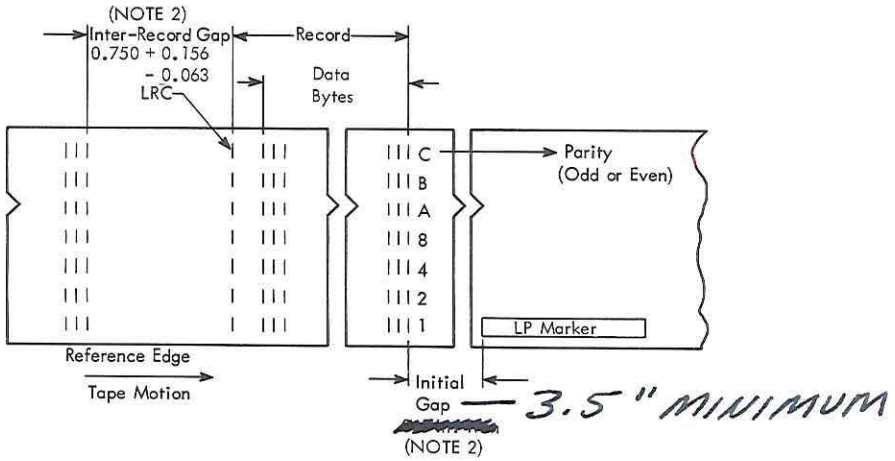
All information transmitted by the channel interface must still have odd parity because System/360 data channels cannot accept even parity bytes. When handling seven-track data, the tape control must adjust the P bit position accordingly.

Check Character Parity

- **Nine-track LRC character always odd parity.**
- **Seven-track LRC character and nine-track CRC character can be odd or even parity.**
- **CRC character will contain an *odd* number of bits *if* there are an *even* number of data characters.**
- **CRC character will contain an *even* number of bits *if* there are an *odd* number of data characters.**

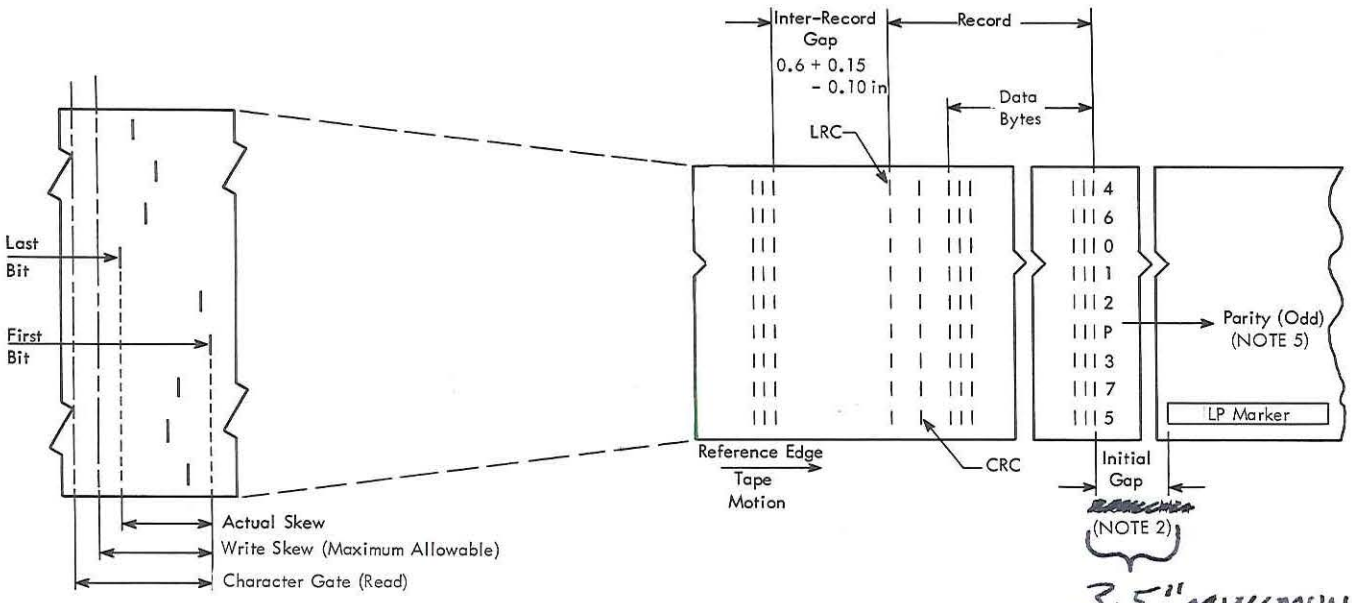
During nine-track operations, the LRCC always contains an odd number of bits. Thus, the vertical redundancy of a nine-track LRCC is always odd. However, the vertical redundancy (number of bits in the character) of the nine-track CRCC can be an odd or even number. As a general rule, the vertical redundancy of the CRCC will be an even number if the total number of data bytes in the entire block is an odd number. Likewise, if the total amount of data bytes in the block is an even number, then the vertical redundancy of the CRCC will be odd.

The error correction circuits generate check character parity and will be discussed in the Error Correction section. A nine-track LRCC will always contain an odd number of bits; the CRCC may be odd or even, depending upon the number of data bytes within the block. The seven-track data format may contain either an odd or even number of bits in the LRC character. This is possible because seven-track operations may employ odd or even redundancy when handling data bytes. The following considerations apply to the redundancy of the seven-track LRCC:



NOTES:
1. Tape is shown with oxide side down. NRZI recording. Bit produced by reversal of flux polarity. Tape fully saturated in each direction.
2. Tape to be fully saturated in the erased direction in the initial gap and the inter-record gap. Erasure such that an N seeking end of compass will point to start of tape.
3. LRC – Longitudinal redundancy check character – odd or even-spaced four bits from data character.
4. Parity Bit – A vertical parity bit is written for each character.
5. Must conform to all 729 specifications.

Figure 5. Data Format – Seven Track



NOTES:
1. Tape is shown with oxide side down. NRZI recording. Bit produced by reversal of flux polarity. Tape fully saturated in each direction.
2. Tape to be fully saturated in the erased direction in the initial gap and the inter-record gap. Erasure such that an N seeking end of compass will point to start of tape.
3. CRC - Cyclic redundancy check character. Parity of CRC character is determined by the number of data characters in record. Odd number of data characters-even CRC character, etc. CRC used only in System/360 800 bpi. CRC character spaced four bits from data characters.
4. LRC - Longitudinal redundancy check character - always odd parity. Spaced four bits from CRC.
5. Parity Bit - A vertical parity bit is written for each character containing an even number of bits.

3.5" MINIMUN INITIAL GAP.

● Figure 6. Data Format — Nine Track

1. Using even redundancy mode will always produce an LRCC containing an even number of bits regardless of the number of data bytes in the block.

2. When using odd redundancy mode, the number of bits in the LRCC will be odd if there is an odd number of data bytes, or the number of bits in the LRCC will be even if there is an even number of data bytes in the block.

Inter Block Gaps (IBG)

- **The gap between blocks is generated by tape unit mechanical inertia and tape control unit delay timings.**
- **Groups of blocks can be separated by a tape mark.**

The gap or blank space between blocks on tape is caused by mechanical limitations in the tape unit. Tape cannot start or stop instantaneously. A definite period of time is required for tape to accelerate to full speed or come to a stop.

When the end of a block is reached, the tape unit is told to stop tape movement. Inertia will move the tape past the end of the block before motion stops. The control unit must make allowance for tape unit limitations by appropriate delay timings.

A command to write another block will start tape motion again. Bits cannot be written properly until the tape is at full speed, so the tape control unit delays transmission of bytes to the tape unit. Waiting time while the tape accelerates is called write delay (or read delay for read type operations).

A gap between blocks is the total length of tape movement during stop time of the first block and the start delay time of the following block. Start and stop times (mechanical), which can be adjusted in the tape unit, have a direct bearing on the length of the gap. The tape control unit has a specific delay timing for each model of tape unit. If the tape unit mechanical start and stop times are changed, the length of the gap is also changed.

Short stop times tend to create shorter gaps. Quick acceleration tends to create longer gaps. When a tape unit gets up to full speed quicker, tape travels a greater distance during write delay time. The normal gap between blocks is 0.6 inch long for nine-track and 0.75 inch long for seven-track. The blocks can be closer in nine-track because nine-track tape unit read/write heads are constructed with a closer spacing between the read head gap and the write head gap.

Seven/Nine Track Data Handling

If the seven-track feature is installed, the control unit can operate in either seven- or nine-track mode, depending on the selected tape unit. Seven- and nine-track tape units Models 1, 2, or 3 can be intermixed on the same line. Following is a brief description of each method of data handling.

The standard control unit handles nine-track data in odd redundancy, employing a density of 800 BPI. In addition, to be compatible to former systems, this control unit may also process seven-track information. When operating with seven-track tape units, the redundancy can be odd or even, while the density may be any one of three densities – 200, 556, or 800 BPI.

Nine-Track

- **When transferring to and from channel, the data bytes will always contain eight data bits plus parity bit.**
- **During nine-track operation, eight-bit bytes are transmitted back and forth between channel and the selected tape unit without change.**

During nine-track operations, the tape control unit will transfer nine bits of information per byte; this includes eight data bits and one parity bit. During a write operation, the channel supplies the control unit with each data byte to be written on the tape unit. During a read operation, the tape unit supplies the control unit with each data byte to be transferred to the channel via the nine "bus in" lines of the channel interface. Thus, when operating with nine-track tape units, each data byte progresses through the control unit unchanged enroute to either the channel or the selected tape unit.

Seven-Track

- **During nine-track operation, eight-bit bytes are transmitted back and forth between channel and the selected tape unit without change.**
- **Translator on mode—eight-bit coding is translated into the equivalent six-bit BCD coding, and vice versa.**
- **Translator off mode—data bytes are transmitted through the tape control without change.**

To provide compatibility with former tape systems, there are two seven-track features which can be installed on the standard 2403/2803 tape control unit. The first seven-track feature (Seven-Track Compatibility) operates in one of two modes – translator on or translator off. The second feature (Data Conversion) can be installed only of the compatibility feature is already installed.

If both features are installed, the control unit must have some method of indicating which mode a par-

ticular seven-track operation must assume. Within the control unit there is a mode register that specifies how to handle seven-track data in both read and write operations. This register (Figure 18) indicates odd or even redundancy, 200 or 556 or 800 BPI, translator on or off, and data conversion on or off.

Setting of the mode register is done by a command called "mode set." Contents of the mode register are applicable only to a seven-track tape unit. Following is a brief description of each seven-track mode of data handling.

Translator Off

On seven-track write operation if the mode register specifies translate off mode, the control unit discards the two high-order positions of the data byte from channel. Only the six low-order positions, plus parity, are transferred to the tape unit to be written. On seven-track read operation, in translator off mode, the control unit inserts two zeros in the two high-order positions of the data byte when transferring to the channel. "Bus out" positions 0 and 1 are discarded on write operations, while "bus in" positions 0 and 1 have zeros inserted in the read operation. The density can be either 200, 556, or 800 BPI and the redundancy of the data bytes to or from the tape unit can be either odd or even.

Translator On

Former tape systems employed the six-bit BCD coding which must now be translated to an equivalent eight-bit coding. The total number of bit combinations used is sixty four. A special unit in the tape control performs the actual decoding and produces equivalent results for both eight-bit and six-bit characters. This unit is the translator, which will be active only if the mode register settings indicate "translator on."

If the mode register is specifying "translator on" mode when reading or writing seven-track data, then each data byte (from channel or from the tape unit) is translated into a different bit configuration before it leaves the control unit.

Figure 4 shows the translation of:

1. a six-bit coding to an equivalent eight-bit coding during a read operation, or,
2. an eight-bit coding to an equivalent six-bit coding during a write operation.

Data Conversion

- **Data conversion is for seven-track forward operation only.**
- **During write operation, convert 3 eight-bit bytes from the channel interface into 4 six-bit characters for the tape unit.**
- **During read operation, convert 4 six-bit characters from the tape unit to 3 eight-bit bytes for the channel interface.**
- **Refer to Figures 309 and 310.**

Data conversion is a means of converting seven-track binary information (six-bit units) into eight-bit bytes and vice versa. Conversion does not change the sequence of bits (as translation does). Conversion merely splits the information into groups of different size. During a write convert operation, 3 eight-bit bytes (24 bits) will be broken down into 4 six-bit characters (24 bits). During a read convert operation, 4 six-bit characters (24 bits) from the tape are assembled into 3 eight-bit bytes (24 bits).

If mode register 2 is on, data conversion is turned off and each six-bit character read from the tape unit is transmitted to the channel in an eight-bit byte in either translate on or translate off mode.

In write operation, without the converter, each eight-bit byte from the channel provides one six-bit character to be written on the tape either in translate on or translate off mode.

Conversion allows a much greater amount of data to be put into a given amount of storage space. In write operation, if the byte count is not a multiple of three, any remaining bits of the last six-bit character are zero. When reading, if the character count of the record is not a multiple of four, any remaining positions in the last byte that has bits will be padded with zeros. When the last data byte of a read operation is padded, data convert check is turned on by a sample pulse. A complete description of the data conversion feature is provided in the Features section.

Tape Control Unit Organization

- **Data flow for all write operations is from the channel to bus out, to the data register, through the translator, to the read/write register, and to the tape unit write triggers via the write bus. Read checking during write operations is from the tape unit read circuit via the read bus, to the final amplifiers in the tape control, to skew registers, to the LRC register.**
- **General data flow for all read operations is from the tape unit preamplifiers to the read bus, to the tape control final amplifiers, to the skew registers, to the read/write register, through the translator, to the data register, and to the channel via the bus in lines.**

Figure 100 is an overall view of tape control and tape unit organization. At the left side of the page is the channel I/O interface. The tape control circuits are in the center and the tape unit is on the right.

Information to be written on tape passes from left to right through the interface bus out lines, to the tape

control unit circuits and to the tape unit. Information read from the tape passes from right to left from the tape unit through the tape control to the interface bus in lines.

The designation "in" and "out" on interface lines is from the channel point of view. "Out" lines transmit data and controls *out* of the channel to the control units. "In" lines transmit data and controls *into* the channel from the control units.

In the tape control, registers are temporary storage places for data or commands. Data can also be checked or modified while in a register. Most of the registers in the tape control unit consists of a group of nine trigger positions which correspond to a byte structure of eight bits plus parity. The data register contains 12 positions. The extra positions are used only during seven-track data convert operations.

Some of the tape control unit registers do not handle data. The cyclic redundancy check register and the error pattern register are used for error detection and correction. The longitudinal redundancy check register is used for error detection.

Addresses and commands to control tape operations are transmitted on the same bus out lines used for write data. Status and sense information is sent to channel on the same bus in lines used for read data. Tag lines are brought up to identify the bytes on the bus lines.

Bits in the command register are decoded to generate control lines. Tape control and tape unit actions are controlled by outputs of the command decoder.

Write clock and read clock circuits provide timing pulses to handle data in the control unit. Data shifting from register to register in the control unit is regulated by the clock timings, the read/write control, and the shift pulse circuits.

Write Data Flow

Actual data handling during a write operation is done by the data register, read/write register, skew registers, and final amplifiers. (To analyze this data flow, follow the heavy black lines in Figure 100, progressing left to right.) During write operation, bus out bytes are gated into the data register. From the data register the bytes go through the translator enroute to the read/write register to the write bus. The tape unit accepts the write bus data and records it on the tape. The density of characters written on the tape depends on how fast the tape is moved and the frequency at which the tape control transmits characters. Once the data bytes are written on tape, each is read back to the control unit to be checked.

During read checking of a write operation, the characters are read from the tape, then amplified by the read preamplifiers and sent to the tape control via the read bus lines, as indicated by the heavy lines progressing from right to left on Figure 100. In the tape control the final amplifiers produce high and low clip outputs according to the amplitude of the read bus bits.

High clip amplifier signals are sent to the high clip skew register, while low clip signals are sent to the low clip skew register. The function of the skew registers is to eliminate minor time differences between bits read from tape. The skew registers accept the bits and store them until the remaining bits of the byte arrive. Outputs of each skew register are gated out in parallel. From the skew register(s), the data is now transferred only to the LRC register for checking purposes. The data being checked cannot be allowed to enter the read/write register because there would be a conflict of data bytes being written and those in the process of being read checked.

Read Data Flow

The actual data flow path used during the read operation is similar to the read checking path of the write operation. As shown by the heavy black lines on Figure 100, the data characters are detected and amplified by the read preamplifiers in the tape unit. The data byte is then sent to the control unit via the read bus lines. In the control unit, the final amplifiers produce high and low clip outputs, depending on the amplitude of the bits on the read bus lines. Again, the high clip amplifier signals feed the high clip skew register, while the low clip amplifier signals feed the low clip skew register. Bytes from either skew register are transmitted to the read/write register and through the translator enroute to the data register to the bus in lines. As data leaves the tape control on the bus in lines, the correct parity is inserted in the P position of each byte. All bytes transmitted on the channel interface must have odd parity including data read from tape as seven-track even parity.

Data Flow Checks

Tape control units have the ability to detect parity errors while transmitting characters. Data passing through the tape control unit can be parity checked at three points.

1. Each byte received from the channel is checked for odd parity on the bus out lines.
2. Characters in the high clip skew register are checked for correct parity (odd or even).
3. The read/write register is parity checked but the output of the checking circuit is used differently in read and write operations. During read operation the read/write parity check circuit is used for error detection. In write operation the parity check circuit is used to generate the correct parity for the character written on tape. The read/write parity check circuit controls

the write bus P bit line in write operation. Each character on the write bus is made to conform to the specified parity by inserting or deleting the P or C bit.

In addition to detecting errors, the tape control units can correct most single track errors. The cyclic redundancy check register (CRCR) and the error pattern register (EPR) are used to determine which track of the tape contains the error and caused the failure.

Channel I/O Interface

The I/O interface is a group of lines connecting I/O control units to a channel attached to a computer system. The interface is not a unit and performs no logical function. Interface is a definition of requirements to standardize communications between data channels and I/O control units. Interface lines transmit all signals between the channel and the I/O control units.

Interface lines can be divided into functional groups (Figure 1). Bus out lines transmit address bytes, data bytes, and command information out of the data channel to the control unit. Bus in lines transmit addresses, data bytes, status bytes, and sense information from the control unit to the data channel.

Inbound and outbound tag lines identify bytes on the corresponding bus line groups. Selection control lines provide interlocks to synchronize operations between the channel and the control unit. The final group of lines, metering, provides machine usage information to be recorded on a running time meter.

Initial Selection Sequence

- **The channel communicates with the tape control unit via 34 interface lines.**
- **To initiate an operation, the tape control must proceed through an initial selection sequence.**

Communication between the channel and the tape control to designate and initiate an operation is called the initial selection sequence (Figure 500). Regardless of the operation to be performed, the initial selection sequence is the same (Figure 7A).

To begin an on-line mode I/O operation, the channel transmits a byte containing the addresses of the tape control and the desired tape unit. The channel activates "address out" to identify the information on the bus out lines. Nanoseconds later, the channel conditions "select out." Interface lines service several control units. All these control units attempt to decode the address. Because each control unit has a different predetermined address, the interface byte designates only one control unit. The tape control recognizes the address if the following conditions exist:

1. The parity of the byte on the bus out lines is odd.
2. Bits 0, 1, 2, 3, and 4 in the character agree with the address card in the tape control (bits 5, 6, and 7 in the address byte select the tape unit).
3. The tape control is not performing a previous command.

If the tape control successfully decodes the address, it resets circuits in preparation to execute the operation and obtains status information from the selected tape unit. The tape control acknowledges receipt of its address by conditioning "operational in." When the channel cancels "address out," the tape control returns its own address over the bus in lines and activates "address in." The control unit address is returned to verify that the correct control unit has responded.

In response to "address in," the channel sends a command byte on the bus out lines and conditions "command out." The command byte designates a tape operation and establishes conditions to control execution of the operation. Although bits in the command byte set the tape control command register identifying the operation, execution of the operation does not yet begin.

The tape control returns a status byte on the bus in lines and activates "status in." The status byte indicates the state of the tape control and the selected tape unit.

A status byte is a means of informing the data channel of conditions in the control unit and selected device. Bits in the initial selection status byte tell the channel whether the control unit and device can perform the operation. During normal operations when there are no interfering status conditions, the initial selection status byte will be blank (P bit only).

The channel conditions either the command out or service out interface line. A channel "service out" response to "status in" signals the tape control that status was accepted and the operation shall begin. In most cases, a "command out" response signals the tape control to stack (retain) the status and end the operation.

Multiplex and Burst Modes

- **If operating in burst mode, the tape control unit remains connected to the channel through an entire operation.**
- **In multiplex mode the tape control unit connects to and disconnects from the channel on each byte of data and status.**

Interface lines permit the channel to operate I/O control units in either of two modes, burst mode or multiplex mode.

Burst Mode: the channel is committed to one I/O operation until the control unit completes its operation. After starting, the control unit need not identify itself prior to communicating with the channel (Figure 7A).

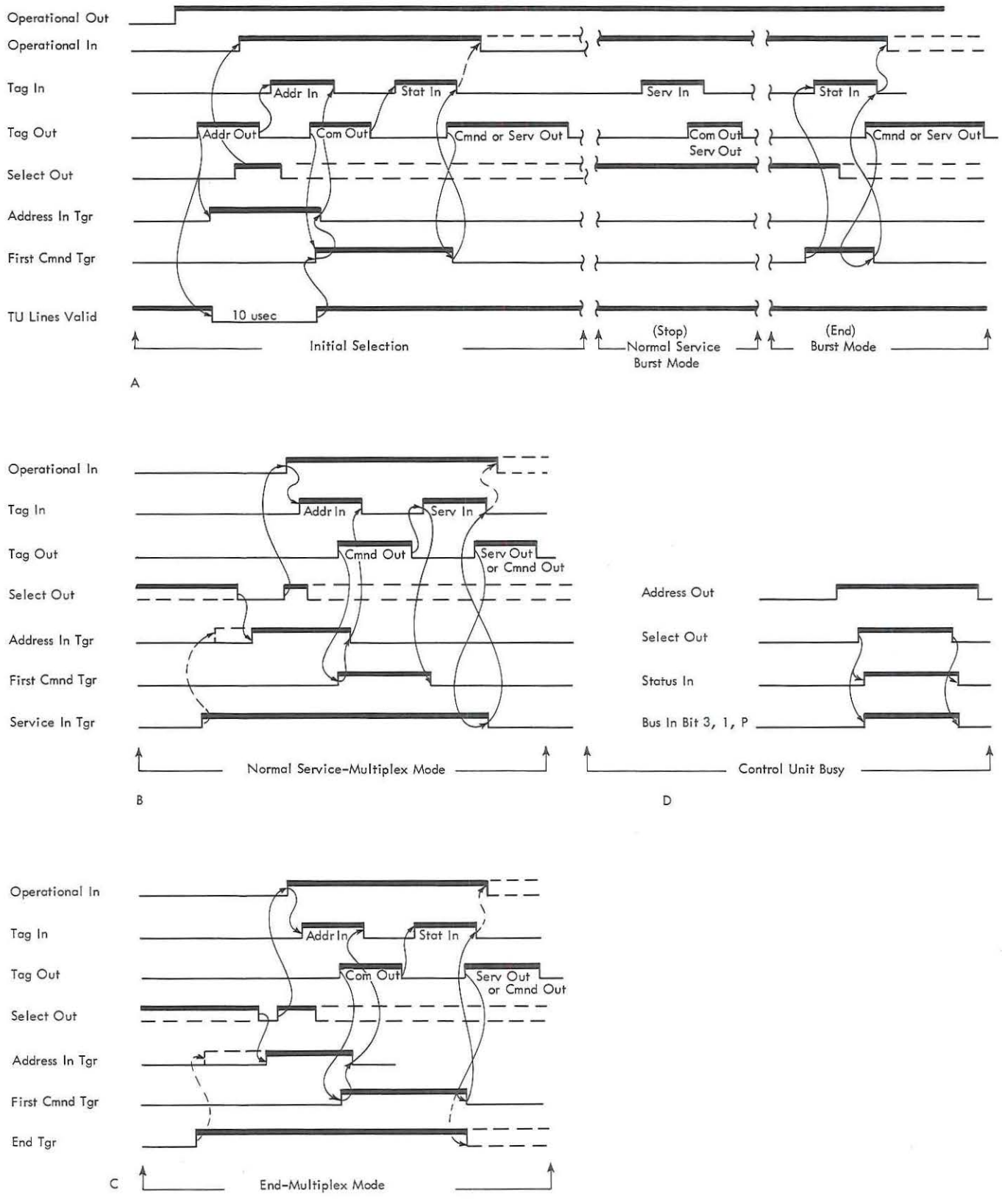


Figure 7. Interface Sequence

Multiplex Mode: the channel is free to control several I/O operations simultaneously. Each control unit must interrupt the channel and transmit its address across the interface and receive a response from the channel before the channel can service the unit (Figure 7B).

The channel can designate that an operation be executed in burst mode by holding the select out line active throughout the operation; "select out" causes the "operational in" line in the control unit to remain in the "on" state. The control unit demands burst mode operation if its force burst mode trigger is on. Force burst mode holds the operational in circuit active even if the channel drops select out. If the force burst mode trigger is off, the channel can revert from burst to multiplex mode after the operation begins.

Data Transmission

Transfer of data across the interface is done one byte at a time in both burst and multiplex mode. "Service in" and "service out" are the tags lines which identify and synchronize bytes during transmission. For write operations, the control unit sends a service in signal each time a new byte of data is needed. The data channel places a new byte on the bus out lines and activates a service out signal to inform the control unit that the byte is available.

During read operations the control unit activates "service in" each time a byte from the tape unit is ready to be sent to the channel. After the channel accepts the byte from the bus in lines, it informs the control unit with "service out."

End Status

At the end of the operation, the tape control sends another status byte to the channel. The end status byte indicates "device end" and whether errors were encountered in executing the tape command.

Depending on the operation being executed, the tape control or the channel can initiate actions to transmit the status byte and end the operation. When the channel starts the end operation, it responds to a request for service ("service in") from the tape control with "command out" rather than "service out." After a delay, a second signal sequence is necessary to transmit the status byte and complete the ending procedure. When the tape control initiates the end operation, only one signal sequence is required, a status in rather than a request for service ("service in").

In multiplex mode, the tape control signals the channel by conditioning "address in" and transmitting its address on the bus in lines when status information is available (Figure 7C). After the channel answers "address in" with "command out," the control unit gates the status byte to the bus in lines and activates "status in." In burst mode, the tape control need not identify itself prior to sending the status byte to the channel. The channel responds to "status in" with "service out" or "command out." "Service out" indicates that the channel accepted the status data and resets the "operational in" in the tape control.

"Command out" indicates that the channel rejected the status byte. Again, in most cases, a "command out" response to "status in" implies that the control unit should "stack" (retain) its status byte and attempt to submit it again. However, if the channel is holding "suppress out" active, the tape control does not attempt to transmit the status byte again until "suppress out" drops. If, however, "suppress out" is inactive, the tape control transmits its address again. When the channel responds with "command out," the control unit gates status information to the bus in lines and conditions "status in." Each time that the channel rejects the status byte (responds to "status in" with "command out") without conditioning "suppress out," the control unit repeats the interrupt procedure.

The following unit status conditions are detected by the tape control unit (TCU) and are indicated to the channel in the status byte:

BIT	DESIGNATION	INTERPRETATION
0	Attention	Not used.
1	Status modifier	Present with busy to indicate TCU busy.
2	Control unit end	Signaled by the TCU: (a) At completion of an operation during which a TCU busy was indicated. (b) At the completion of an operation during which the TCU remained busy and a unit check or unit exception is detected. (c) At completion of a command on the alternate interface of a simultaneous read/write TCU that caused the TCU busy to be given.
3	Busy	When presented without bit 1 (status modifier bit), indicates that the addressed tape unit is busy, or that the TCU has an interrupt pending.
4	Channel end	Indicates that a read, read backward, write, control, or sense has been completed; or in the case of certain control commands involving tape motion, that the operation has been initiated at the TCU and the channel has been released.
5	Device end	Indicates: (1) the tape unit has completed a command, (2) the tape unit has changed from not ready to ready if an attempt had been made to select it while it was not ready, e.g. by issuing a test I/O instruction, (3) a tape unit has reached load point as the result of a program initiated rewind, or (4) a rewind unload is completed at the control unit level, i.e., when the tape unit becomes not-ready.

BIT	DESIGNATION	INTERPRETATION
6	Unit check	Set whenever: (a) Any bit is on in sense byte 0. (b) Tape unit performs read backward, backspace block, or backspace file into or at load point. (c) A rewind and unload is completed at the TCU level.
7	Unit exception	Set when: (a) A write, write tape mark, or erase gap operation is performed in the end of tape area. (b) A tape mark is sensed during a read, read backward, forward space block, or backspace block.

Tape Commands

- **Five basic commands plus control commands can be performed in the tape control unit:**
  1. **Write**
  2. **Read Forward**
  3. **Read Backward**
  4. **Sense**
  5. **Test I/O**
  6. **Control**
- **"Bus out" bits 5, 6, and 7 denote the basic category of the command being issued.**
- **Command register decodes the command byte.**
- **Control commands do not transfer data to or from the channel.**

All commands which can be executed by the 2403/2803 tape control unit are described in general terms within this section. The actual bit configuration of each command byte is shown on Figure 101. The eight bits shown on Figure 101 do not include the parity bit. Each command byte must contain odd parity since it is transmitted on the interface bus out lines.

Write Command Operation

- **Accept data from the channel interface and transmit the data to the tape unit.**
- **Read check the data written by the tape unit and indicate any errors detected.**

In a tape write operation, channel transfers one byte at a time across interface lines to tape control. Tape control initiates all data transfers from channel and checks each input byte for odd parity, discards the check bit, and stores other bits of the character in the data register. In processing bytes from the data register through the read/write register to the selected tape unit, tape control either:

1. transfers all bits in eight-bit bytes (nine-track operation) (Figure 305), or
2. transfers six bits of the eight-bit byte and discards the two high-order positions (seven-track translator off mode) (Figure 307) or
3. converts 3 input eight-bit bytes to 4 six-bit characters (seven-track data converter on mode) (Figure 309) or
4. translates each eight-bit byte to equivalent six-bit BCD character (seven-track translator on mode) (Figure 307).

A mode set command previous to the write operation sets the mode register to establish how data are processed during seven-track write operations. In all cases, however, the tape control transfers one character to the tape unit during each write clock cycle.

The designated tape unit writes each byte received from the tape control. As the tape moves across the head, the tape unit reads each byte written previously and returns the data to tape control. Tape control examines the returned tape unit data for recording errors that might have occurred. Tape control processes channel data to the tape unit at the same time that it performs a check operation on bytes that the tape unit has written earlier.

When channel transfers the last byte in the block, it answers the subsequent request for data from tape control with command out, indicating that the last byte has been transmitted. Tape control causes the tape unit to write the check character(s) at the end of the block. When the tape unit reads the check character(s) and returns them to tape control, tape unit actions in the write operation end. Tape control examines the returned check characters, resets circuits employed in the write operation, and transmits a status byte to channel. Channel must accept the status byte before tape control can actually complete the write operation and execute another tape command.

Read Command Operation

- **Accept data from the tape unit and transmit the data to the channel interface.**
- **Check the data as it is read and indicate any errors detected.**

In a read operation, tape control receives six-bit or eight-bit characters from the tape unit and transmits odd parity eight-bit bytes to channel. Tape control initiates all byte transfers to channel. Data transfer rate from the tape unit is controlled by tape unit reading speed. In processing a byte, tape control strips the parity bit before storing bits in the data register. Tape control assigns the P bit in each byte sent to channel to insure that each byte on the bus in lines contains odd parity.

In processing tape data to channel, tape control either:

1. transfers to channel all bits in the eight-bit byte received from the tape unit (nine-track operation) (Figure 306) or

2. changes each input six-bit character to an eight-bit byte by adding two zeros in the high-order position and transfers the eight-bit byte to channel (seven-track, translator off mode) (Figure 308) or

3. converts 4 input six-bit characters to 3 eight-bit bytes for transmission to channel (seven-track converter on mode) (Figure 310) or

4. translates each six-bit BCD character to an equivalent eight-bit BCD byte (seven-track translator on) (Figure 308).

Bits in the mode register (from the previous mode set command) control the manner in which the tape control processes data characters during seven-track operation.

Channel is not required to accept all bytes in the tape block. After accepting the first byte, channel can terminate output transfers from tape control at the end of any cycle. However, tape control and the tape unit are committed to the operation until the tape unit reads the complete block. If the tape unit has not transferred the last normal byte in the block when channel indicates that it will not accept more data bytes in the operation, tape control does not initiate more data transfers to channel. It receives, checks, and then discards bytes subsequently received from the tape unit.

At the end of the operation, tape control transmits a status byte to channel. If the record was a tape mark, no bytes are transferred to channel and the end status byte contains a unit exception status bit. Channel must accept the status byte before tape control is considered finished with the read operation.

Read Backward Command Operation

- **Set the tape unit to backward read status.**
- **Accept bytes from the tape unit and transmit the bytes to the channel interface.**
- **Check the bytes as they are read and indicate any errors detected.**

Read backward operation is similar to read forward except that the tape moves in the reverse direction and the check characters are read before the data. The data flow path and controls are the same as for read forward except that the data converter mode cannot be used. Read backward operations will automatically be in data converter off mode. The translator will operate normally.

A tape mark record will appear the same when read forward or backward. A tape mark consists of two identical characters, a tape mark, and a tape mark check character. Because the configuration is symmetrical, the detection circuits are similar for forward and backward.

If the tape unit is at load point, a read backward operation will be terminated immediately.

Motion Control Commands

- **Control commands that require tape motion are:**
  1. **Erase**
  2. **Rewind**
  3. **Rewind unload**
  4. **Write tape mark**
  5. **Backspace record**
  6. **Backspace file**
  7. **Forward space record**
  8. **Forward space file**
- **A motion control command byte contains "1" bits in positions 5, 6, and 7.**
- **Bits in positions 2, 3, and 4 specify the operation to be performed.**

Motion control commands provide the ability to move tape without transmitting data to and from the channel. The write tape mark command does write on the tape but the bytes written are a pre-determined combination that requires no data from the channel.

The initial selection status byte of a motion control command contains only a channel end bit. Channel end signifies that the channel is no longer required for the operation and channel disconnects from the control unit at the end of initial selection. Execution of control commands is completed by the control unit and tape unit.

When all motion control operations are completed, except rewind unload, an end status byte is sent to the channel. This status byte must be submitted via a multiplex mode (request to lock to channel and identify itself) because channel was disconnected some time ago (end of initial selection). The end status byte contains normally only a device end bit to indicate that the device (tape unit) has completed an operation. Tape unit and control are then free to execute a new command.

Rewind

A rewind command causes the tape to move backward to the photo reflective marker (load point) at the beginning of the tape. As soon as the tape unit begins rewinding, the control unit is free to perform other operations. When the marker is reached, tape motion stops and the tape unit is available for a new operation.

When the tape unit has rewound tape to load point, it signals the tape control. The tape control sends a "request" to channel (multiplex mode) to accept status. This time the (end) status byte contains device end.

Rewind Unload

A rewind unload command moves tape backward to the photo reflective marker at the beginning of tape and then unloads the tape. The tape must be manually re-

loaded before further operations can be done on that unit.

After initial selection is complete, tape control begins the rewind-unload operation at the tape unit. Then, unlike rewind, tape control requests channel (multiplex mode) to accept a second status byte regardless of whether or not channel has channel end from first status byte. The second status byte contains unit check, device end and control unit (CU) end status bits. If channel did refuse the initial status byte (channel end), then CU end is replaced by channel end. After the tape unit completes the rewind-unload, and the unit is reloaded and made ready, the tape control is again signaled. Channel is then requested to accept an end status byte (third byte) which includes only a device end bit.

Erase

Execution of the erase instruction causes the selected tape unit to erase approximately 3¾ inches of tape. Tape control starts the erase and write operations in a similar manner. In both operations it conditions a write delay period to start tape motion and start the delay counter to measure the length of the erased portion of the tape. When the delay counter reaches a count of WD-320, the operation is ended and a status byte containing only device end bit (multiplex mode) is sent to the channel.

Write Tape Mark

In a write tape mark operation, tape control generates a tape mark character internally, transfers the character to the selected tape unit, then resets the tape unit write triggers. In switching to their off states, write triggers in the tape unit record a check character. After writing the tape mark and LRC character, the tape unit reads them back into the tape control. Tape control checks the returned characters and ends the operation. Tape control actions in the write tape mark operation are similar to the actions that it performs in the execution of a normal write operation. In a write tape mark operation, however, communication between channel and tape control after the initial selection sequence is required only in the status byte transfer (device end via multiplex mode). Data are not processed across interface lines in executing a write tape mark operation.

Backspace Record and Backspace File

In a backspace record operation, the selected tape unit moves tape backward through one record to the interrecord gap, or to the gap between load point and the first tape record. In a backspace file operation, the selected tape unit moves tape backward to the first interrecord gap after sensing a tape mark record, or to the gap between load point and the first tape record. The tape unit reads characters recorded on tape and sends them to the tape control. Tape control does not execute data transfers either to channel or to the tape unit when performing the backspace record or backspace file operation. The tape control unit scans the bytes to find the correct stopping point. A backspace record or file into or at load point constitutes an error condition.

Forward Space Record and Forward Space File

In a forward space record operation, the selected tape unit moves tape forward through one tape record to the next interrecord gap or to the gap between the last record and the end of tape. In a forward space file operation, the selected tape unit moves tape forward to the first interrecord gap that it senses after reading a tape mark record, or to the gap between the last record and the end of tape. The tape unit reads characters recorded on tape and sends them to tape control. Tape control does not execute data transfers either to channel or to the tape when performing the forward space record or forward space file operation. The tape control unit scans the bytes to find the correct stopping point.

Sense Command

In a sense operation, tape control transfers up to six sense bytes across the interface lines to channel. Bit positions in each sense byte represent conditions in various tape control circuits. During a sense command the tape control does not communicate with any tape units. During transfer of the sense bytes, TC and the channel operate in burst mode. The normal end status, also burst mode, contains channel end and device end bits. Figure 8 lists the sense byte positions and the con ditions they represent. There are no circuits in the tape control to provide data for the sixth sense byte, therefore it remains blank.

Test I/O Command

Test I/O provides a means for the channel to determine if I/O units are available for use. Test I/O requires only initial selection time since only the initial selection status byte is used by the channel. If the tape unit and tape control are available the initial status byte will be returned blank (P bit only). If there is status information stored in the control unit, the status can be sent to the channel in the test I/O status byte and relieve the TC of any outstanding (stacked) status.

Byte 1	Unit Check	Byte 2	
0	Command Reject	0	Noise
1	Not TU Status A	1	TU Status A
2	Bus Out Check	2	TU Status B
3	Equipment Check	3	Seven Track
4	Data Check	4	At Load Point
5	Overrun	5	Not Read (Write) Status
6	Word Count Zero	6	File Protected
7	Data Converter Check	7	Spare
Byte 3	**Track In Error**	**Byte 4**	**Data and Equipment Check**
0		0	R/W VRC
1		1	LRCR
2		2	Skew
3		3	CRC
4		4	Skew Reg VRC
5		5	(Spare)
6		6	Backward Memory
7		7	C Compare
Byte 5	**Equipment Check**	**Byte 6**	**All Blanks**
0	Echo Error	0	
1	Reject Tape Unit	1	
2	Read Clock Error	2	
3	Write Clock Error	3	
4	Delay Counter Error	4	
5	Sequence Indicator C	5	
6	Sequence Indicator B	6	
7	Sequence Indicator A	7	

● Figure 8. Sense Operation Data

Mode Set

- **Set the mode register from the bus out positions 0, 1, 2, 3, and 4 during initial selection.**
- **Set mode register to inform TC the mode of seven-track operations.**

A mode set control command establishes conditions for all succeeding seven-track operations. Mode register triggers are turned on by corresponding bits 0 through 4 of the mode set command byte. The mode set command requires only an initial selection period to be executed, so the command register is not set.

Positions 0 and 1 of the mode register are decoded to establish the density of characters on tape. Position 2 controls the data converter. Position 3 determines whether the parity circuits operate in odd or even redundancy. Position 4 controls the translator.

Once the mode register is set, it will remain in that status until the next mode set command. Data transmission commands executed during this time will not change the setting of the mode register. Each seven-track read or write command will operate under the conditions specified by the last mode set. Nine-track operations override mode register settings.

All nine-track operations are performed in the standard mode: 800 bytes/inch density, odd parity, translator off, and data converter off. The mode register setting is not used or changed for nine-track operations. It is merely ignored. Nine-track operations automatically use the standard mode. If a seven-track operation is performed later, it will be subject to the conditions still in the mode register from before.

At the end of initial selection, the mode set command has been fully executed and the initial selection status byte normally contains channel end and device end bits.

Diagnostic Mode

Diagnostic mode is used with the CE panel to check tape operation. With the diagnostic plug in place in the CE panel, a diagnostic mode command will turn on the diagnostic trigger. During succeeding write operations, the diagnostic trigger inhibits the TC from placing any bits on the P track of the write bus. Thus, no parity bits are written on tape. Although the diagnostic command is listed as a mode set command, it does not alter the contents of the mode register. It requires only an initial selection period to be executed and will force a channel end and device end bit in the initial selection status byte.

No-Op Command

A no operation (no-op) command does not transmit data or move tape. No-op performs no function in the tape control or tape unit. At the end of initial selection of a no-op command, the tape control is reset and available for the next command. No-op is used as a time delay or to fill a blank space in a program. It actually operates the same as a mode set command but does not alter the contents of the mode register.

Request TIE Command

Request TIE command is used primarily in nine-track error correction and will transmit track in error (TIE) information from the channel to the control unit. If a correctable error was found while reading tape, a request TIE is used to inform the tape control to correct any errors on the next read operation and which track of the tape should contain the error.

During the request TIE command, one byte of data is transmitted from the channel to the tape control after initial selection in burst mode. This byte should contain one bit indicating which track was previously in error. Request TIE also turns on the correction trigger to indicate that the next operation will be a correction read. Thus, on the following read command, if any errors occur, TC will know to perform correction (correct trigger on) and which track to correct (byte from request TIE command).

Turnaround

- **Turnaround is needed if selected tape unit is in opposite directional status of command to be executed.**
- **Delay tape control action until turnaround is completed on selected tape unit.**

- **Possibility of turnaround is checked prior to any movement of tape.**

The command, brought out during initial selection, indicates whether the selected tape unit must move tape forward or backward. If the new command must move tape in the opposite direction from the condition left by the previous command, a turnaround delay is required. Turnaround delay holds up control unit action until the tape unit has had time to change the status of relay circuits and the reel drive mechanism.

If a backward type operation is begun when a tape unit has just completed a write type (forward) operation, the turnaround sequence first advances and erases tape forward. This precaution is taken to avoid leaving "noise" information on tape too close to the last good record. As the tape unit switches from write to read status, a weak, but potentially troublesome, noise splash is "written" by the write heads being turned off. By moving the last tape record farther away from the write heads, the noise will not cause subsequent read errors.

Error Detection Circuits

The 2403, 2404, 2803, 2804 tape control units employ a variety of error detection circuits. These circuits can be grouped as follows:

1. Data checking circuits.
2. Equipment checking circuits.
3. Checking circuits which do not set data check or equipment check.

Data checking circuits are designed to monitor the condition of data bytes as they pass through the tape control unit. Data errors will set a bit in a sense byte (Figure 8). Data check circuits are:

1. Read/write vertical redundancy.
2. Skew register vertical redundancy.
3. Longitudinal redundancy.
4. Skew.
5. Cyclic redundancy check.
6. C-Compare.

Equipment checking circuits monitor the performance of the circuits in the tape control. Equipment check circuits include:

1. Reject tape unit.
2. Read clock VRC.
3. Write clock VRC.
4. Delay counter VRC.
5. Echo check.
6. Sequence indicators.

The six preceding causes of equipment check also gate bits in the sense bytes.

The third group of error check circuits do not set data check or equipment check. Each of these circuits will set a bit into a sense byte (Figure 8).

1. Forward stop delay or write delay noise.
2. Command reject.
3. Bus out check.
4. Overrun check.
5. Word count zero.
6. Data converter check.

Any check circuit in the three groups (except FSD/WD noise) will cause a unit check bit in a status byte. Noise turns on an indicator but does not turn on unit, data, or equipment check. When a unit check is indicated in a status byte, the computer program usually performs a sense command to find out which condition caused the unit check.

Read/Write Vertical Redundancy Check (VRC)

Bytes in the read/write register are checked for odd or even parity during read and write operations. For seven-track operation the mode register specifies whether data parity should be odd or even. Nine-track data bytes should always be odd.

The read/write register VRC circuit indicates errors detected during read or read backward operations only. It does not indicate errors during write. For write operation the read/write VRC circuit is used to generate correct parity in the bytes written on tape.

Skew Register Vertical Redundancy Check (VRC)

During read and write operations bytes in the high clip skew register are checked for odd or even parity. For seven-track operations the mode register specifies whether the data bytes should be odd or even. Nine-track data bytes should always be odd.

The skew register VRC circuit indicates errors only during checking of a write operation. For seven-track read and read backward the skew register VRC is used to indicate whether the byte in the high clip or low clip skew register is transmitted to the R/W register.

Longitudinal Redundancy Check Register (LRCR)

The LRCR checks the longitudinal bit count in each track of a block. A trigger for each track turns on or off for each bit in the track. Since each track in a block has an even number of bits, all the LRCR triggers should be off at the end of the block. An active output from any LRCR trigger at the end of an operation turns on LRCR error.

An LRCR error can be indicated during read, read backward, or write (read checking).

Skew Error

During write, skew error circuits check bytes (after they are written) to see if all the bits in each byte are written at a right angle to the edge of the tape.

The write head on the selected tape unit should write all bits in a character at a 90-degree angle to the

edge of the magnetic tape. When the character passes the read head, the tape unit transfers all bits to the tape control.

Several conditions can cause improper alignment of bits across the face of the tape, preventing the simultaneous transfer of all bits in the character to the tape control. The time interval between the first and last bits must not exceed a predetermined amount. Otherwise, a "skew gating" circuit indicates a skew error.

CRC Error

The CRC error indicates that the cyclic redundancy check character, computed during a read or read backward, produced an improper result (bit failure). At the end of a read or read backward operation the CRC should contain the "match pattern" of bits 111 010 111. If this pattern is not produced, the CRC error is turned on.

C-Compare

The C-Compare circuit is a check on data transmission from bus out to the read/write register for write operation, or from the read/write register to bus in for read operation. A C-Compare error indicates that the parity of a byte changed during transmission through the tape control. C-Compare is active for all read and write operations except seven-track translate on mode. Bus out parity errors, or attempted error correction on more than one track at a time can cause C-Compare errors.

Reject Tape Unit

Reject tape unit is a check made on the selected tape unit by TC under three different conditions:

1. Loss of select and ready from the tape unit after an operation has started tape motion (after initial selection).
2. The selected tape unit did not revert to a read status for read type commands (write status still active).
3. The selected tape unit did not revert to a write status for a write tape command (read status still active).

Any of the three conditions produce a reject tape unit error and terminate the operation immediately.

Read Clock, Write Clock, and Delay Counter VRC

The clock and counter VRC circuits detect malfunctions in the stepping of the triggers in these timing circuits. If the triggers do not step in the correct sequence, a clock error is signaled.

Echo Error

The echo check circuit checks to see that the tape unit actually writes each byte transmitted during write operation. During each data cycle the no echo trigger is turned on in the tape control. When the tape unit writes the bits on tape an "echo" is returned to the tape control. The echo pulse resets the no echo trigger. If the no echo trigger remains on, an error is signaled because the tape unit did not return the expected echo.

Sequence Indicators A, B, and C

Sequence indicators check to see that operations in the tape control progress through the correct steps (sequence).

As each operation progresses through various stages, the sequence indicators (three-stage counter) step through a cycle. This counter is stepped by pulses which originate if particular conditions are met as the operation progresses correctly. At the end of each correct operation, the indicators (counter) should have stepped up to a maximum count and back down to a reset condition. If the control unit does not progress through correctly, one or more sequence indicators will be left on at the end of the operation. At the end of each operation the sequence indicators are sampled. If any of the three remain on, a sequence error is signaled.

The control unit could "hang up" and not complete an operation. When this condition arises, the sequence indicator lights on the CE panel will show how far the operation progressed before "the hang up."

Forward Stop Delay or Write Delay Noise

At the end of a read or forward space operation, the tape should be blank after the check characters. The forward stop delay circuit detects characters or noise bits on the tape. Bits read from tape during write delay will also turn on the noise trigger.

The noise trigger indicates that the noise condition has occurred. Noise does not set an error bit in the *status byte*. The channel is notified of FSD noise by a bit in a *sense byte*.

Another use for FSD: if for any reason the tape unit stops in the middle of a long record, the FSD circuit will allow easier recovery. On the following tape motion command the tape cannot physically move at full speed immediately. When a restart is attempted from the middle of a record, the slow tape speed makes normal character times appear to be check characters so the tape control would attempt to stop further in the record. FSD will keep the tape moving to the next record gap and turn on the FSD noise trigger, thereby skipping over the entire record or block.

Command Reject

Command reject is indicated when the tape control receives certain commands it cannot perform. A write, write tape mark, or erase command addressed to a file protected tape unit will turn on command reject. A

"data converter on" mode set will be rejected if the control unit has no data converter feature installed.

Bus Out Parity Error

All bytes from the channel interface bus out lines should be in odd parity. If a byte is received with even parity, a bus out parity error is signaled.

Overrun

Overrun error is turned on whenever the control unit requests service and finds the channel has not acknowledged a previous request for service. If the read/write control trigger indicates that the previous character has not been processed when the next one is received, an overrun error is signalled.

Word Count Zero

In write operation, the word count zero trigger is set when the channel responds with command out to the first service in. No data will be written on the tape and the tape does not move.

Data Converter Check

Data converter check can be set only during a seven-track read operation in data convert mode. When the data converter is on, bytes are handled in groups of four characters, when read from tape (read operation). If the tape record does not contain an exact multiple of four characters, the last data byte to channel might be padded with zeros. A data convert check is indicated only if the last data byte to channel has been padded. This procedure is explained in greater detail in the features section of this manual.

The following chart summarizes the error checks and sense byte conditions. All the errors listed in this chart have indicator lights on the CE panel.

Error	Check Type			Sense Byte				
	Unit Check	Equip Check	Data Check	1	2	3	4	5
Bus Out Parity	X			Bit 2		E.		
Command Reject	X			Bit 0				
Word Count Zero	X			Bit 6				
Reject TU	X	X		Bit 3		P.		Bit 1
Delay Counter VRC	X	X		Bit 3				Bit 4
Write Clock VRC	X	X		Bit 3				Bit 3
Read Clock VRC	X	X		Bit 3		R.		Bit 2
R/W Register VRC	X		X	Bit 4			Bit 0	
Skew Register VRC	X		X	Bit 4		0	Bit 4	
LRC	X		X	Bit 4			Bit 1	
Skew Error	X		X	Bit 4			Bit 2	
Echo Error	X	X		Bit 3				Bit 0
C Compare	X		X	Bit 4			Bit 7	
Sequence Indicator A	X	X		Bit 3				Bit 7
Sequence Indicator B	X	X		Bit 3				Bit 6
Sequence Indicator C	X	X		Bit 3				Bit 5
Noise					Bit 0			
Overrun	X			Bit 5		7		
CRC Error	X		X	Bit 4			Bit 3	
Data Convert (Feature)	X			Bit 7				

Error Correction

- **Error correction technique employs a modified cyclic code in conjunction with character parity to correct error bursts of unlimited length in any *one* of the nine tracks.**
- **Employed only on nine-track tape units.**
- **Errors involving more than one track within the same record are detected but *not* correctable.**
- **Encoding and error correction are performed in the tape control unit.**

The IBM 2400 tape system can correct most single track read errors when operating in nine-track mode. Errors can be caused by the following defects:

1. Tape damage due to improper handling.
2. Oxide particles on the tape and head.
3. Voids in the oxide coating of the tape.
4. Dust and other foreign particles on the tape and head.

These defects usually affect bits in only one track because the spacing of the tracks is many times the spacing between bits along a track. Error correction is designed to correct almost any pattern of erroneous bits as long as they are in the same track within a record.

Error correction is based on the assumption that the record was written correctly, and that defects occurred after writing. No attempt is made to correct bits on the tape. Only the information sent to the data channel is corrected.

Cyclic Redundancy Check

- **To permit error correction, the tape record contains a cyclic redundancy check character.**

To identify which track contains errors, two check characters are written at the end of the record (instead of only one as in former tape systems). The added check character is called a cyclic redundancy check (CRC) character; it is written after the data but before the usual longitudinal redundancy check (LRC) character. The circuit that generates the CRC character is based on a complex equation that makes the mathematical probability of an undetected error almost zero. The mathematical formula used is involved and is of no use in troubleshooting, but it is possible to shift characters into the cyclic redundancy check register one at a time and predict what the results will be after each shift.

The CRC character aids in error detection, but its primary function is determining which track contains the error. When a record is read, a new CRC character is computed and combined with the CRC character from the tape. The result should be a "match pattern,"

111 010 111 in the CRCR. If the record contains an error, the erroneous CRC value is used to determine the track in error.

Error Pattern

The erroneous CRC character by itself does not contain enough information to find which track contains the error. The error pattern register (EPR) supplies the missing information. During reading, the EPR keeps a record of the read/write register redundancy errors in the form of an error pattern. If there are no errors, the EPR remains blank.

At the end of an error record, the error pattern (saved in the read/write register) is compared with the CRCR. The error track is located by counting the number of shifts needed to reach a zero pattern. If all nine tracks have been tried unsuccessfully, shifting and comparing stop because no track in error can be found. Failure to find the track in error could be caused by read errors in more than one track, or electronic error in the circuits.

When the error correction circuits locate a track which contains an error, the bit from the EPR is set into sense byte 3.

The tape control informs the CPU program with the bit in sense byte 3. If no track is found, bits 6 and 7 are forced on. Sense byte 3, therefore, contains either a single bit denoting the track in error or bits 6 and 7 indicating that no error track could be found.

Repositioning Tape

Error detection and correction can take place in either forward or backward read operations. If an error is detected in a forward read, it should be corrected in a forward read; a backward error should be corrected in a backward read. Consequently, the tape must be repositioned to read the record again in the same direction.

It is technically possible to correct a forward read error during a backward read, and vice versa. However, this not recommended because the errors tend to differ when read in opposite directions. The tape may shift slightly, or oxide particles may change position with a reversal of movement.

In some cases the change in physical conditions may be an advantage. A record read in one direction may produce uncorrectable errors, yet produce good data when read in the opposite direction. An error record should be read in both directions before it is considered unreadable.

Request TIE for Correction

After a track in error is found and the tape is repositioned, the track in error bit must be sent back to the tape control. The track in error bit in the EPR will be reset during the repositioning operations. A request track in error (TIE) command requests a byte from the channel. The channel sends back the track in error byte on the bus out lines.

It is the responsibility of the CPU program to be certain that the correct track in error byte is sent to the tape control. The CPU program must also make sure that repositioning and correcting instructions are addressed to the same tape unit that read the errors.

The track in error byte should contain one bit position, indicating which track read the errors. In normal correction operation this should be the same bit position that the tape control transmitted to the channel in sense byte 3. If the byte in the EPR from the request TIE command contains 6 and 7 bits, no correction takes place. (The 6 and 7 bits in sense byte 3 indicated that no track in error was found.)

Correction

- **During a correction read, characters are corrected at the outputs of the read/write register.**
- **The correction is checked by computing a new CRCC from the corrected data.**
- **Error detection is performed automatically on read operations but error correction can only occur if TC is programmed to do so.**

Actual correction of bytes is at outputs of the read/write register. Bit positions from the R/W register pass through exclusive OR circuits. With no bit from the EPR, the data passes through unchanged. When a VRC error is detected in the R/W register, the outputs of the EPR are gated into the exclusive OR circuits. EPR positions that contain zero have no affect but the track in error bit inverts the corresponding output line. If the R/W register output in the TIE position has a bit missing, it is restored and a bit picked up is deleted. The byte on the R/W output lines should then be restored to its correct condition. This procedure continues for each error byte in the record.

To guard against false correction, a new CRC character is computed using the corrected data. At the end of the record, the corrected CRC character is combined with the CRC character from the tape. If they do not form a match pattern, CRC error turns on at the end of the correction read.

Because the LRCR receives data from the skew register before it is corrected, the LRCR may have an erroneous bit left at the end of the correction. EPR output circuits block LRCR errors from the track in error. The remaining positions of the LRCR are sampled as usual.

CE Panel

- **A CE control panel is built into the front of the A gate in each tape control unit.**

- **CE panel is used on 2403, 2803, and 2804 control units.**

Using the panel, a CE can check out most of the circuitry in the control and tape units without disturbing channel. Most of the CE panel is operative only when the off-line switch is on, logically isolating the control unit lines from the channel interface. A select out signal from channel will bypass the control unit and proceed to the next control unit on line.

Data transmission operations, such as reading and writing, can be simulated without data or controls from the channel.

The CE panel is organized into three sections: indicator lights, control switches, and plugboard. ALD page locations for the CE panel are in Figure 9.

Indicators which are logically related are designated by a horizontal line above all the positions in the group. A lighted indicator designates the 1 bit or "on" conditions of the corresponding circuit.

A description of the CE panel operation is included in the Console and Maintenance Features section.

Simultaneous Read/Write Tape Control Units

The IBM 2404/2804 tape control units attach to two data channels and can perform write and read operations simultaneously on two different tape units. Up to eight tape units, of any speed, can be attached to each control unit. Any two of the tape units can be used simultaneously, provided the operations are not both read or both write.

Tape units attached to a simultaneous tape control must be modified. An additional set of read bus lines is added to allow normal reading on one tape unit at the same time as write/read checking on another tape unit. Duplicate control lines are added, where needed, to perform two operations independently.

Simultaneous tape controls are divided into four logical parts: Interface A, Interface B, read control, and write control. Interfaces A and B connect to two data channels, while the read and write control sections connect to the tape units as in Figure 10.

Data flow and controls operate the same as 2403/2803 tape control circuits. The read control section contains only those circuits needed for read operations and the write control section contains only the circuits needed for write operations. The error pattern register (EPR) is used only by the read control section for read operations. There is no EPR in the write control section.

In the read control section there are no circuits to generate write bus lines to the tape unit. The write control section does contain read bus circuits to read check data written on tape. Two sets of read bus lines come from the tape unit, one set to the read section and one set to the write section. Read and write select circuits determine which set of read bus lines will transmit data from the tape.

The two interface sections (A and B) each contain circuits to communicate with a data channel and process read, write, and control commands. The read control section must be used for these commands:

1. Read
2. Read backward
3. Forward space
4. Backspace
5. Rewind
6. Rewind-unload

The write control section must be used for these commands:

1. Write
2. Write tape mark
3. Erase

Sense, test I/O, no-op, and mode set commands can be performed from either interface and either the read or write control sections. These commands can be received from either interface but they will try the read control section first. If the read control is busy, a sense, test I/O, no-op, or mode set will attempt to select the write control section. Two commands can be processed at the same time as long as they do not both require the same section of the control unit or the same tape unit.

Command interference circuits are provided to prevent conflict between A and B interface commands. If one channel interface attempts to perform a read operation while the other interface is using the read control section, a busy response will be returned to whichever interface attempted to break in. The same conditions apply to write operations and operations that try to select the same tape unit.

When no interfering conditions exist, the command interference circuits provide control lines to connect an interface with a control section and select a tape unit. The interference circuits "lock" the control unit sections in this configuration and generate a busy response if the other interface attempts to select the section or tape unit in use. The control unit will remain "locked" until the command is completed. If a read or read backward command detects a CRC error and identifies the track in error, the read section of the control unit will remain connected to the interface until the sense command is executed. This is done to keep the other interface from selecting the read control section and destroying the track in error information in the error pattern register. The sense command can send the track in error information only to the data channel that originated the command during which the track in error was found. The track in error information must come from the read control section because there is no EPR in the write control section.

If the data channel ignores the read error, a new read type command, from the same interface, will

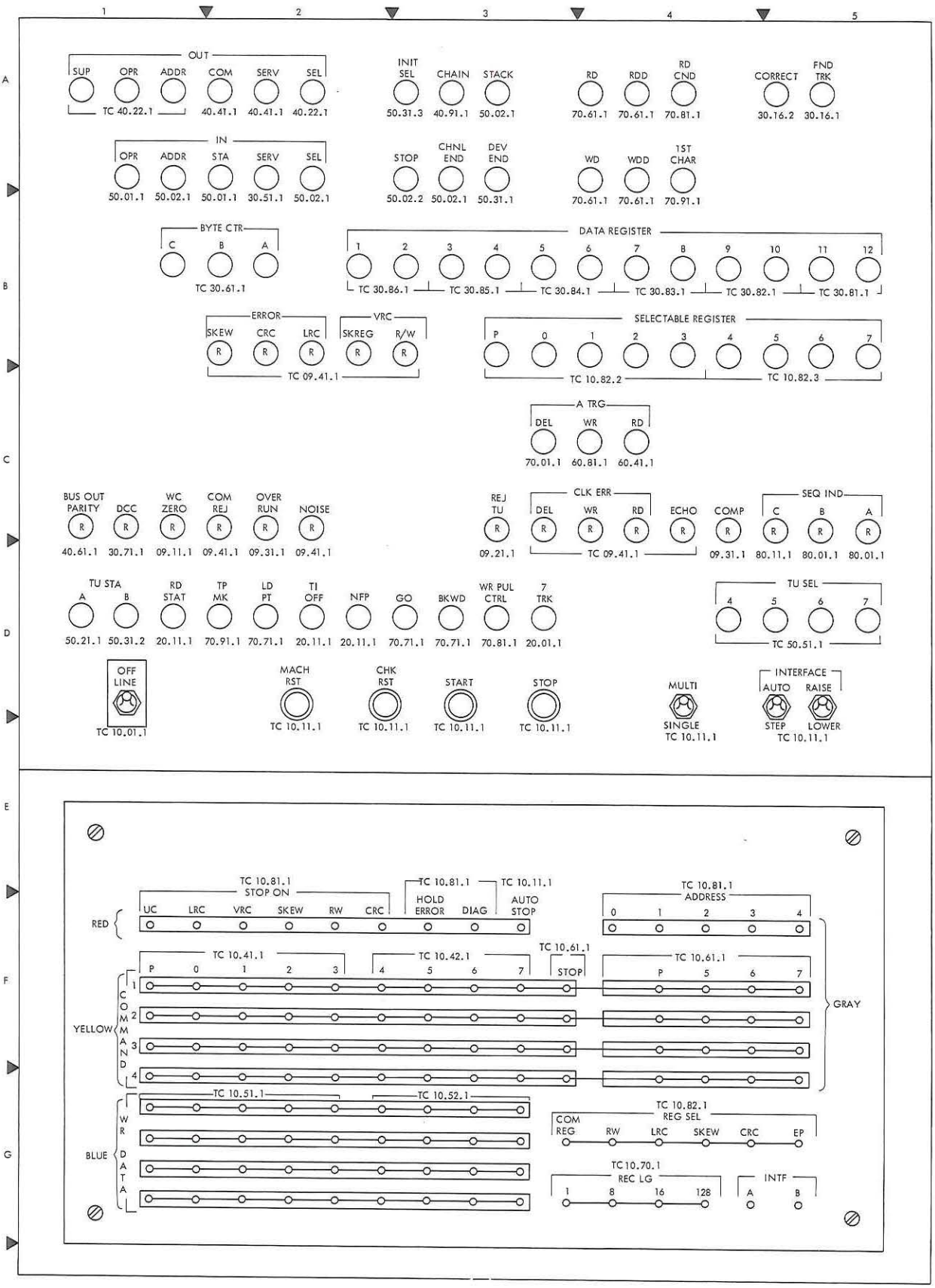


Figure 9. CE Panel Locations 2403/2803

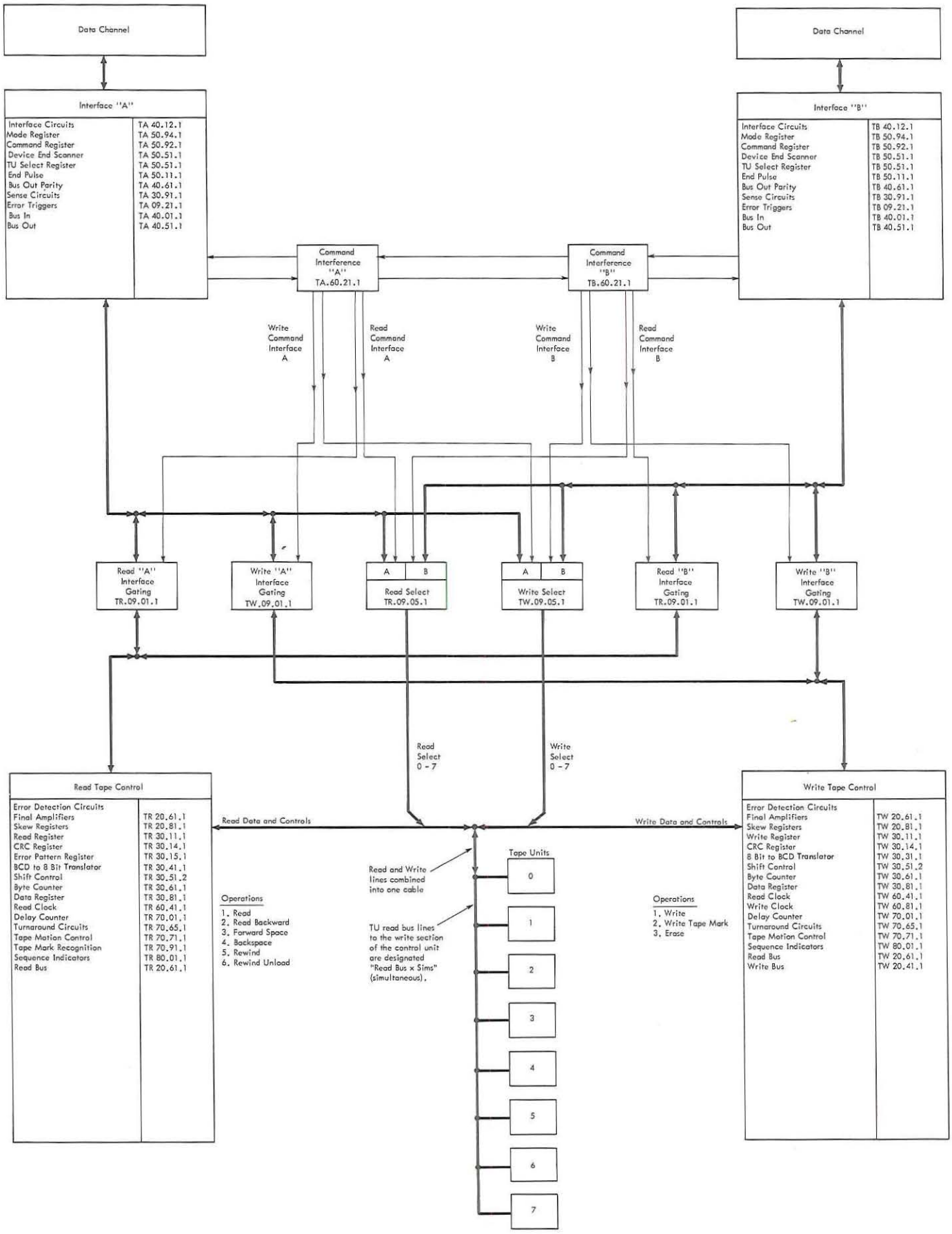


Figure 10. Simultaneous Read/Write Tape Control

reset the found track condition. A write type command from the same interface will "unlock" the command interference circuits and select the write control section if the other interface is not performing a write. In either case the sense information will be changed by the new operation.

The first four functional units in this chapter are data transmission units arranged in the order used during a read operation:

1. Final amplifiers
2. Skew registers
3. Read/write register
4. Data register

Three units are used to provide timing pulses:

1. Read clock
2. Write clock
3. Delay counter

Shift control regulates the movement of data from register to register in both read and write operations. Units used specifically for error correction are grouped together.

The remainder of the functional units provide controls, error checking, and miscellaneous support functions.

Final Amplifiers

- **Final amplifiers shape and amplify read pulses from the tape unit.**
- **Only seven of the nine amplifiers are used for seven-track operations.**
- **Outputs of each amplifier are high clip and/or low clip pulses according to the amplitude of the input pulses.**
- **High and low clip acceptance requirements are variable and are higher for write operations than for read operations.**
- **Output of the final amplifiers is gated to the skew registers.**

Final amplifiers accept pulses from the tape unit read bus lines and convert the pulses to high and low clip data. The high and low clip outputs are used to set the high and low clip skew registers.

Final amplifiers consist of biasing circuits and nine identical amplifier tracks. In tape controls equipped with the seven-track feature, only seven of nine amplifier tracks are used for seven-track operation.

Biasing circuits establish voltage levels for clipping within the final amplifiers to compensate for:

1. frequency differences in data signals from various tape units.
2. different input requirements that the tape control establishes in read and write operations.

Figure 11 illustrates signal amplitude and clipping levels.

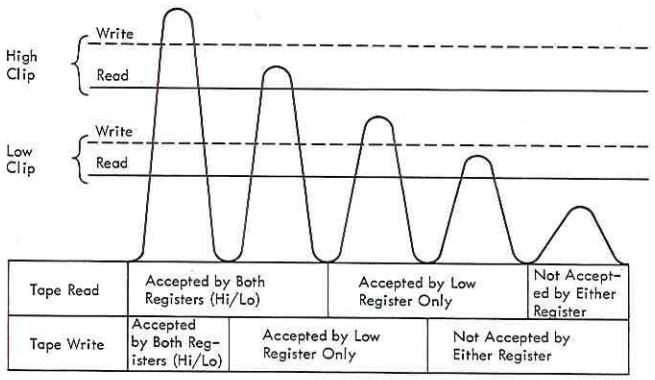


Figure 11. Final Amplifier Clipping

The term "clip" in this usage refers to a method of establishing amplitude requirements for the input pulses. To generate a high clip output pulse, the input must be above a certain level. An input signal below this level will not generate a high clip output but may still cause a low clip output. If the signal is lower than the requirement for low clip output, no output will be generated. A high amplitude input signal will generate both high and low clip outputs.

The low clip outputs are used only for seven-track operation. Nine-track operation establishes more critical requirements so only high amplitude pulses are accepted from the tape unit. Input requirements are higher in write checking operations than in read operations to insure that characters written on tape are of sufficient amplitude to be read in subsequent read operations.

To eliminate noise transfers from final amplifiers, the tape control conditions gates to designate intervals in which final amplifier outputs are acceptable. The final amplifier outputs are deconditioned during the IBG to eliminate noise that may be present in the gap.

Skew Registers

- **Skew registers accept data from the final amplifiers.**
- **Bytes are parity checked in the high clip skew register (high clip VRC).**
- **Seven-track gates out of the high clip skew register if data does not have parity error and out of the low clip register if data has parity error.**
- **Nine-track always gates out of the high clip skew register.**

- **Output of skew register is fed to R/W register and LRC register during read operation and only to LRC register during write operation.**

The high clip and low clip skew registers are temporary storage for bytes being read from the tape unit during both read and write operations.

The name "skew" register is derived from their basic purpose: to eliminate skew (time differences) between bits on the lines coming from the tape unit. All the bits in a byte may not arrive from tape at the same time. Skew registers gather all the bits in one byte as they arrive and then gate them out in parallel.

Low-clip pulses from final amplifiers set the low-clip skew register, high-clip pulses set the high-clip skew register. Both the high- and low-clip skew registers contain nine triggers. However, when the seven-track feature is in use on a tape control, only seven of the available nine triggers are used. In read operation, outputs from a skew register set the longitudinal redundancy check register (LRCR) and, the read/write register.

In write operations, skew register output sets only the LRCR.

Because final amplifier tracks can produce low-clip outputs without conditioning the corresponding high-clip output line, the same character might not be loaded in the high and low-clip skew registers. High-low compare circuits compare the character in the high-clip skew register to the low-clip skew register character. If the bit configurations of both characters are not equal, the tape control signals compare (not a C-Compare). High-low compare does not generate an error condition. It is used only to gate the first bit trigger if the high-clip skew register happens to be blank during seven-track read (Figure 331, coordinate 1E). Ordinarily there must be a high-clip bit to set the first bit trigger. A compare indication while the high-clip skew register is blank means that the low-clip skew register must contain at least one bit. If this situation occurs during seven-track read, the first bit trigger is turned on to generate a read clock cycle which processes the information in the low-clip skew register even though the high-clip register is blank.

Skew Register VRC

- **The skew register VRC circuit checks the parity of bytes in the high-clip skew register.**
- **In nine-track operation all bytes should have odd parity except certain check characters.**
- **In seven-track operation the correct parity (odd or even) is specified by bit 3 of the mode register.**
- **In write operation, a skew register VRC error turns on the high-clip VRC check trigger, the data check trigger, and the unit check trigger.**
- **In read operation no errors are signaled. The skew register VRC circuit determines whether the high-clip or low-clip skew register is gated out during seven-track read or write.**
- **Nine-track operation always uses the high clip.**

Each byte of data entering the tape control from the tape unit passes through the high-clip or low-clip skew register. The skew register VRC circuit checks the parity of bytes only in the high-clip skew register. Bytes from the low-clip skew register are used only during seven-track operation when the high-clip register VRC finds incorrect parity. For all nine-track operations or seven-track operations without errors, the high-clip skew register bytes are used.

The skew register VRC circuit looks for odd parity during all nine-track operations. In seven-track operation the skew register VRC looks for odd or even parity according to the condition of mode register 3.

Skew Register Parity Detection

Actual parity checking is done by checking the register positions in three groups of three. The left side of Figure 208 illustrates the division of positions. Each of the three circuits performs the same operation. Outputs of the three circuits, in turn, form inputs for a fourth check circuit. The fourth circuit produces the single output line representing the parity of all nine positions.

Figure 208, positions 5, 6, and 7 are checked by four AND circuits and an OR circuit. Each AND circuit represents one of four possible odd combinations of 5, 6, and 7. The four AND circuits represent:

5 not 6 or 7 6 not 5 or 7 7 not 5 or 6 5, 6, and 7	Any one or all three

Any other combination would have an even number of bits. No bits are considered even parity.

No more than one of the AND circuits should be active at any time. If one of the AND circuits is active, then 5, 6, and 7 must be odd. If none of the AND circuits is active, then 5, 6, and 7 must be even. Any active line to the OR circuit means that 5, 6, and 7 have odd parity.

The same logic applies to the other two groups of positions. Positions 2, 3, and 4 are checked as a group and position P, 0, and 1 are checked as a group. The nine positions have been reduced to three parity check lines. These three lines represent input conditions for the fourth check circuit.

The fourth check circuit performs the same function as the first three except that each input represents three register positions instead of one. The four AND circuits of the fourth group then represent four possible odd parity group combinations.

P, 0, 1 odd 2, 3, 4 and 5, 6, 7 even
2, 3, 4 odd P, 0, 1 and 5, 6, 7 even
5, 6, 7 odd P, 0, 1 and 2, 3, 4 even
All three groups odd

Only one of the four conditions can be active at any given time. Any active condition indicates that the total number of bits (in all three groups combined) is an odd number. If none of the fourth group AND circuits is active, then the total number of bits must be even. Output of the fourth group OR circuit is active (−B level) if the total number of bits is odd. Output of the OR circuit is +B if the total number of bits is even. The output is one line which represents the parity of the high-clip skew register: minus for odd parity, plus for even.

Skew Register Parity Conditions

The skew register parity circuit checks for odd or even parity depending on the situation. For nine-track operation all data bytes are checked for odd redundancy. For seven-track operation the setting of bit 3 in the mode register determines whether the check circuit checks for odd or even parity.

The line called "+B odd redundancy" is generated by mode register circuits. During nine-track operation the "odd redundancy" line remains active. This means that the skew register VRC circuit looks for odd parity (except check characters). If an even parity byte is found, a high-clip VRC is generated.

In seven-track operation the "odd redundancy" line condition is determined by mode register 3. If mode register 3 is on, odd redundancy is active. If mode register 3 is off, the VRC circuit checks for even parity bytes.

The odd redundancy line is held inactive (even redundancy) during seven-track write tape mark operations regardless of the mode register. A seven-track tape mark and its LRCC each contain an even number of bits. The skew register VRC circuit must accept even redundancy, when read checking the seven-track write tape mark, even though mode register 3 specifies odd parity.

CRCC and LRCC Parity depends on whether the number of bytes in the block is odd or even.

Before first check character time, (read checking) each input character from tape steps the binary odd-even character count trigger. After the first check character trigger is set at RDD-36, (before CRCC or LRCC arrive) count trigger stepping is blocked. If the count trigger is left on, the record character count is *odd;* if the trigger is off, the count is *even.*

In nine-track operation, when the CRCC is read, it is parity checked in the high-clip register. If the character count was odd (count trigger on), the CRCC must have even parity; if the count is even, the CRCC must be odd. AND circuit 3F (in the center of Figure 208) adjusts the parity circuit error output at nine track CRC character time. The LRCC should always be odd (for nine-track). If these conditions are not satisfied, a skew register VRC is signaled in seven or nine track:

1. Even parity blocks, LRCC redundancy must also be even. The record character count has no bearing.
2. Odd parity blocks, LRCC must be odd if character count is odd, or LRCC must be even if the count is even.

AND circuit 3G (in the center of Figure 208) adjusts the parity check circuit for error output at seven-track LRC character time. If the above conditions are not satisfied, a skew register VRC is signaled.

Skew Register Parity Errors

The type of operation being performed determines how the skew register parity (VRC) signals are used. In seven- and nine-track write operations the errors turn on a VRC check trigger. In seven-track operations only, the errors gate out the low-clip skew register.

Write op gates the AND circuit that turns on the high-clip VRC check trigger (Figure 208). The write op line is active for write, write tape mark, and erase commands. If the high-clip skew register contains incorrect parity during one of these commands, the error trigger is turned on at read clock 9. At the same time a signal is generated to turn on data check and unit check. During other operations no data or unit checks are generated by the high-clip VRC circuit.

During all nine-track operations the byte in the high-clip skew register is gated out regardless of errors. If an error is detected during seven-track read or write, the low-clip byte is gated out. The low-clip information cannot be gated out during nine-track operation. The only time a low-clip byte is actually used is when a high-clip error occurs during a seven-track operation. Gate out low-clip or high-clip can be generated during read and write but the bytes can enter the read/write register only during read (or read backward). Refer to Figure 331, coordinates 3D.

Read/Write Register

- **During write operations the read/write register accepts bytes from the data register and transmits the bytes to the tape unit via the write bus lines.**
- **During read operations the read/write register accepts bytes from the skew registers and transmits the bytes to the data register.**
- **A vertical redundancy check (VRC) circuit looks for parity errors during read operation and generates parity bits during write operation.**
- **Error correction is done at the outputs of the read/write register during error correction read operation.**

- **R/W register contains nine latches, P through 7, to store data.**
- **R/W register 0 and 1 positions are not used for seven-track operations.**

As its name implies the read/write register is used for both read and write operations.

In write operations, the read/write register accepts data from the data register and/or the translator and transmits characters to the selected tape unit via the write bus. In read operations, the read/write register receives tape unit input data from the high-or low-clip skew register and transfers characters to the translator and/or data register. Although the read/write register contains nine data triggers, only seven triggers are used in seven-track operation. The read/write register is in Figure 401. During read forward operation all characters from the tape except the LRC character pass through the read/write register. During read backward *all* characters from tape pass through the read/write register, however, the LRC character is not gated into the CRC register.

During write operations all data bytes pass through the read/write register. The CRC character also is gated through the read/write register but the LRC character is not.

In a write tape mark operation, normal inputs to the read/write register are not conditioned. A nine-track write tape mark sets read/write register positions 3, 6, and 7. A seven-track write tape mark sets read/write register positions 4, 5, 6, and 7. The tape mark byte is transmitted to the tape unit on the write bus lines.

Vertical redundancy check circuits test each character loaded in the read/write register for parity. The status of mode register 3 determines whether the character is checked for odd or even parity during seven-track operations. In write operations, the read/write register parity circuits generate the output parity line bits (P bit) to the tape unit. A parity error in the read/write register during a read operation sets the read/write register VRC, data check triggers, and unit check.

Actual correction of error bytes is done at the outputs of the read/write register. As an error byte passes through the read/write register in a correction read, the track in error bit in the EPR inverts the track in error position. If a bit had been dropped, it would be replaced; if a bit had been picked up, it would be deleted.

To write the LRC character at the end of a block the write pulse control line sets read/write register positions 0 to 7 and resets the P position. The read/write VRC circuit supplies a write bus P to the tape unit. Active outputs from the read/write register are used to deskew the LRC character at the end of a write operation.

Read/Write Register VRC

- **The read/write register VRC circuit checks the parity of bytes in the read/write register.**
- **In read or read backward, the VRC circuit is used to detect data errors which turn on data check and unit check.**
- **In write operation, the VRC circuit does not detect errors. It generates the correct parity for each byte written on tape.**
- **Parity checking is suspended (by forcing correct parity) during RDD95-158. This is done during read or read backward operations while the track in error is being located.**
- **CRCC parity is checked only during forward read in the R/W VRC.**
- **LRCC parity is not checked in the R/W VRC.**

Each byte that passes through the read/write register is parity checked by the read/write (R/W) register VRC circuit. Output of the R/W VRC circuit performs different functions for read and write. During read or read backward operations, the R/W VRC circuit checks the parity of data bytes from the tape unit. If an error is detected, the R/W VRC error trigger is turned on and data check and unit check are set.

During write operations, the R/W VRC circuit does not indicate errors. If a byte with incorrect parity is detected in the R/W register, the R/W VRC circuit generates a parity bit to be written on tape with the data portion of the byte to replace the stripped P bits. All of the data bytes on tape should then have correct parity. Parity bits are stripped from the bytes at the data register.

R/W Register Parity Checking

Parity checking is done by checking the R/W register positions in three groups of three. The left side of Figure 207 illustrates the division of positions. Each of the three circuits performs the same operation. Outputs of the three circuits, in turn, form inputs for a fourth check circuit. The fourth circuit produces the single output line representing the parity of all nine positions.

In Figure 207, positions P, 0, and 1 are checked by four AND circuits and an OR circuit. Each AND circuit represents one of four possible odd combinations of P, 0, and 1. The four AND circuits represent:

P	not	0 or 1	Any one or all three
1	not	0 or P	
0	not	1 or P	
0, 1, and P			

Any other combinations would have an even number of bits. No bits are considered even parity.

No more than one of the AND circuits should be active at any time. If one of the AND circuits is active then P, 0, and 1 must be odd. If none of the AND circuits is active, then P, 0, and 1 must be even. Any active line to the OR circuit means that P, 0, and 1 have odd parity.

The same logic applies to the other two groups of positions. Positions 2, 3, and 4 are checked as a group and positions 5, 6, and 7 are checked as a group. The nine positions have been reduced to three parity check lines. These three lines represent input conditions for the fourth check circuit.

The fourth check circuit performs the same function as the first three except that each input represents three R/W register positions instead of one. The four AND circuits of the fourth group then represent four possible odd parity group combinations:

P, 0, 1 odd	2, 3, 4, and 5, 6, 7 even
2, 3, 4 odd	P, 0, 1, and 5, 6, 7 even
5, 6, 7 odd	P, 0, 1, and 2, 3, 4 even
All three groups odd	

Only one of the four conditions can be active at any given time. Any active condition indicates that the total number of bits (in all three groups combined) is an odd number. If none of the fourth group AND circuits is active, then the total number of bits must be even. Output of the fourth group OR circuit is active (−B level) if the total number of bits is odd. Output of the OR circuit is +B if the total number of bits is even. The output is one line which represents the parity of the R/W register: minus for odd parity, plus for even.

R/W Register Parity Conditions

In Read: The R/W VRC circuit checks all data bytes in seven- or nine-track mode. In read or read backward operation, the R/W VRC indicates data errors. The correct parity for the R/W register can be either odd or even, depending on the situation. Nine-track data bytes are always odd. Seven-track data can be odd or even, depending on the setting of bit 3 in the mode register.

In Figure 207, just above the test point chart, is a line called +B odd redundancy. This line is developed by the mode register circuits. During nine-track data transmission it remains plus to make the VRC circuits accept odd parity. In seven-track write tape mark or seven-track with mode register 3 off, the odd redundancy line will be minus to accept even redundancy bytes. Seven-track operation with mode register 3 on will be in odd redundancy mode (the same as nine-track).

Output of the odd redundancy adjusting circuit indicates whether the R/W register contains correct parity, either odd or even, according to the operation. The VRC error indication is further conditioned by what part of a block is being processed. Check characters can be odd or even depending on the byte count of the block.

R/W VRC errors turn on the R/W VRC error trigger during read or read backward operations. Normally the status of the R/W VRC circuit is sampled at shift data time. Each data byte in the R/W register is checked when the shift data pulse is active. If an error is detected during read or read backward, the R/W VRC error trigger, the data check trigger, and the unit check trigger are turned on.

In Write: The R/W VRC generates the correct data parity bit to be written on tape but does not indicate errors.

The R/W VRC circuit does not indicate errors during write operations. The R/W VRC circuit generates the parity bit to be written on the tape. Write data bytes in the R/W register do not have a P position. The R/W VRC circuit generates a P bit, when needed, to write correct parity bytes on tape. Output of the R/W VRC circuit drives the write bus P line during write operations. Either write trigger release or write pulse control must be active to gate the R/W VRC line onto write bus P.

During nine-track write, the CRC character is developed in the CRC register. The CRCC to be written passes through the R/W register to the write bus lines, to the tape unit. Since the CRCR automatically generates parity (P bit) according to byte count, the R/W VRC circuit must not modify the CRCC parity. The R/W VRC circuit is forced to indicate correct parity from WDD47 to WDD76, while the CRCC is written on tape, to prevent alteration of the previously generated parity bit.

During data writing, the write trigger release line gates the R/W VRC output to write P bits. The write pulse control line remains active, after the write trigger release falls, to generate a P bit to deskew the LRCC. Write pulse control forces a R/W VRC error. At the same time it gates the VRC error to the P write bus.

When a CRCC Is Processed in nine-track read forward, no shift data pulse is generated. A substitute sample pulse must be generated to check the CRCC parity. In the lower right corner of Figure 207, the CRCC sample pulse is generated at RDD83-86 time of a nine-track read (forward) operation. If there is a parity error in the CRCC, the R/W VRC error trigger, data check, and unit check are turned on.

Parity of a CRCC cannot be checked in read backward because the byte count of the block is not known until after the CRCC has been read. During backward check character time, the read/write VRC circuit is forced to a correct indication by the backward trigger. After the check characters have been read, the backward trigger is reset and normal data checking resumes during the data portion of the block.

If An Error Is Detected during a nine-track read or read backward, an attempt is made to identify the track that contains the error. R/W register parity checking is suspended by a line called "+B gate CRCR to R/W bus" from RDD95 to RDD158 at the end of a block which has an error. During this time the R/W register is used to locate a track in error. The VRC circuit is forced to indicate correct parity regardless of the R/W register status.

During nine-track read or read backward operations, each R/W VRC error generated an input pulse to the 7 position of the error pattern register (EPR). The EPR develops a bit pattern that represents the number of VRC errors and their locations in the block. This error pattern is used to identify a track in error for error correction.

C-Compare Checking

The C-Compare circuit monitors the parity of data as it is transmitted through the tape control unit. Parity of the read/write register is one of the check points used in the C-Compare circuit. Further information can be found under "C-Compare" in the functional units section of this manual.

The odd/even count of R/W register positions is used without considering the condition of the odd redundancy line from the mode register. Output to the C-Compare circuit is called "+B R/W 76543210 odd." This line does not include P bit condition as a factor. Parity of bits 0-7 is derived from P, 0-7 R/W VRC circuits adjusted by four AND circuits at the top of Figure 207. The four AND circuits are needed because of error correction.

In AND circuits 1D and 1E, the two center ones, adjust the parity to leave out the P bit during normal read and write operations. The line "–B EPR to corr" can go minus only during error correction, when a R/W VRC error occurs and the track in error is not the P track. For non-correction operations the two center AND circuits effectively eliminate the P bit from the parity indication line to the C-Compare circuit. If the total P-7 parity is even and there is no P bit, then 0-7 must be even (1D). If the total P-7 parity is odd and there is a P bit, then 0-7 must still be even (1E). If the reverse of either two conditions exists, then 0-7 must be odd. Any active input to OR circuit 1F indicates that 0-7 parity is even.

The other two AND circuits are used only when a byte is being corrected with no bit in the P position of the EPR. Adding or deleting (correcting) a bit in 0-7 will reverse the parity. The two outside AND circuits (1C and 1B) detect when the R/W register 0-7 parity is odd but about to be changed by correction. The output to the C-Compare circuit reflects correct R/W 0-7 parity before that byte is actually corrected.

Data Register

- **During write operations, the data register accepts data from the "bus out" lines and transmits the data to the translator or read/write register.**
- **During read operations, the data register accepts data from the translator or read/write register and transmits the data to the "bus in" lines.**
- **In seven-track write operation, data converter on mode, the data register converts 3 eight-bit bytes to 4 six-bit characters and for read operation 4 six-bit characters to 3 eight-bit bytes.**
- **Data register contains 12 latch positions (1 through 12).**
- **Positions 5 through 12 are used both for seven-track compatibility and nine-track operations (Positions 1 through 4 are only used for seven-track Data Conversion feature).**
- **Data register itself is not parity checked; its data is parity checked prior to entry and after it exits.**

The data register accepts eight-bit data bytes from the interface in write operations and processes eight-bit bytes to the interface in read operations. The data register contains twelve triggers. Only eight of the triggers are used for nine-track operation because the data register does not store parity bits. The four remaining data register positions are used during seven-track data convert operation. In write operations, the tape control checks interface data characters for correct parity and discards the character's parity bit prior to transferring the character to the data register. In read operations, parity bits are not set in the data register, but the tape control checks output data register characters and adds parity bits when necessary to produce correct parity for transmission to the channel.

Nine-track Operation

A tape control not equipped with the seven-track feature can perform read and write operations only in nine-track mode. Each byte transferred to the data register sets data register triggers 5 through 12 corresponding to the bits in the byte; triggers 1 through 4 are not used (Figures 305 and 306). Bytes transfer from data register triggers 5 through 12 unchanged to either read/write register triggers 0 through 7 in write operations or to the interface in read operations. Each byte loaded in the data register must be unloaded before more data can be stored.

Seven-track Operation

Tape controls equipped with the seven-track feature can execute read and write operations in either DC on or DC off mode. All data transfers between the tape

control and the interface are in eight-bit code; six-bit characters are stored in the read-write register to be transferred to the selected tape unit.

DC On Mode (Seven-track only)

When the mode register designates that the operation be performed in DC on mode, the tape control forces odd redundancy mode. In a write operation (using all 12 positions), the data register transfers 4 six-bit data characters to the read-write register for every 3 input eight-bit characters. In a read operation (using all 12 positions), the data register transfers 3 eight-bit characters to the interface for every 4 six-bit input characters from the tape unit. For more information refer to "Data Conversion" in the features section of this manual.

DC Off Mode (Seven Track)

In DC off mode each character transferred to the data register sets data register triggers 5 through 12 corresponding to bits in the character; triggers 1 through 4 are not used. Each character loaded in the data register must be unloaded before more data can be stored.

Translator Active: When the mode register designates DC off mode and specifies that the translator be activated, the data register and the bi-directional translator combine to convert either BCD characters to eight-bit code in read operations or eight-bit code characters to BCD in write operations (Figure 307). To effect the translation in a write operation, the character in the data register transfers to the translator, but bits in data register positions 10 and 12 transfer directly to the read-write register to set read-write register 5 and 7 triggers, respectively. Bits from the translator set read-write register triggers 2, 3, 4, and 6.

In a read operation, the character in the read-write register transfers to the translator, but the "on" states of read-write register 5 and 7 triggers set data register triggers 10 and 12, respectively. Bits from the translator set data register positions 11 and 5 through 9.

Translator Inactive: When the mode register designates DC off mode and specifies that the translator not be activated:

1. The data register transmits the eight-bit interface byte to the read-write register in a write operation, but the read-write register accepts only six of the eight bits. Bits in data register positions 5 and 6 do not transfer to read-write register positions 0 and 1. Bits in data register positions 5 and 6 are lost.

2. The data register transmits eight-bit bytes to the interface in read operations although the read-write register transfers only six bits to the data register. Data register positions 5 and 6 always contain zeros (bit positions 0 and 1 in the character transferred to the interface are zero).

Bus Out Parity Check

- **Nine bus out lines can contain a data byte for write operations, a command byte or an address byte during initial selection.**
- **Information on bus out lines must always be in odd parity.**
- **Bus out parity is checked during write data, command byte, and address byte time, however, a bus out *error* is only indicated during write data and command byte time.**
- **Bus out parity error will set unit check of the status byte.**

Each byte entering the tape control from the bus out lines should have odd parity. The data channel inserts a P bit, when needed, to make the bus out parity odd. Bus out parity check examines every byte to detect errors in transmission from the channel to the tape control.

Parity

Parity checking is done by checking the bus out lines in three groups of three lines. The left side of Figure 200 illustrates the division of bus out lines. Each of the three circuits performs the same operation. Outputs of the three circuits, in turn, form inputs for a fourth check circuit. The fourth circuit produces the single output line representing the parity of all nine lines.

In Figure 200, bus out lines 0, 1, and 2 are checked by four AND circuits and an OR circuit. Each AND circuit represents one of four possible odd combinations of 0, 1, and 2. The four AND circuits represent:

2	not	0 or 1	Any one or all three
1	not	0 or 2	
0	not	1 or 2	
0, 1, and 2			

Any other combination would have an even number of bits. No bits are considered even parity.

No more than one of the AND circuits should be active at any time. If one of the AND circuits is active, then 0, 1, and 2 must be odd. If none of the AND circuits is active, then 0, 1, and 2 must be even. Any active line to the OR circuit means that 0, 1, and 2 have odd parity.

The same logic applies to the other two groups of bus out lines. Positions 3, 4, and 5 are checked as a group and positions 6, 7, and P are checked as a group. The nine bus out lines have been reduced to three parity check lines. These three lines represent input conditions for the fourth check circuit.

The fourth check circuit performs the same function as the first three except that each input represents three bus out lines instead of one. The four AND circuits of the fourth group then represent four possible odd parity group combinations:

6, 7, P odd	0, 1, 2, and 3, 4, 5 even
3, 4, 5 odd	0, 1, 2, and 6, 7, P even
0, 1, 2 odd	3, 4, 5, and 6, 7, P even
All three groups odd	

Only one of the four conditions can be active at any given time. Any active condition indicates that the total number of bits (in all three groups combined) is an odd number. If none of the fourth group AND circuits is active, then the total number of bits must be even. Output of the fourth group OR circuit is active (−B level) if the total number of bits on the bus out is odd. Output of the OR circuit is +B if the bus out bits are even (error condition).

Parity Error Trigger

Even parity represents an error condition. The even parity condition gates two AND circuits which can turn on the bus out parity error trigger. The top AND circuit samples the parity condition each time a command is received during initial selection. The bottom AND circuit samples the parity of each byte of data during write operation. The bottom AND circuit also samples the track in error byte transmitted during a no-op request (request TIE) operation.

If an even parity byte (command or data) is received, the bus out parity error trigger turns on. Output of the trigger places a 2 bit in sense byte 1. Bus out parity error or command parity error will turn on unit check (status bit 6) to signal an error condition.

Address Parity

The parity checking circuit is active all the time but the error trigger can be turned on only during initial selection (command out), write operation or no-op request. Output of the parity check circuits is used during address out time of initial selection even though the error trigger cannot be turned on at that time. The parity check circuit gates the "control unit addressed" circuit if the address byte contains the correct (odd) parity. If the address byte contains even parity, the control unit will not accept it.

Shift Control

- **All data processed by the tape control unit requires shift data pulses.**
- **Shift data pulses are generated only during write, read, or read backward.**
- **During write operations, shift data pulses control data transfers from the channel interface to the data register and from the data register to the read/write register.**
- **During read or read backward, shift data pulses control data transfers from the read/write register to the data register and from the data register to the channel interface.**
- **Error correction circuits require shift data pulses to develop the cyclic redundancy check character (CRCC) and error pattern.**

"Shift data" is the controlling pulse for data transfers in the tape control unit. Write, read, and read backward require shift data pulses to synchronize data movement. Shift control circuits are shown on Figure 408. Figure 600 illustrates shift data timing in write and read operation.

The heart of shift control is two singleshots: one 600 nanoseconds, the other 1 microsecond. A small timing chart at the bottom of Figure 408 shows the time relationship of the two singleshots. Three distinct pulse timings are provided by using both singleshot outputs separately and combining them to form a third (ss-2) pulse.

Output of the 600 ns singleshot drives the tape demand circuit and the data shift circuits. Output of the 1 μs singleshot drives the overrun and service in tag circuits. The combined or ss-2 pulse is used for resets because it is the last 400 ns of the 1 μs pulse. Data should have been completely transferred (between the read/write and data register) before ss-2 rises.

The key to shift data operation is the "start shift" line which drives both the 600 ns and the 1 μs singleshots. A positive shift on the "start shift" line starts both singleshots at the same time. Timing of the start shift pulse determines when data will be moved through the tape control unit.

Start shift is generated by an AND circuit with 8 legs (Figure 408). During normal operation the controlling leg of the AND circuit is the R/W control trigger. Six of the eight legs establish conditions under which data is allowed to move. Data will not be processed if the first character tape mark trigger is on or if the transfer data line is down. Set overrun, set R/W control, service in trigger, or service out can block the start shift pulse.

The circuits which set the R/W control trigger also block the start shift pulse. The start shift pulse will not begin until the originating signal falls. In read operation, start shift will begin at the end of RC reset. In write operation start shift occurs when WC-2 falls or actually WC-3.

Data Transmission

The read/write control trigger indicates when the read/write register requires a data shift. During write operation the R/W control trigger is set at WC-2. The following shift pulse loads data into the read/write

register to be written on tape at WC-6-10. For read operation the R/W control trigger is set at RC reset time, after a byte has been read from tape into the read/write register. A shift pulse then moves the byte into the data register, making room for the next byte in the read/write register.

During the data transmission portion of an operation, the output of the 600 ns singleshot provides gating pulses for data. The line is called +B shift data. The shift data line also gates the demand circuit to turn on service in. The byte counter is stepped by the same shift data line only during data conversion (optional features).

Four gates, controlled by the shift data line, transmit data back and forth between the read/write register and the data register. The four gates are:

Rd shift byte 2 or 4	R/W reg to data reg
Wr shift byte 2 or 4	Data reg to R/W reg
Rd shift byte 1 or 3	R/W reg to data reg
Wr shift byte 1 or 3	Data reg to R/W reg

The last two gates are active only for data conversion (lower right corner of Figure 408, and Figures 309 and 310).

Six other data gates are controlled indirectly by shift data. These six gates control transmission of data from the bus out lines to the data register (write) and from the data register to the bus in lines (read). The six gates are "read bytes 2, 3, and 4" and "write set bytes 1, 2, and 3."

Read byte 4 and write set byte 3 are used for standard operation (Figures 305, 306, 307, and 308). The other four gates are used only in data convert operations.

"Service in" is the line which controls the six data register gating circuits. Service in is controlled by shift data. Shift data generates a demand pulse that turns on the service in trigger. For read operations, service in gates the data register onto the bus in lines with read byte 4. After a delay of about a microsecond, service in notifies the data channel that the byte is available.

During write operation, service in tells the channel that a byte is needed. When the channel has placed the new byte on the bus out lines, service out rises (from channel). Service out and service in (delayed) are AND'ed with write to bring up write set byte 3. Service out (delayed) resets the service in trigger in preparation for the next shift.

Ending Conditions

The R/W control trigger is reset each time a shift data pulse occurs. The R/W control trigger was turned on to indicate that a shift was needed. When the data shift accomplishes its purpose, the control trigger must be reset to prepare for the next byte of data.

At the End of a Write Operation, shift pulses are stopped by the stop data transfer trigger. Each clock cycle attempts to set the trigger but the trigger is held reset until write op and the stop trigger are on at the same time. The stop trigger is set when the channel responds to service in with command out.

The shift data line, to turn on the stop data transfer trigger, is used during data conversion operation. The turn on circuit for normal write operations is activated by the write clock D trigger. The D trigger is on during the last half of the write clock (8-15). A stop during this time will immediately bring up the stop data transfer line to block shift data.

At the End of a Read Operation, shift pulses are stopped by blocking the circuit which turns on the R/W control trigger. "RC reset not write" is blocked by read op and first check character (bottom of Figure 206). The same circuit blocks shift pulses during check character time of a read backward, while the "backward trigger" is on.

The data channel can terminate a read or read backward operation before all the data has been read. When a service in is sent to the channel the expected response is service out. If the channel (program) decides to end the operation, a command out is sent in place of service out. Service in and command out turns on the stop trigger. The stop trigger, in turn, blocks demand but does not block shift data.

The remaining bytes in the block are read and shifted as far as the data register but no more attempt is made to transfer data across the interface.

Unusual Shift Conditions

A shift data pulse can be generated by the circuit which sets "data convert check." This circuit provides an extra shift pulse when data conversion is performed on data which does not contain a multiple of four characters on tape. Further explanation can be found in the section on "Data Conversion Feature."

It is possible for the shift control circuits to generate two shift data pulses in the same read clock cycle. If the first byte of a block happens to have the bit configuration of a tape mark, the tape control will retain that byte to see if it is a true tape mark. If the next byte proves that the configuration is data, the tape control must transfer both bytes to "catch up" before the third byte arrives. The "extra" shift pulse is generated when the first character tape mark trigger is reset, allowing a start shift pulse at RC-0. A regular start shift will be generated at RC reset of the same clock cycle.

The "first character tape mark trigger" line was able to generate a shift pulse in this situation because the R/W control trigger remained on from the first read clock cycle. "RC reset not write" turned on the R/W control trigger in the first read clock cycle but since

the first byte appeared to be a tape mark the first character tape mark trigger blocked the start shift pulse. Because no shift pulse was generated, the R/W control trigger remained on to "remember" that the read/write register contained a byte. When the second byte proved to be data, the first character tape mark trigger was reset at RC-0, which allows the extra shift pulse.

A similar situation can arise during backward read when there is no CRCC (nine-track) or when there is no check character at all (seven-track). If the control unit determines that bytes are being read at data spacing instead of check character spacing, an extra shift pulse must be generated to catch up. The read/write control trigger is set at read clock 0 and again at read clock reset. This condition is explained in more detail in the read backward operation.

During read or read backward operations, a "lost character" indication will generate a start shift. When loss of a complete character is detected, the shift pulse is needed to keep error correction circuits in step. See lost character under "Functional Units."

Overrun

- **Overrun basically checks the capability of the channel to service data requests from the control unit—both read and write.**
- **During a read operation, the overrun is detected if channel response is two clock cycles late.**
- **During a write operation, the overrun condition is detected if the channel response is one clock cycle late.**
- **Overrun condition will inhibit any more data requests to the channel.**
- **An overrun error will set "unit check" in the end status byte.**

The overrun circuit monitors data transfer timing between the control unit and the channel. If the read/write control trigger or shift data indicates that the channel is not accepting or sending data at the same rate as the tape control unit, an overrun error is signaled. An overrun condition can also occur if the channel responds late with a command out prior to the beginning of the next write clock cycle. An overrun condition is relative to both read and write operations. The effect of an overrun error is to set unit check and to inhibit any more data transfer between the channel and the tape control unit. Figure 600 illustrates shift control timing and overrun timing. The overrun conditions are indicated by the dotted lines.

Write Overrun

In a write operation, there are two conditions which constitute an overrun error. The first condition is the assumption that the channel fails to respond with service out to service in which was sent by the tape control unit. This is the overrun condition shown on the timing chart of Figure 600. Because the channel did not respond with service out, the service in trigger was not reset on this clock cycle. This is shown on Figure 408. Thus, no "start shift pulse" could be generated on the next write clock 3 time. The read/write control trigger did turn on at WC-2 of the next clock cycle but there is no shift data pulse to reset it.

With reference to Figure 408, the output of the R/W control trigger attempts to turn on the set overrun trigger every cycle. Normally, the set overrun trigger will not latch on because the R/W control trigger is reset by shift data before the next write pulse (WC-6-10). Because the output of set overrun is taken from the AND half of the trigger, no output is available until the "latch back" (write pulse) is brought up.

If the R/W control trigger is still on when the next write pulse arrives, the set overrun trigger will turn on the overrun error trigger which in turn sets the unit check and sets the stop trigger. An overrun error condition occurred because the service in/service out response did not supply a new byte in time for the following write clock cycle.

The second type of overrun error in a write operation can be caused by a "late stop" condition. Normally, a stop (command out from the channel) will occur during the last half of a write clock cycle. The stop data transfer trigger can be turned on while the write clock D trigger is on from 8 to 15 time. The write clock will reset and stop when it reaches WC-14-15. If the stop occurs after WC-14-15, the write clock must continue to run and complete another cycle. A stop which occurs after WC-14-15 time, but before the shift data pulse (WC-3) of the following cycle is called "late stop." During this time, it is too late to stop the write clock so a shift data will occur after the stop trigger turns on. The stop data transfer will turn on at shift data time in this case. Overrun is signaled because the extra shift pulse occurred after data should have stopped.

The circuit to detect the late stop condition is in the lower left portion of Figure 408. The "late stop" AND circuit is the key point. If this AND circuit is made a "late stop," overrun error is signaled.

The top leg of the AND circuit is "stop data transfer" which can only come up after the "stop" trigger turns on and either a shift data or WC-D time. The two center legs are write clock timings, not WC-D and WC-A. This limits the set overrun to write clock 1 and 2 or 5 and 6. When a "late stop" condition occurs, an overrun error is turned on at WC-5-6. Because stop data

transfer cannot turn on until shift data time (WC-3) overrun is not turned on at write clock 1 or 2.

The gate overrun trigger is set to remember that a stop signal has not been received between write clock 8 and 15 time. Thus, there is a possibility that a late stop may occur. The gate overrun trigger is set by the WC-D trigger (8-15 time) and is only reset if a stop signal occurs during write clock 8 to 15 time. Notice the reset to the gate overrun trigger is activated by the setting of another trigger at 5D, 5E. This trigger will be set if the stop trigger is activated during WC-D time which is 8 through 15 time. If a stop condition occurs while the WC-D trigger is on, the gate overrun trigger will be held reset to block a late stop overrun error.

Each time a WC-D pulse occurs without a stop, the late stop AND circuit has the two lower legs conditioned. A late stop (command out after WC-15) followed by a data shift pulse at WC-3 will turn on stop data transfer to condition the top leg of the "late stop" AND circuit. When WC-A comes on at write clock 5-6 time, the gate overrun AND circuit turns on the overrun error trigger.

Read Overrun

The overrun circuits perform a timing check during read and read backward operations. The overrun circuits check the ability of the channel to accept data bytes from the control unit at the same rate that the control unit is accepting the characters read from tape. If channel cannot accept at the same rate, the overrun error indication will be set which in turn sets the unit check and will inhibit any more data transfers to the channel. Once again, on Figure 600, an overrun condition is indicated by the dotted lines on the timing chart. The circuits for overrun recognition are on Figure 408.

The tape control will allow the channel more time to respond to service in during read or read backward than in write operations. Since the control unit has the storage area to store one data character in a data register and another in the read/write register, it will allow the channel more time to respond to the service in request during the read and read backward operations. The timing chart on Figure 600 shows the effect of an overrun condition caused by the channel's failure to respond with service out on the first service in request. On the first read clock cycle all conditions occur normally; however, due to the absence of the service out response, the service in trigger will not be reset. Thus, on a second data byte, or read clock cycle, read/write control trigger is set but a shift data pulse is not generated because service in is still active. The first data byte is sitting in the data register while the second data byte is located in the read/write register.

In Figure 408, it will be seen that the active state of the read/write control trigger will retain a constant set to the set overrun trigger. Therefore, on the third read clock 6 pulse, because read/write control trigger is still active, the set overrun trigger will be allowed to latch back and indicate an overrun error. Thus, setting unit check and setting the stop trigger will inhibit any more data transfers. Immediately upon setting the stop trigger, the tape control unit can no longer generate any more tape demand pulses to activate the service in to the channel.

During read and read backward commands, the overrun error condition cannot be generated until the third read clock cycle. During the first read clock cycle of any group of three, service in is activated and shift control functions normally. During the second read clock cycle, read/write control trigger is again set but a shift data pulse cannot be generated while service in is still active (no service out to reset service in). During the third read clock cycle, since TCU cannot accommodate another data byte, overrun error becomes active at read clock 6 time.

C-Compare

- **C-Compare check circuit checks parity changes of the data byte between bus out and the R/W register on write operations.**
- **C-Compare will check the parity changes of the data byte between the R/W register and the bus in lines during a read/read backwards operation.**
- **A C-Compare check is performed on all seven- and nine-track read/write operations *except* seven-track "translate on" mode.**
- **A C-Compare error will set data check and unit check.**

C-Compare check is a data check not an equipment check. Bytes transmitted through the tape control in read or write operations should not change parity except during translate mode. C-Compare detects parity changes during data transmission through the control unit. Figure 212 illustrates C-Compare checking circuits.

During write, each odd parity character from the bus out lines complements the data parity check trigger. The same character should again complement the trigger when it causes odd parity in the read/write register. During read, the read/write register and bus in parities are used. In read or write the data parity check trigger should be off after each character cycle. If it remains on, the C-Compare and equipment check are turned on. During data conversion the data parity check trig-

ger must be off only when the byte counter completes a conversion cycle. The byte counter remains reset during non-convert operations. The line "−B byte counter A or B trigger" will remain + for normal read and write operations. (See section on Data Conversion Feature). Translate mode blocks setting of the C-Compare trigger. This is done because the total number of bits in a byte may change during translation, so C-Compare cannot be checked when the translator is operating.

The data parity check trigger output is sampled each time a byte is transmitted, except during conversion operations. In write operation, data parity check trigger is sampled by service in and not service in delayed (the next data byte). In read or read backward it is sampled by service out. If the data parity check trigger is on at sample time, the C-Compare error trigger is set.

Read C-Compare

During read or read backward operations, the C-Compare circuit monitors data transmission from the R/W register through the data register to the bus in lines. Each odd parity byte in the R/W register complements the data parity check trigger (Figure 212). When the same byte reaches the bus in lines, no P bit is generated because the byte already has odd parity. Not bus in P complements the data parity check trigger back to its original state (off). If the parity of the byte changed during transfer, the data parity check trigger would have remained on to signal an error.

In Figure 212, two AND circuits (upper left) sample the parity conditions. At shift data time the R/W register parity is checked. Bus in parity is checked when service in delayed indicates that the byte is on the bus in lines. When the channel responds with service out, the data parity check trigger should be off.

Write C-Compare

During write operations, the C-Compare circuit monitors data transmission from the bus out lines through the data register to the R/W register. Each odd parity byte (no P bit) on the bus out lines complements the data parity check trigger. When the same byte reaches the R/W register, the data parity check trigger is complemented back to its original condition (off). Even parity bytes do not complement the data parity check trigger at either point. With odd or even bytes, the data parity check trigger will be left on if the parity of the byte changes during transmission.

Seven-Track Write C-Compare

During seven-track non-translate, write operations bus out 0 and 1 are not transmitted to the R/W register. The data channel, however, can place bits on bus out lines 0 and 1. For this reason bus out P, 0, and 1 must be checked instead of only the P bit to determine if 2-7 are odd or even. If 0 and 1 remain off, the P bit indicates 2-7 parity. If 0 or 1 contains a bit, the circuit is adjusted to represent the parity of 2-7, the portion transmitted to the R/W register.

Output of the adjusted bus out parity circuit is used for the C-Compare check. At service in/service out time of a write command the adjusted parity can complement the data parity check trigger if the byte sent to the R/W register is odd.

When the byte is transferred to the R/W register, an SS-2 pulse gates the R/W VRC output to the data parity check circuit. If the byte is still odd (no parity change) the data parity check trigger will be complemented back to its original state (off). If the parity of the byte changed during transfer, the data parity check trigger would have remained on to signal a C-Compare error.

During nine-track or seven-track convert, only the P bit is needed for the C-Compare circuit because all positions of the bus out lines are in use. A P bit indicates that lines 0-7 are even. No P bit means 0-7 are odd.

Error Correction

- **In nine-track write operations, the CRCR generates a CRCC to be written at the end of the block.**
- **In nine-track read operation the CRCR recalculates the CRCC. When the original CRCC is combined with the new CRCC, the result should be a "match pattern" of 111 010 111 in the CRCR.**
- **If read errors occur, the EPR contains an error pattern. The error pattern is used in conjunction with the faulty character in the CRCR to locate which tape track was in error (TIE).**
- **The TIE information is sent to channel in a sense byte.**
- **After tape is repositioned, the TIE bit is sent from channel back to the EPR in TCU.**
- **As the data is read the second time, each error byte is corrected by inverting the bit in the track in error position indicated by the EPR bit.**
- **The corrected bytes are checked to see that correction is done properly.**
- **Refer to Figures 402, 403, 404, and 509.**

Philosophy of Error Correction

Error correction circuits provide the ability to correct the majority of errors encountered in normal tape operations. Tape control error correction assumes that the data on tape was written correctly and any errors detected are a result of tape damage or read failure. No attempt is made to correct the tape. Correction is performed only on the data sent to the channel inter-

face. Corrected bytes are automatically checked to be sure the correction is done properly.

Not all errors are correctable. The 2403/2803 error correction circuits can correct almost any combination of errors as long as they all occur in the same track within a nine-track block. If a block contains errors in more than one track, they cannot be corrected. The key to error correction is determination of which track contains the errors. Once the track in error (TIE) has been located, actual correction becomes a simple matter.

Most of the error correction circuits and procedures are concerned with finding the TIE. Locating the TIE is the basic purpose of having two check characters at the end of each block instead of the previous single LRC character. The added CRC character provides part of the information needed to determine a TIE. The remainder of the information is supplied by an "error pattern" generated in the error pattern register.

As a block is being written on tape, each byte is shifted (in parallel) into the CRC register to generate a CRC character. At the end of the block the CRCR is shifted once, without inputs, then moved into the read/write register to be written on the tape.

When a block is read from tape, each of the bytes is again shifted into the CRC register. At the end of the block the CRC character from the tape is also shifted into the CRC register on top of the data bytes. If no errors have occurred, the result will be a "match pattern" (111 010 111) in the CRC register. The same process is used for read backward except that the CRC character is shifted into the CRC register before the data.

A match pattern should still be produced if no errors occurred. Errors in reading will cause an erroneous pattern in the CRC register at the end of the block, and generate a CRC error. The CRC error sets data check and unit check triggers.

While a block is being read, any R/W VRC errors detected are recorded in the error pattern register. If no errors are found, the error pattern register (EPR) remains blank. At the end of an error block, the error pattern in the EPR and the erroneous CRC character in the CRC register are used to determine a track in error.

The process of looking for the TIE occurs about the same time as the LRC character is read from tape. The LRC character is not needed for the TIE calculations so it is gated to the LRC register without disturbing the error correction circuits.

TIE searching requires the use of the EPR so the error pattern that was calculated must be stored elsewhere. At RDD95-98 time, the error pattern in the EPR is moved into the read/write register. At RDD99, the EPR is reset; then a P bit is forced into the EPR at RDD103 time. This P bit will act as a track indicator during the searching process.

The erroneous CRC register contents must be matched against the error pattern now stored in the read/write register. A group of exclusive OR circuits at the outputs of the read/write register are used to search for an "all zero" match. The matching process is not a direct "one for one" comparison. Positions P, 0, 1, 3, 5, 6, and 7 of the CRCR are complemented as they are gated to the exclusive OR circuits. The remaining positions (2 and 4) are gated without inversion. If the exclusive OR of the error pattern (read/write register) and the complemented CRC character equals all zeros, the track in error has been found.

If a zero match is produced on the first try, the P bit in the EPR indicates the track which contains the error (TIE). If a zero match did not occur on the first try, the CRCR and EPR are shifted and again matched. (A shift pulse with no input data will move the CRCR and EPR information into the next position.) The P bit in the EPR then becomes a 0 bit. The byte in the CRCR will be shifted according to the exclusive OR feedback paths described under CRC character generation. After the shift pulse the error pattern in the read/write register is matched against the new CRCR byte. If a zero match occurs, the 0 bit in the EPR indicates the TIE. If a zero match *does not* occur, the CRCR and EPR are shifted and matched again. Shifting and matching continue until all the tracks have been tried unsuccessfully or a zero match occurs.

When a zero match occurs, the found track trigger is turned *on* to indicate that the TIE has been found. If the found track trigger remains *off* after all tracks have been tried, no TIE can be found. Either condition will be transmitted to the data channel in sense byte 3, of the sense operation. A single bit in the sense byte indicates which track contains the error. If no TIE is found, bits 6 *and* 7 are forced into the sense byte.

After the data channel receives the TIE information via a sense command following the read command, it is up to the program to decide if the error will be corrected or ignored. If the error is to be corrected, the tape must be repositioned to read the same block again in the identical direction (FSR/BSR). During repositioning the TIE bit in the EPR will be lost. The channel must replace the TIE bit before correction can take place. A command called "request TIE" transmits a byte containing the TIE bit from the channel to the EPR. Request TIE also turns on the "correcting" trigger to indicate that the next read operation should be a correction read. With the correcting trigger on and a TIE bit in the EPR, the tape control is ready to correct the block on the next read operation. Bytes are read from tape and checked by the tape control in the same way as a non-correction read. If no read/write VRC errors are

detected the bytes are processed to the channel without change. Each time a R/W VRC error is encountered, that byte is "corrected."

Correction of bytes assumes that the R/W VRC errors are caused by the same track that caused the errors on the first read pass. Each time a R/W VRC error is found during a correction read, the bit position indicated by the EPR bit is complemented on the output of the R/W register. After complementing of the error bit, the byte should be back to its original configuration. For example, if a byte lost a 5 bit during the reading error correction circuits should have found the 5 track to be the TIE. The request TIE operation should have placed a 5 bit in the EPR for correction. On the correction read that same byte will probably lose the 5 bit again. The missing 5 bit will cause a R/W VRC error during correction. Since there is a TIE 5 bit in the EPR, the 5 position of the byte will be complemented. The 5 bit lost during reading will be replaced.

The same process can be repeated for as many bytes as necessary provided all the errors are in the same track. A bit picked up can be deleted or a lost bit can be replaced.

To be sure that the correction is done properly, CRC is again computed from the corrected bytes. If all the bytes are restored to their original condition, the CRC register should produce a match pattern (111 010 111) when the original CRC character from tape is shifted into the CRCR. If a match pattern is not produced, an error has occurred in the correction process or in one of the other tracks. When no match pattern is developed, a CRCR error and data check/unit check will result.

Cyclic Redundancy Check Register

The cyclic redundancy check register (CRCR) consists of nine binary triggers and their associated circuits. Positions are designated P, 0 through 7. Characters are shifted into the CRCR from the read/write (R/W) register to develop a cyclic redundancy check character. Output of the CRCR returns to the R/W register. In the following chart, inputs on the left are exclusive OR'ed with the present status of the CRCR positions in the center to determine which CRCR positions will be changed at shift pulse time. In the case of CRC register positions P, 0, 1, 6, and 7, any one and only one of the two conditions active (R/W input and one CRCR position) will set the corresponding trigger in the CRCR to a one. "Active" implies a one (1) bit. In the case of positions 2, 3, 4, and 5, any one or all three conditions active (input and two CRCR positions) will set the corresponding trigger to the "one" state. This is accomplished by first exclusive OR'ing two of the three conditions and then exclusive OR'ing that result with the third condition.

READ/WRITE REGISTER INPUTS		OE	CRCR POSITION			=	CRCR CHANGED
READ BACKWARD	WRITE AND READ FORWARD						
7	P	OE	7			=	P
6	0	OE	P			=	0
5	1	OE	0			=	1
4	2	OE	1	OE	7	=	2
3	3	OE	2	OE	7	=	3
2	4	OE	3	OE	7	=	4
1	5	OE	4	OE	7	=	5
0	6	OE	5			=	6
P	7	OE	6			=	7

Circuit Description

As each character passes through the R/W register, it is shifted in parallel into the CRCR. Between the R/W register and the CRCR is a group of exclusive OR, (OE), circuits (Figure 12). The R/W register outputs are one set of inputs to the OE circuits; CRCR feedback forms the other inputs. Data from the R/W register passes through the OE circuits and is gated into the CRCR triggers by a shift pulse. The character in the CRCR after each shift is a combination of the R/W register character and the previous CRCR characters. If the CRCR is blank, the first character passes unchanged through the OE circuits into the CRCR triggers. When there are bits in the CRCR, a character from the R/W register will be modified.

The OE circuits are arranged in two levels (Figure 12). Characters from the R/W register are reversed at the input of the OE first level for read backward. Position 3 is the same for forward and backward.

Feedback from the adjacent CRCR position forms the other first level input. Positions 2, 3, 4, and 5 have three-way OE circuits. Outputs from these OE circuits will be active when any one or all three inputs are active (an odd number of inputs). Notice in Figure 12 that CRCR positions P, 2, 3, 4, and 5 each have an input from position 7. CRCR 7, therefore, is a key factor in generating the CRC character.

The second level of OE circuits are alike for all nine positions. First level outputs feed one side of the second level inputs. The other second level inputs are from the CRCR triggers in the same bit positions.

Outputs of the second level OE circuits gate shift pulses, into the binary trigger inputs. The word binary is important. An active output from the second level OE will complement the corresponding trigger at the fall of the shift pulse; if the trigger is on, it will be turned off and vice versa.

Error Pattern Register

The EPR assumes three different roles in error correction operations. Each will be described as all phases of error correction are discussed. However, the pri-

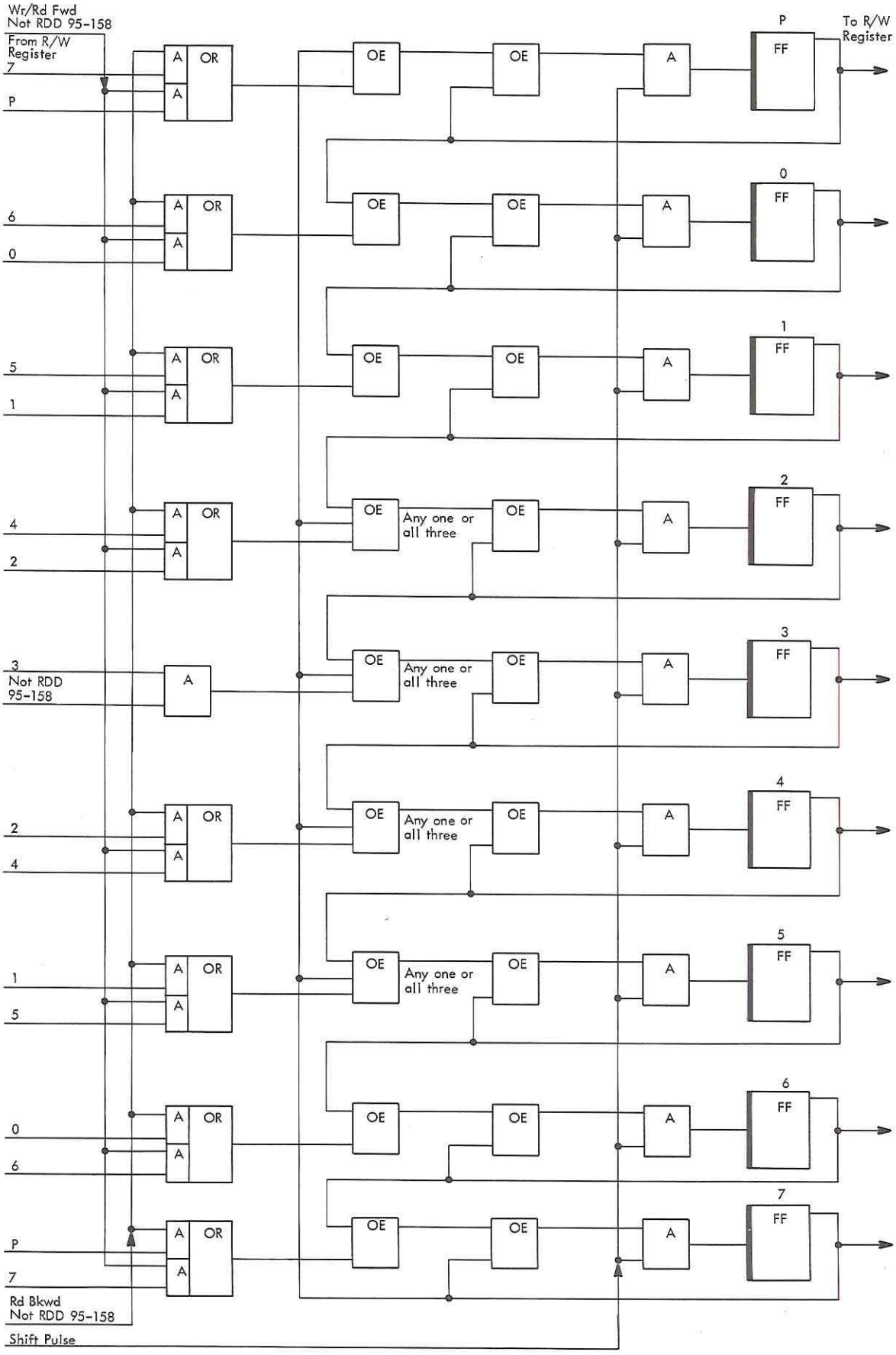


Figure 12. Cyclic Redundancy Check Register

mary purpose of the EPR is to compile all R/W register vertical redundancy (VRC) errors in the form of an error pattern for each record. This compiled error pattern is used to determine which track caused the failures. If there were no R/W VRC errors encountered, the EPR will remain blank.

The EPR consists of nine binary triggers and their associated circuits. Positions are designated P, 0 through 7. The operation of the EPR is similar to that of the CRC register except that it has only one external input. This input affects only the 7 position. The EPR will be shifted each time that the CRC register is shifted during read and read backward operations.

In the following chart, the only input on the left, R/W VRC error, is exclusive OR'ed with EPR position 6 to determine if EPR 7 will be changed at shift pulse time. Consider a R/W VRC error as an "active" input to the exclusive OR circuit. The present contents of positions 1, 2, 3, and 4 are exclusive OR'ed with position 7 to determine if positions 2, 3, 4, and 5 (a "1" input) will change at shift pulse time. Contents of positions 7, P, and 0 are shifted unchanged into positions P, 0, and 1 respectively.

INPUT	EPR POSITION		=	EPR CHANGED
	7		=	P
	P		=	0
	0		=	1
	1	OE 7	=	2
	2	OE 7	=	3
	3	OE 7	=	4
	4	OE 7	=	5
	5		=	6
R/W VRC Error		OE 6	=	7

Circuit Description

The EPR is similar to the CRCR in that it has two levels of OE circuits that gate shift pulses into the binary triggers (Figure 13). One input to the first level OE circuits for positions 2, 3, 4, and 5 comes from trigger 7 output. Position 7, therefore, is a key factor in determining the error pattern just as it is in the CRCR. The other inputs to the first level OE come from the adjacent positions.

First level OE for position 7 is the entry point for VRC errors from the read/write register. As a record is read, each redundant character will condition one leg of the OE circuit. If 6 and 7 are off, position 7 will be turned on at the fall of the shift pulse. If EPR 6 and 7 are both on, then position 7 will be turned off at the fall of the shift pulse. Active outputs from second level OE circuits allow the shift pulse to complement the corresponding triggers.

Cyclic Redundancy Check Operation

The appendix of this manual contains two slide charts to illustrate operation of the CRCR and the EPR. Data bytes used in the slide chart examples are the same as the bytes used in the following text examples. Two slides are provided for the CRCR chart (one for write operation and one for read operation). The slide charts can be used to follow data through the circuits while reading the text.

To use the slide charts:

1. Remove the slide chart and overlay from the manual.
2. Place the overlay on top of the slide chart and align it.
3. Each increment of overlay movement after the first move represents one shift pulse.
4. The condition of each circuit element can be observed during pauses between shifts.

When using the CRCR chart, data bytes or the CRC character enter the circuit through the series of AND/OR circuits on the left. Bits from the input circuits condition the OE circuits along with feedback from each trigger position. Outputs of the OE circuits are gated into the trigger inputs when a shift pulse occurs. After moving the overlay one increment (one shift), the new value will appear in the CRCR and the next input byte appears at the input gates.

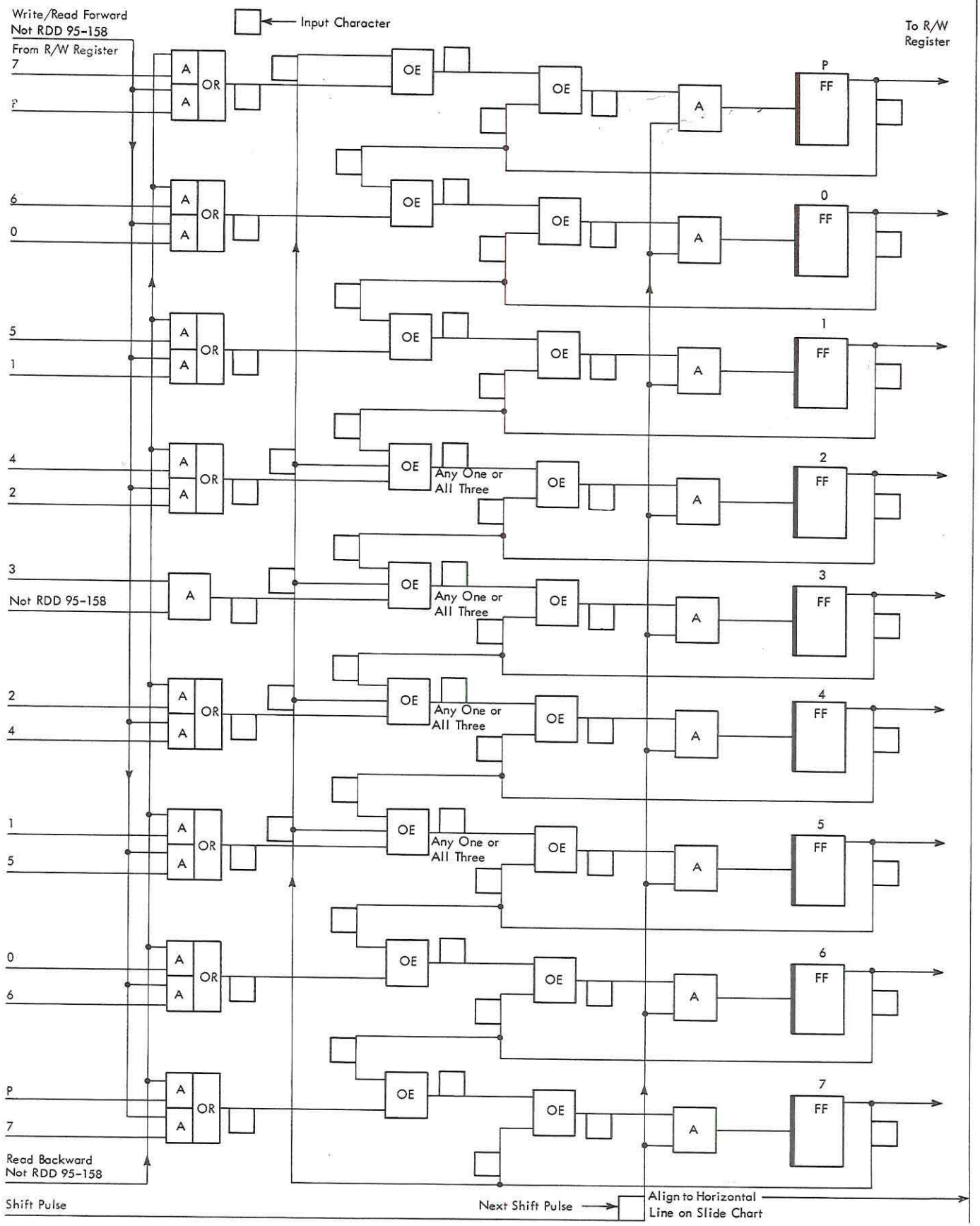
The EPR slide chart illustrates the cyclic nature of the shift pattern. Only one bit is shifted into EPR position 7. The entire sequence of patterns in the register results from the one input bit on the first shift. Note that after 18 shift pulses the pattern has cycled back to the same condition as after the first shift (bit in position 7 only). If no more bits (VRC errors) are gated into the register, the cycle keeps repeating until shifting stops at the end of the block. Each additional input bit would alter the sequence at that point.

Three flow charts show the overall relationship of the CRC operation in the tape control operations: Figure 14 shows a write operation; Figure 15 shows a read operation. If correction is attempted, see Figure 16.

Figure 509 shows error correction circuits. The three sections of Figure 509 correspond to Figures 14, 15, and 16. Figure 509 contains more detailed machine circuit information for use during machine servicing.

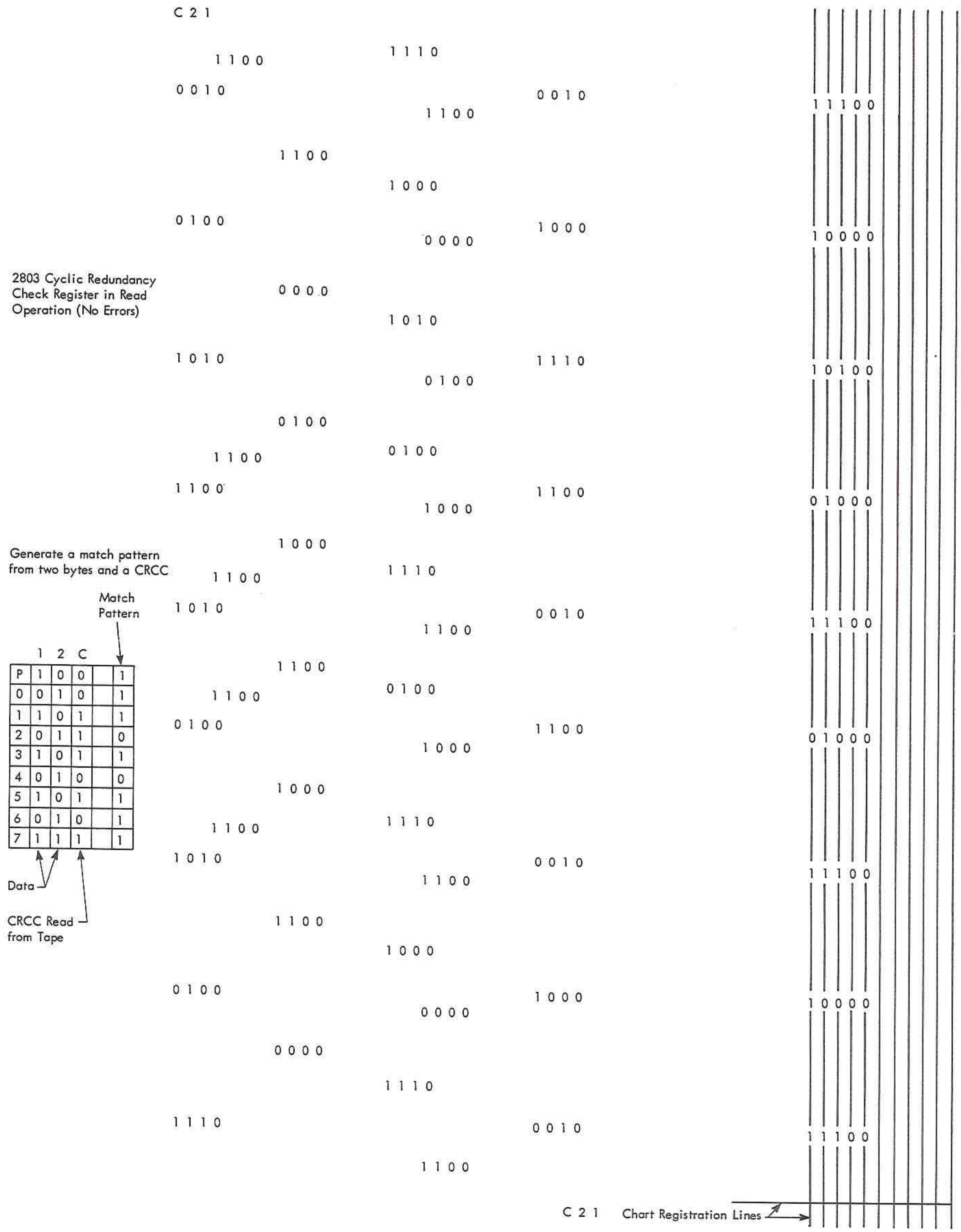
Write (Figure 14)

During write, each character is shifted in parallel from the R/W register into the CRCR. Because of the shift pattern of the CRCR, a check character is developed which can be written on tape at the end of the record. After all the data bytes have been shifted into the CRCR, one more shift pulse is generated (at WDD35-42) with no inputs. The extra shift at the end of a write operation will make the total number of shifts equal to the total number of data bytes plus one additional shift. The CRCR character (first check char-



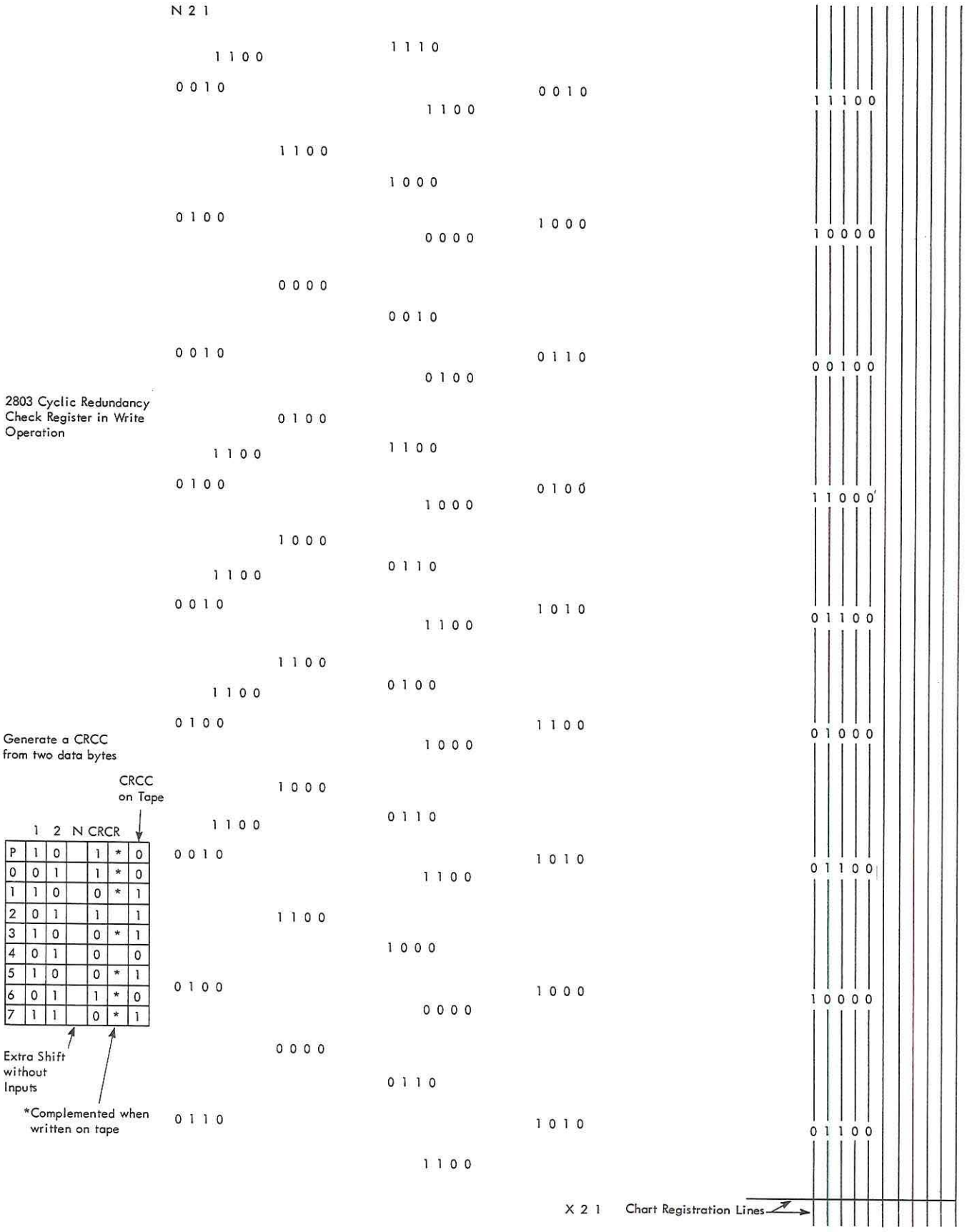
● Figure 41. Cyclic Redundancy Check Register Overlay

Figure 41. Cyclic Redundancy Check Register Overlay



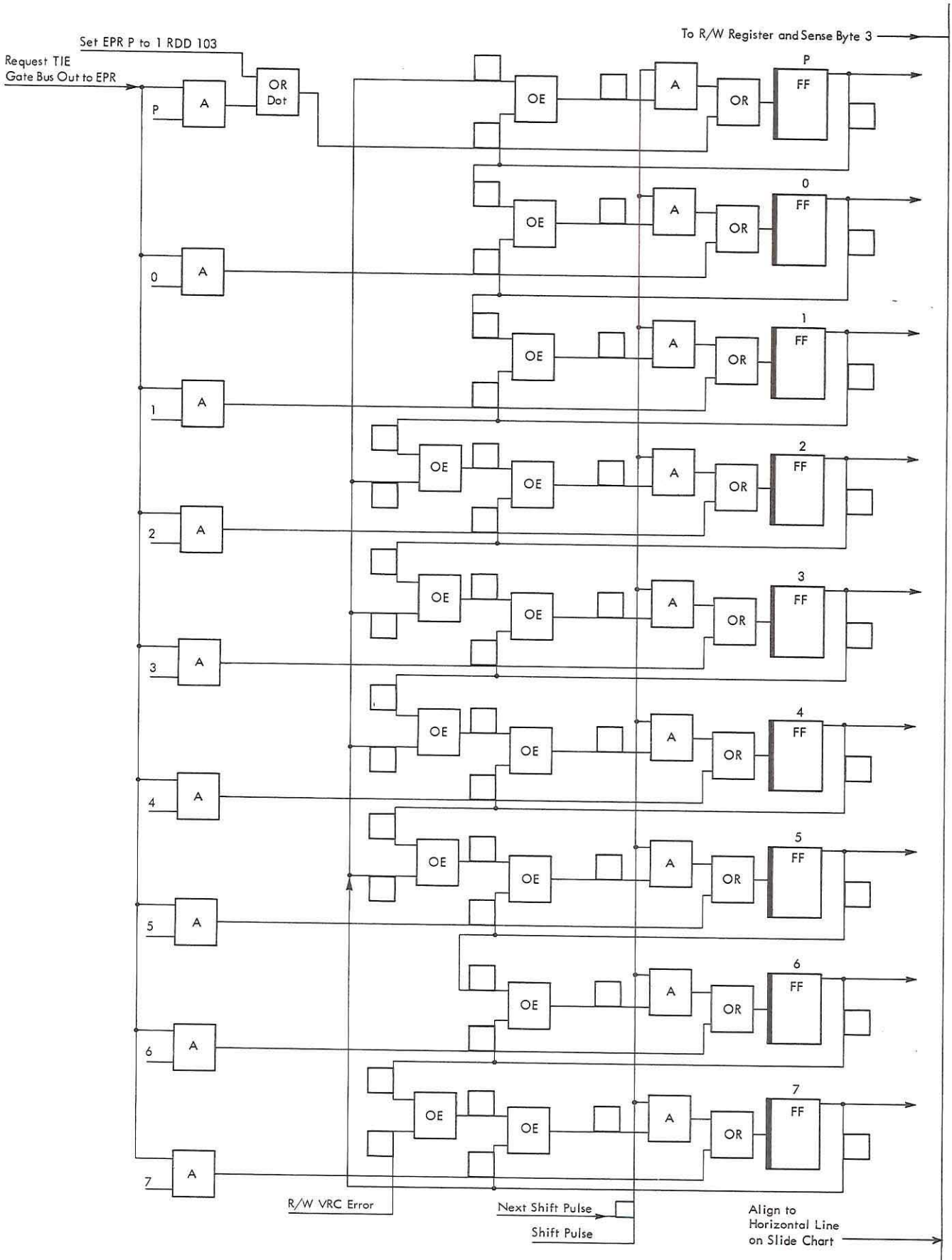
● Figure 43. CRCR Slide Chart No. 2 (Read Operation)

Figure 43. CRCR Slide Chart No. 2 (Read Operation)



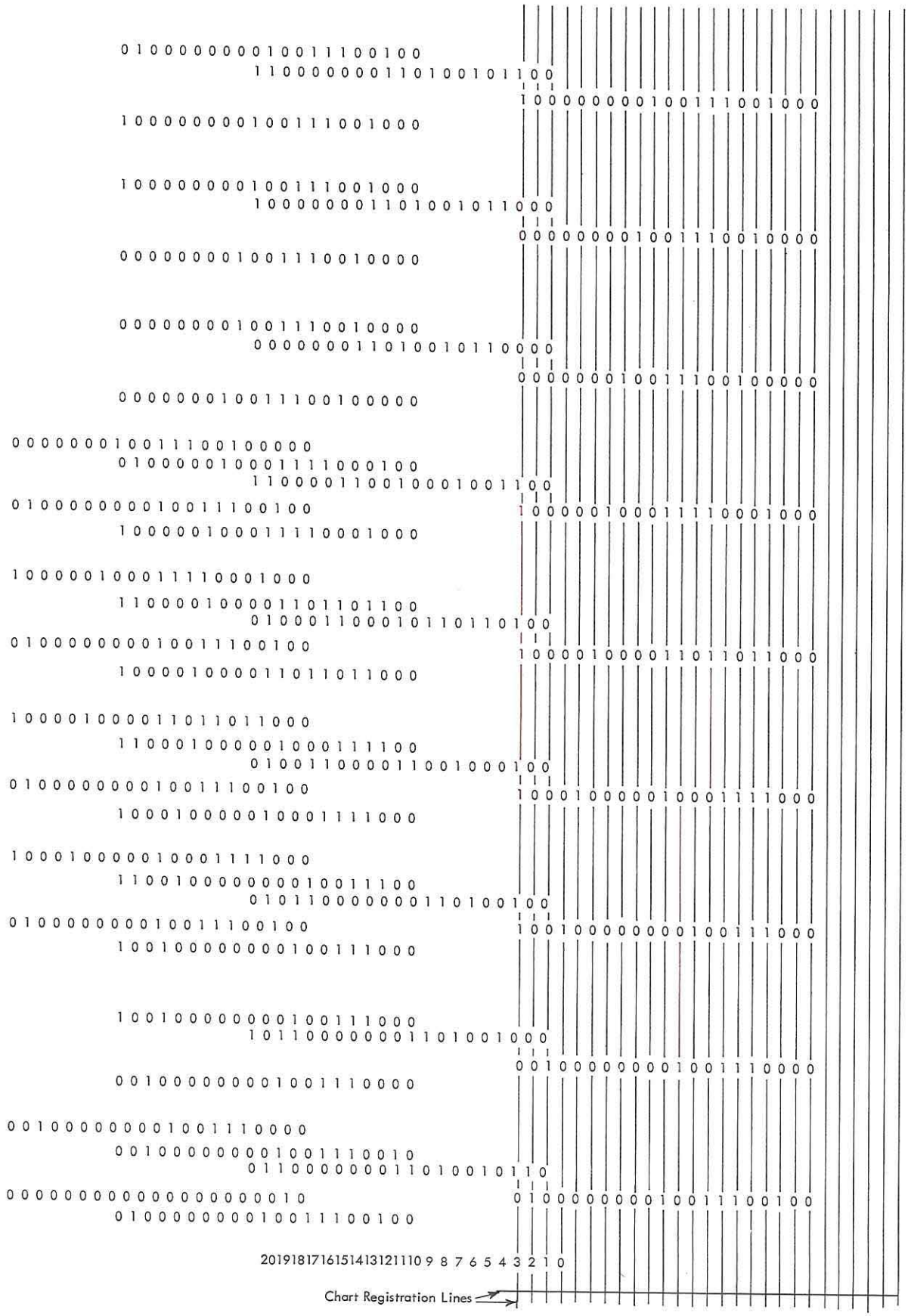
● Figure 42. CRCR Slide Chart No. 1 (Write Operation)

● Figure 42. CRCR Slide Chart No. 1 (Write Operation)



● Figure 44. Error Pattern Register Overlay

Figure 44. Error Pattern Register Overlay



● Figure 45. EPR Slide Chart

Figure 45. FPR Slide Chart

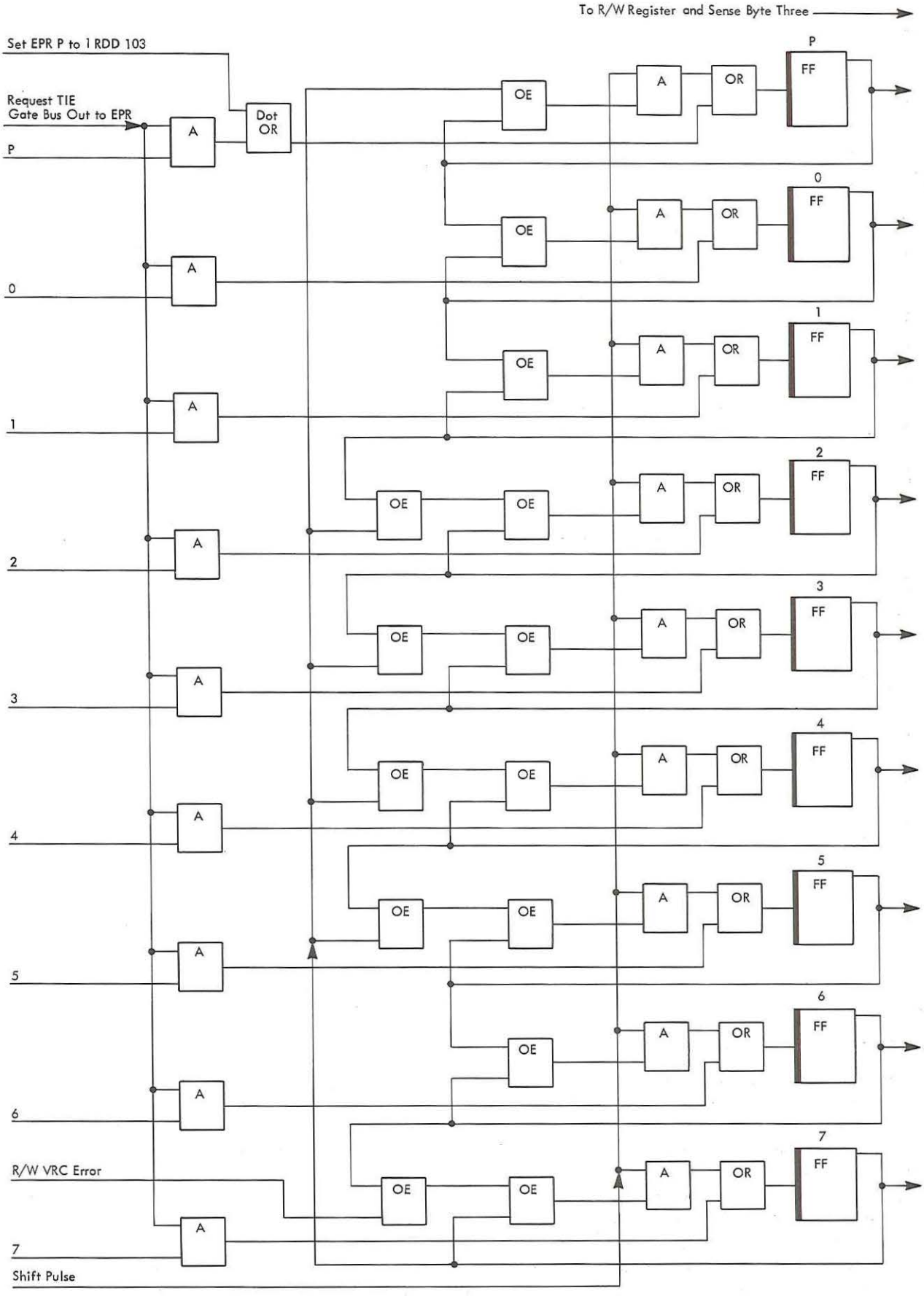
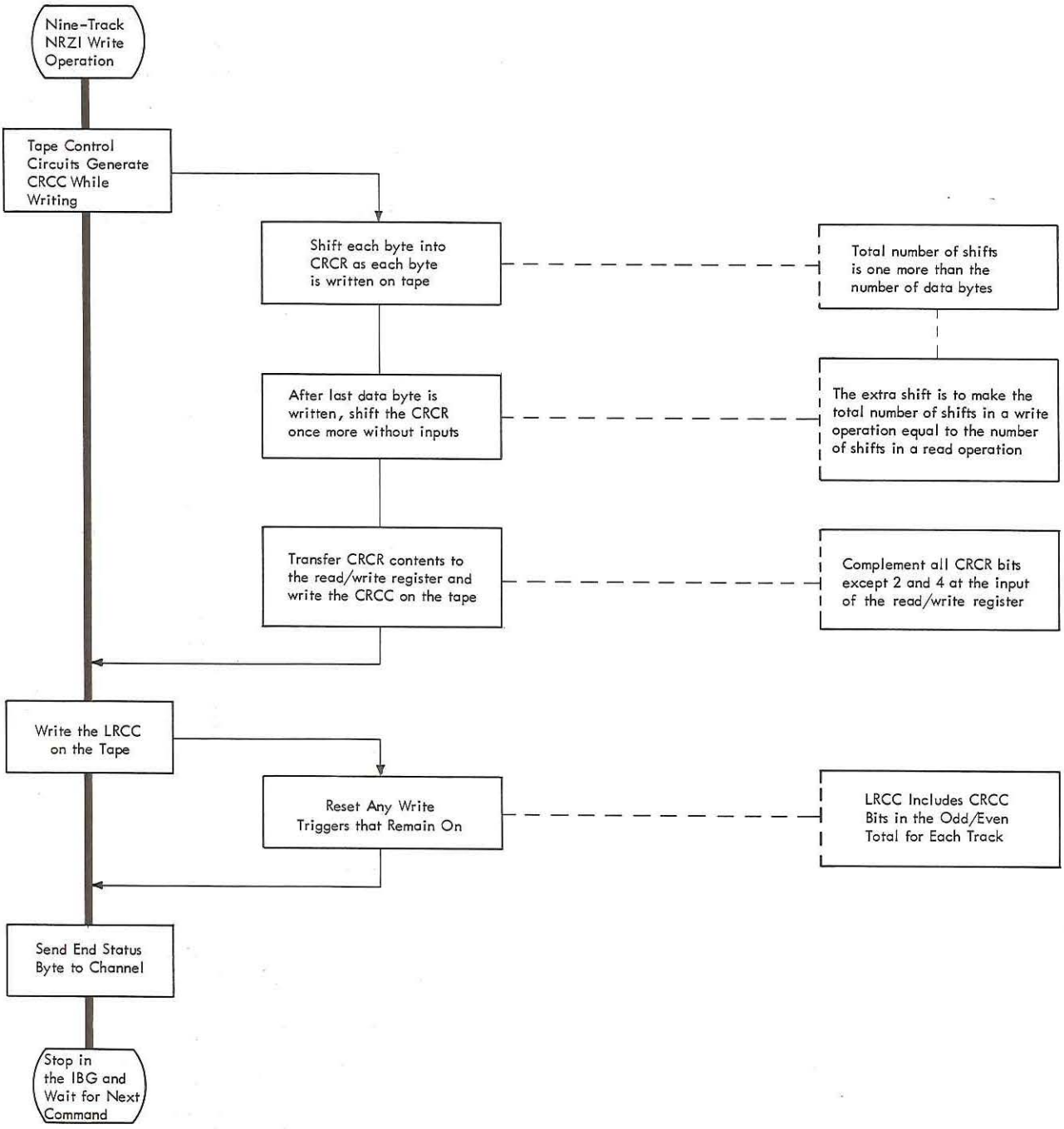


Figure 13. Error Pattern Register



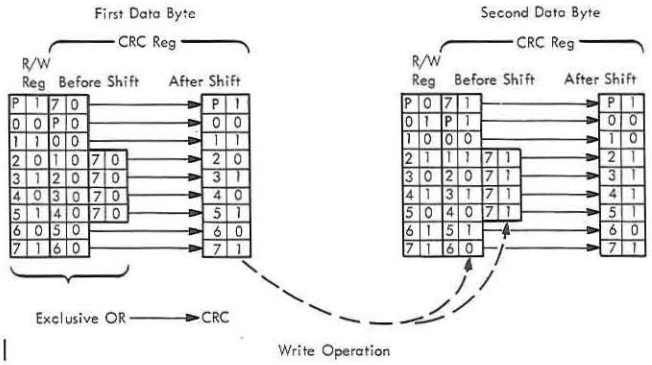
● Figure 14. Cyclic Redundancy Check Flow (Write Operation)

acter) is written four character spaces after the data and four character spaces before the LRC character (second check character).

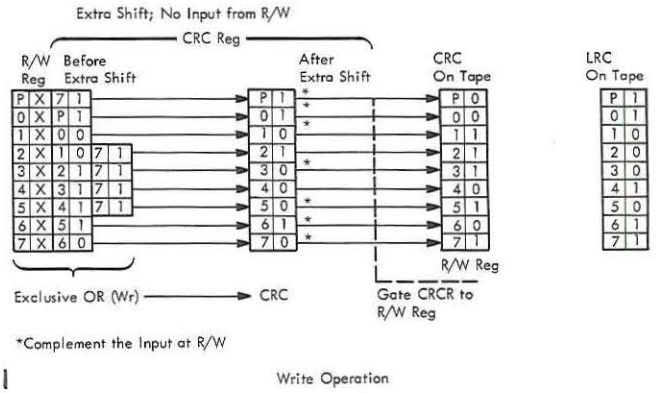
To write the CRC character on tape, the CRCR is gated into the R/W register at WDD47. Outputs from the CRCR triggers go back to the R/W register via "gate CRCR to R/W reg" on Figure 401. The CRC character written on tape is not the same bit configuration as the contents of the CRC register. CRCR bit positions are complemented (except 2 and 4) as they are transferred into the R/W register. If the CRCR has all positions on (111 111 111), the CRC character written on tape will be 000 101 000. The control circuits for CRC generation during a nine-track write operation are shown on Figure 404. "Shift CRCR" every WC-11 and WDD35-42 via OR circuit at 1F; "gate CRC to R/W register" via AND circuit at 5D.

The following is an example of CRC register operation during write. In this example two data bytes are written on tape and the CRC character is computed

and written on the tape. Data bytes used in this example are 101 010 101 and 010 101 011:



Notice that the first data byte enters the CRC register without change. The second byte is modified because the CRCR contains the first data byte at the time of the second shift. During write the CRCR is shifted one extra time, with no inputs, to produce the following result:



The CRC character written on tape will be 001 110 101. The LRC character will be 110 001 011. Notice that the CRC character contains odd parity. The shifting pattern of the CRC register always results in an *odd* parity CRC character when the block contains an *even* number of bytes. An *even* parity CRC character will result from a block with an *odd* number of bytes.

In nine-track operation the LRC character will always be *odd*. The pattern of bits in the CRC character assures that every LRC character will have *odd* parity.

Read Forward (No Errors)

Refer to CRC flow chart (Figure 15). During a read operation, the CRCR calculates a check character from the R/W register data (as in a write operation). Each character including the CRCC is shifted into the CRCR from the R/W register. The total number of shifts is equal to the number of data characters read plus one for the CRC character read from tape.

The odd/even character count trigger (TC.30.01.2) automatically adjusts the VRC circuits for correct parity when reading the CRC character as previously described in R/W VRC section.

At forward read, RDD83-86 time, the CRCR character read from tape is shifted into the CRCR on top of the calculated CRCR character. If no errors have occurred, the CRCR should contain a "match pattern." The match pattern is 111 010 111. Until the CRCR changes, the match pattern blocks setting of the CRCR error and the found track trigger. On Figure 404 the control circuits for CRC checking during read forward are:

1. "Shift CRCR" via shift data pulse, AND circuit at 5F and RDD83-86, AND circuit at 5H.
2. "Match pattern" recognition by AND circuits at 5F and 5G.

The following example illustrates CRCR and EPR operation during a read command with no data errors. Data used in this example is the same two byte block used in the write operation example:

1st data byte	101 010 101
2nd data byte	010 101 011
CRC character	001 110 101
LRC character	110 001 011

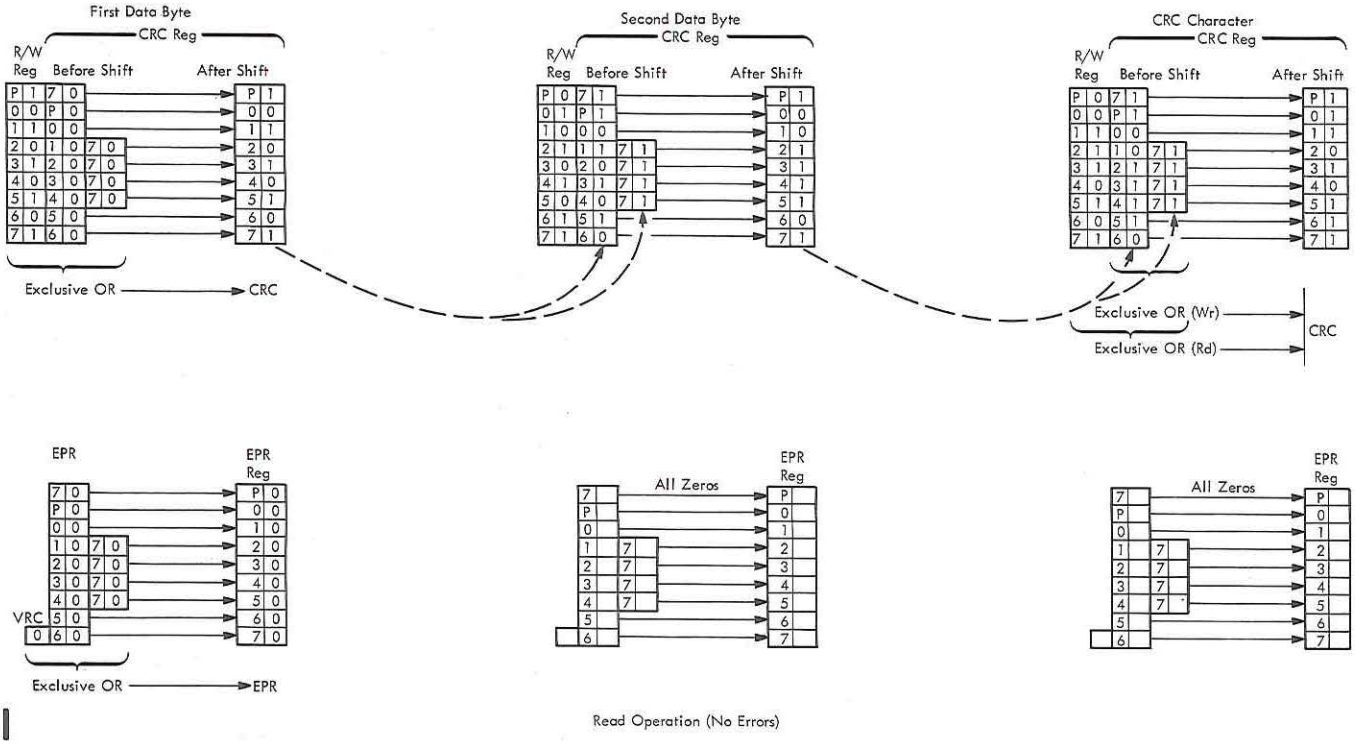
The first and second data bytes produce the same CRCR bit pattern that was produced during the two data cycles of the write operation. When the CRC character is shifted into the CRC register a match pattern (111 010 111) is produced. The EPR remains blank because no VRC errors were detected.

Read Backward (No Errors)

CRCR operation during a read backward is similar to read forward operation as indicated on Figure 15. The major differences are in the sequence of characters received from the tape, and inversion of the charactrs as they are shifted from the R/W register into the CRCR. R/W register P position goes to CRCR position 7; R/W 0 goes to CRCR 6, etc.

Inversion is required by the CRCR shift pattern when the data is received in reverse sequence. At the end of a read backward, if no errors are encountered, the CRCR should have a match pattern of 111 010 111 even though the CRC character is read before the data.

Because the parity of the CRC character depends on the odd/even character count of the record, CRC character parity cannot be checked at the beginning of a read backward operation. At this time, the backward trigger blocks VRC error circuits during both check character times. To resume normal VRC error checking, the backward trigger is reset before reading the data portion of the record. A read error in the CRC character will not be detected directly. If the CRC char-



Read Operation (No Errors)

acter is read incorrectly, a track in error cannot be found, and the CRCR error trigger turns on at RDD169. A CRCR error indicates that the CRCR did not contain a match pattern (111 010 111) at the end of a nine-track read or read backward. The control circuits used for read backward are the same as those used during read forward. The only exception on Figure 404 is the source to shift the CRC register when the CRC character is read. This is accomplished by the AND circuit at 5G which is controlled by the "backward" trigger and RDD27-30 time.

Search for Track In Error

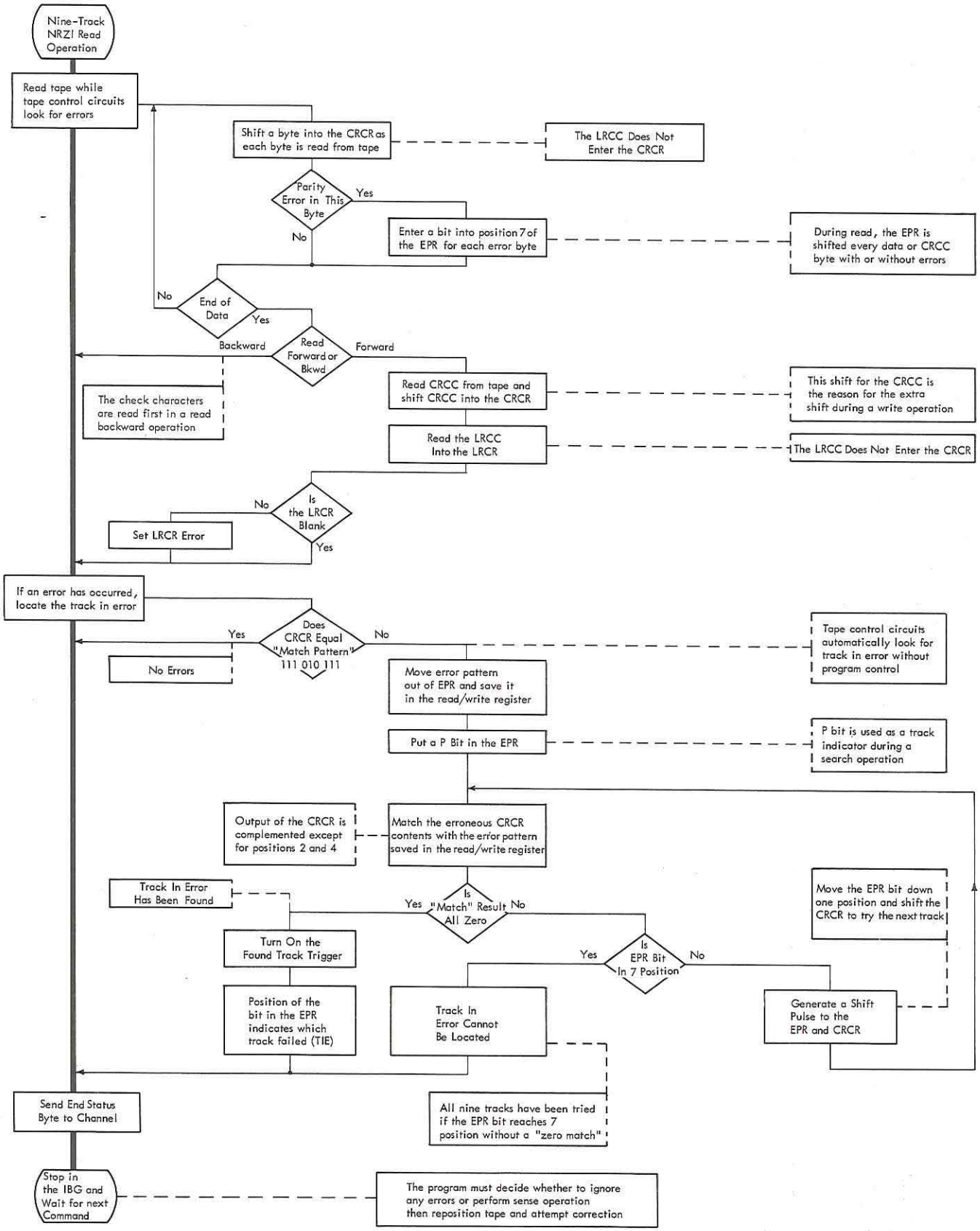
While a record is being read during either read forward or backward operations, each R/W register VRC error detected will attempt to set the EPR 7 trigger (exclusive OR'ed with EPR 6). The EPR is shifted for each byte in the block at the same time that the CRC register is shifted. Thus, an error pattern is developed in the EPR which is used to determine the track in error (TIE). If no R/W VRC errors occurred, the EPR would be blank at the end of the block.

Utilizing Figure 15, under the section labeled "locate the track in error," the error pattern is saved by transferring it to the read/write register at RDD95-98 time. To determine the actual track in error, the error pattern (now in the R/W register) must be exclusive OR'ed with the CRCR character. Before the exclusive OR action occurs, the EPR is reset and "primed" with a P bit at RDD103 time. This bit will be used to indicate the failing track when it is located.

If the R/W VRC error trigger is on, the CRCR is gated to the R/W register outputs at RDD95-158. Each bit position is complemented except 2 and 4. Note on Figure 401 that CRCR positions are not gated into the R/W register triggers. They are gated to the outputs of the triggers.

The R/W register triggers contain the error pattern from the EPR. If a bit position in the CRCR (after complementing) matches the same bit position in the R/W register, the corresponding output is inactive or zero level (Figure 401). When the complemented (except 2 and 4) CRCC and the error pattern combine, during error search, so that all outputs are down, the result is a plus zero match (Figure 404). If the error pattern in the R/W register does not produce an all zeros output (zero match) with the complemented CRC register contents, a shift pulse is generated to shift both the CRCR and the EPR. This shift pulse will shift the CRC register within itself (no inputs) and shift the EPR bit down one position. The newly shifted CRC register content is again matched with the error pattern in the R/W register. Shifting and matching continue until either a zero match occurs or the bit in the EPR reaches the 7 position. A bit in EPR position 7 blocks any further shift pulses because all nine possible tracks have been tried.

If an all zeros match occurs between the error pattern in the read/write register and the complemented CRCR, the shift pulses are blocked and the "found track" trigger is turned on. The track in error (TIE) is indi-



● Figure 15. Cyclic Redundancy Check Flow (Read Operation)

cated by the position of the bit in the EPR. When reading forward, the EPR bit stops in the reverse position from the actual track in error. A P bit in the EPR indicates a 7 track in error; a 0 bit in the EPR indicates a 6 track in error, etc. When reading backward, the track in error bit is in the corresponding position (not reversed). The EPR output gating automatically reverses the bit positions to the bus in lines after reading forward. Sense byte 3 will contain the track in error bit in the true position for both forward and backward operation.

If no zero match occurs, then the track in error cannot be found. This condition could be caused by errors in more than one track during reading or an uncorrectable combination. After all nine tracks have been tried unsuccessfully, a 7 bit remains in the EPR. This bit does not necessarily mean that the 7 or P track is in error (because found track is not on).

When a zero match occurs, the found track trigger is turned on and the outputs of the EPR are available to the bus in lines when sense byte 3 is called for. Sense byte 3, therefore, contains one bit indicating the track in error. If no track is found, bits 6 and 7 are forced on in sense byte 3.

The following example uses the same data block as the write and read examples:

1st data byte	101010101
2nd data byte	010101011
CRC character	001110101
LRC character	110001011

During reading, assume that the 6 track indicates a 1 bit in every byte. The first data byte and the CRC character will indicate VRC errors. The second data byte and the LRC character will not be affected because they already have bits in position 6. CRCR and EPR operations will be as shown below.

Notice that the match pattern was not produced in the CRC register. Because VRC errors were detected, the EPR developed an error pattern. The erroneous CRC register contents and the error pattern can now be used to determine the track in error (TIE).

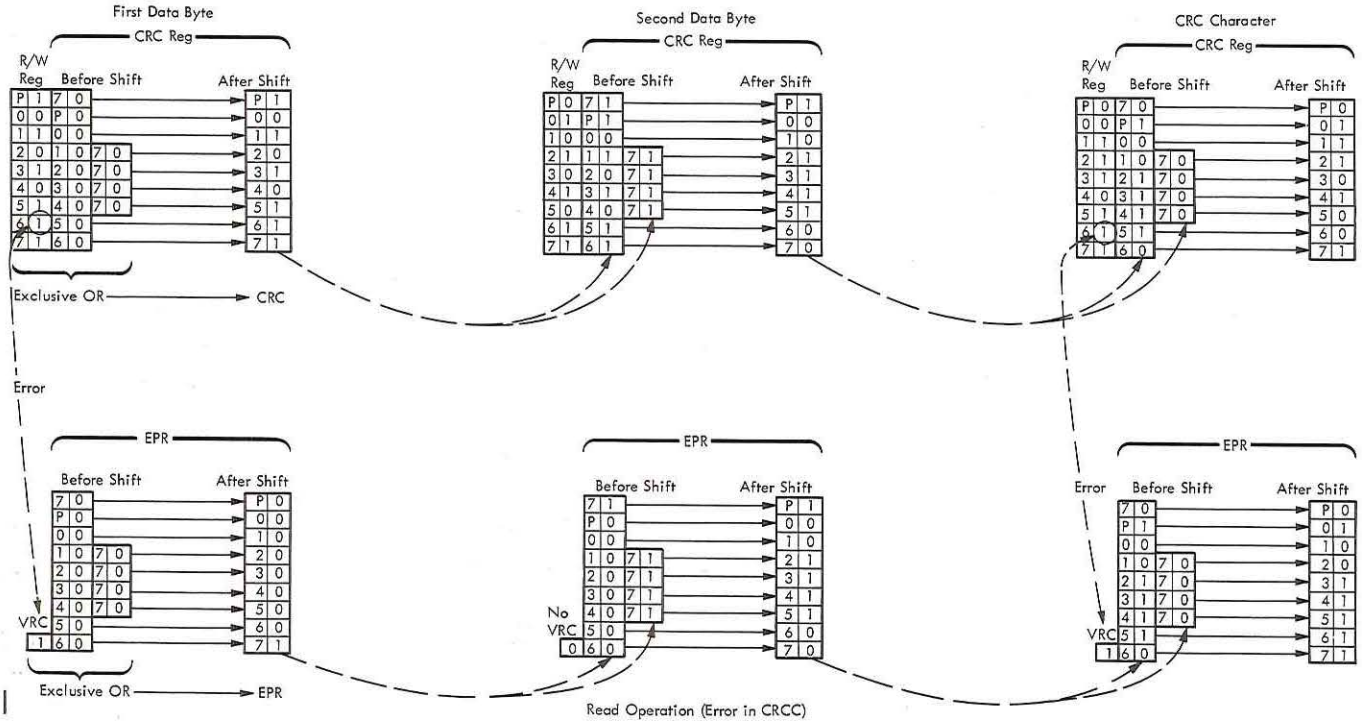
At RDD-95-98 time the error pattern is moved into the read/write register.

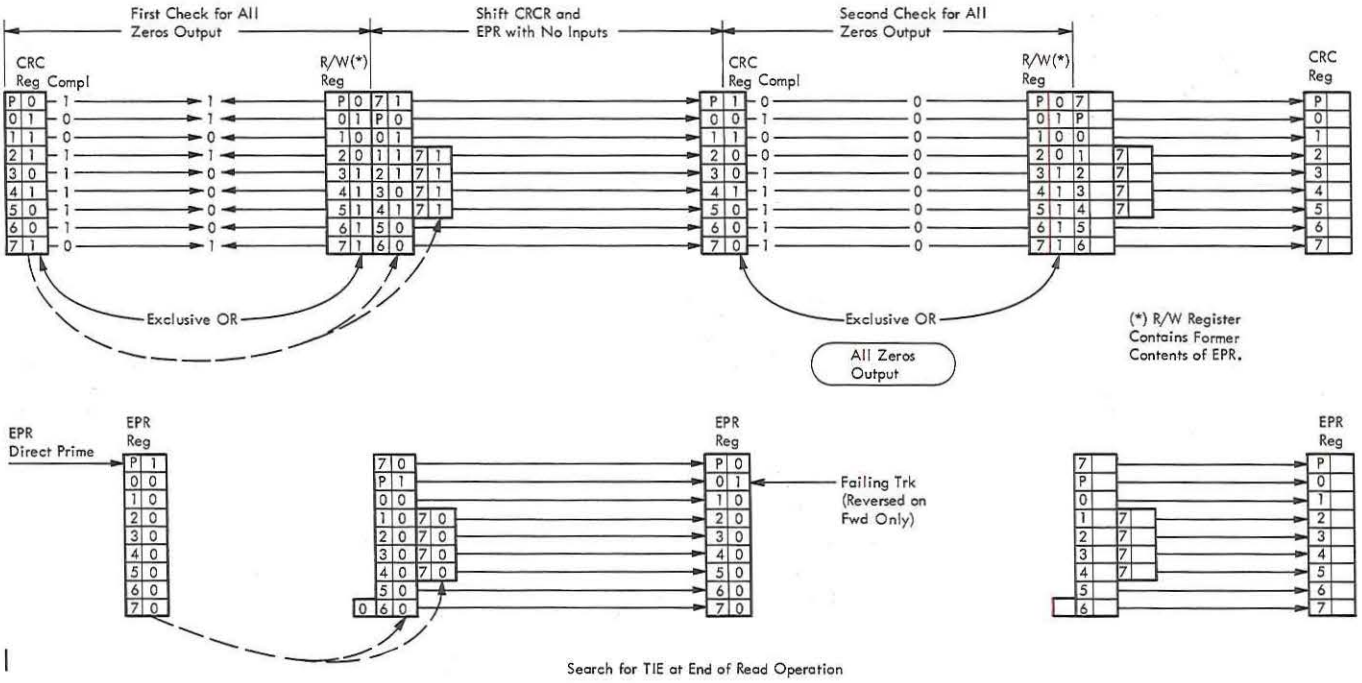
At RDD-103 time the EPR is primed with an indicator bit. The erroneous CRC register contents are compared with the error pattern now in the read/write register. If a zero match is not produced, the CRCR and EPR are shifted and again compared.

Because a zero match occurred on the second try, the EPR bit stopped in position 0. In forward operation this indicates that the track in error is 6. Notice that the error correction circuits identified the TIE even though the CRC character contained a parity error. This holds true as long as all the errors are in the same track within the block.

Read Correction Operation (Figure 16)

Correction of the failing track is not performed automatically. Correction is accomplished only if the con-





trol unit is programmed to do so. The channel should have received the end status byte of the read operation. This status would contain a unit check as a result of the error. A sense command should be executed immediately to obtain the TIE bit in sense byte 3. After the tape has been positioned to reread the same record in the same direction, a "request TIE" command loads the track in error bit back into the EPR from the bus out lines and turns on the correcting trigger. As the record is read again each VRC error gates the EPR bit output to the output of the R/W register. The corresponding bit in each redundant character is inverted on the bus lines. Good parity characters are not affected.

By inverting the track in error bit in each redundant byte, the data and CRC character should be restored to their original condition. If a bit had been picked up it would be deleted, and if a bit had been dropped it would be restored. The error bytes corrected may not be the same error bytes detected on the first read pass. The correction is valid as long as all errors encountered are in the same track on the read correction pass. To guard against false corrections a new CRCR character is calculated using the corrected data. The new CRCR character is combined with the CRC character read from the tape. The two characters are combined in the CRCR. If they do not form a match (111 010 111), the CRCR error trigger is turned on at RDD169.

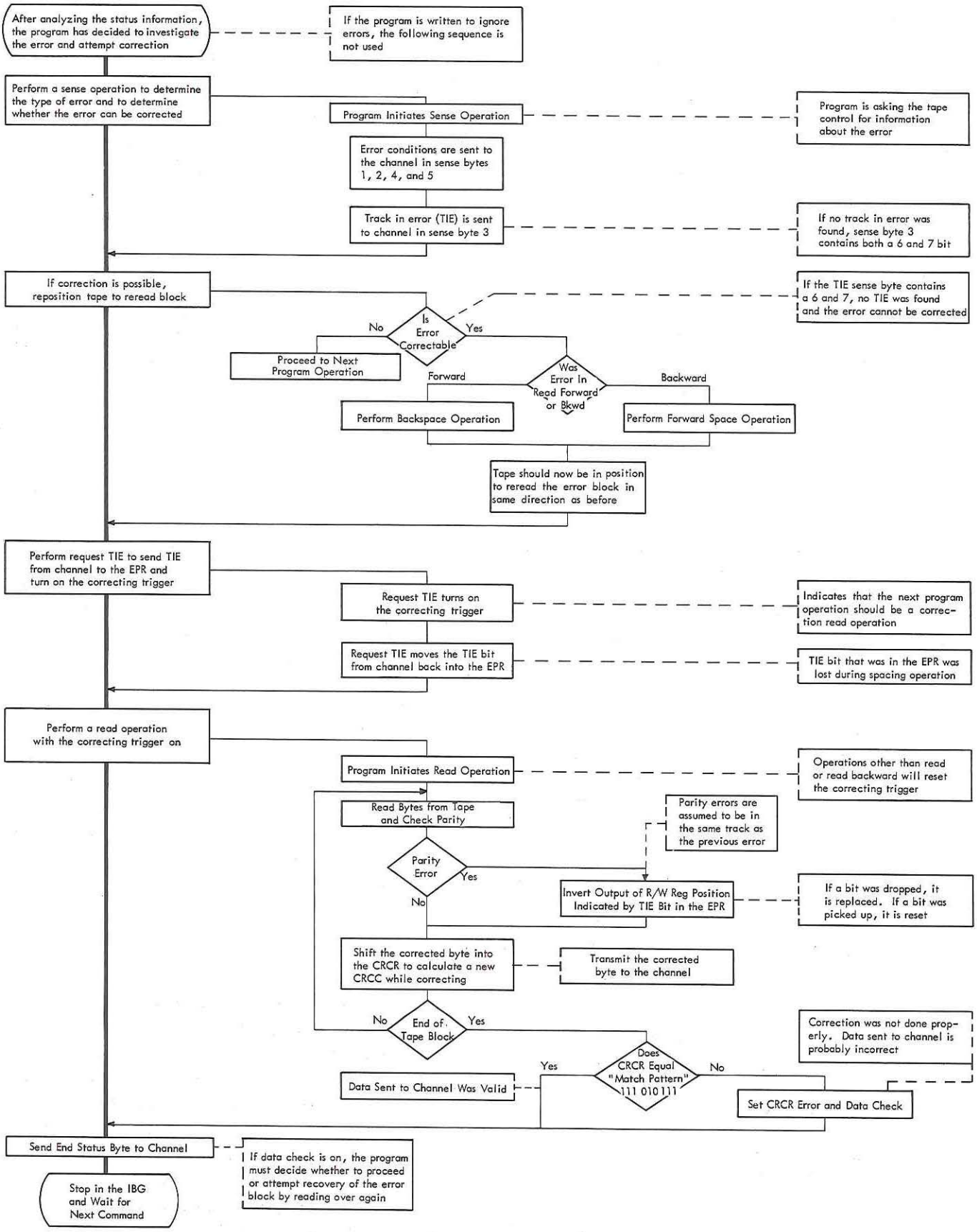
When correcting a block, no attempt is made to determine a new track in error. No error pattern can be developed because the EPR is holding the track in error bit for the correction circuit. During an error correction read operation, the LRCR receives uncorrected data from the skew register. Bits remaining in the LRCR may not represent an error condition in the data sent to the channel. To prevent false LRCR errors, the EPR track in error bit blocks LRCR indications from the faulty track. The remaining positions are sampled at RDD169 time as usual.

Refer to the control circuits on Figure 404, lower-right corner. "Gate EPR to correct" is only active when a R/W VRC occurs and the correction trigger is set.

The parity position of the R/W register is inverted if correction is to occur and EPR is indicating that the "P" bit is the TIE (Figure 401).

Read Clock

- **Controls data timing in read operations *and* read checking of write operation.**
- **Three binary triggers stepped by drive pulses.**
- **Drive pulses originate from oscillator timings selected by density and mode of tape unit.**
- **Drive pulses gated to clock by "first bit" latch active.**
- **"First bit" latch set at the arrival of the first high-clip pulse from the final amplifiers.**
- **Proper stepping sequence of the clock is checked by a VRC circuit; an error will set equipment and unit check.**



● Figure 16. Cyclic Redundancy Check Flow (Correction Operation)

- **One complete read clock cycle per tape character read.**

The read clock is a circuit to provide timing pulses to accept data from the read bus lines. Whenever bytes are read from tape, the read clock must run to provide gating and synchronizing pulses. Operations which read bytes from tape include:

1. Read and read backward
2. Write and write tape mark (read checking)
3. Forward or backward spacing

The circuit to start the read clock is blocked on a rewind or rewind unload. Reading is not needed during a rewind instruction because the tape unit is searching for the load point reflective marker (not tape marks or gaps).

Read Clock Circuits

The heart of the read clock is three binary triggers on Figure 206. A negative shift at the input of a trigger will complement that position. If it was on, it will turn off. If it was off, it will turn on.

Drive pulses to the read clock vary in frequency according to the model of tape unit and the density of the bytes on tape. The frequencies are:

TU MODEL	DENSITY	OSC FREQUENCY
1	200	58.33 kc
1	556	166.75 kc
1	800	233.30 kc
2	200	116.66 kc
2	556	333.50 kc
2	800	466.60 kc
3	200	175.00 kc
3	556	500.00 kc
3	800	700.00 kc

The proper oscillator frequency is selected and gated to the read clock drive for each operation.

When not in operation, the binary triggers are held reset by a positive level. After the reset is removed the drive pulses begin stepping the triggers in a cyclic pattern. Only one trigger at a time changes state. If the circuits stepped in a binary pattern it would be possible for all the triggers to change on the same drive pulse shift.

When all the triggers change at once, a time period is wasted while the triggers "settle down." The cyclic pattern, with only one trigger changing at a time, provides "cleaner" outputs and faster stepping.

Read clock drive will change one clock trigger with each shift (both plus and minus). Two triggers will change for each complete drive pulse cycle.

A timing chart in Figure 206 shows the relationship of triggers and drive pulses.

Each time the plus read clock drive goes positive, the line is inverted to switch trigger A. Trigger A goes on and off alternately at every positive drive pulse shift.

The B trigger will be switched when the A trigger is on and the minus read clock drive goes positive.

The C trigger will be switched when the A trigger is off and the minus read clock drive goes positive.

During read operations the read clock steps up to RC7 before it is reset. For write operation (read checking) the read clock continues stepping into a second cycle and is reset at RC9.

Read clock cycle times in relation to trigger states are shown in the following table:

READ CLOCK TRIGGERS			CYCLE
C	B	A	TIME
0	0	1	RC 0
0	1	1	RC 1
0	1	0	RC 2
1	1	0	RC 3
1	1	1	RC 4
1	0	1	RC 5
1	0	0	RC 6
0	0	0	RC 7
0	0	1	*RC 8
0	1	1	RC 9

*Beginning of the second cycle.

The key to operation of the read clock is the reset line to the triggers. The clock can run only when the reset line is deconditioned. "Start read clock" and "read condition" are the two conditions that control the reset circuit.

Figure 206 illustrates the circuits to control starting and resetting of the read clock. "Read condition" remains on during the time when information is expected from tape. When data bits leave the high-clip amplifiers, the first bit trigger turns on. If the operation is not a rewind or rewind-unload, the reset line is deconditioned and the read clock begins to run.

Read clock stepping continues until the reset line is made active again. In read type operations the reset singleshot will be pulsed at the end of RC6 time. In write operations the reset singleshot will be pulsed at the end of RC5 time. Output of the singleshot turns on the inhibit read clock trigger which in turn blocks the start read clock line. The read clock triggers will be held reset until the next byte leaves the final amplifiers and turns on the first bit trigger again.

Read Clock VRC

As the clock triggers step, the vertical redundancy check (VRC) circuit monitors the sequence. If the clock

triggers do not step correctly, an error is signaled and a "CLK ERR" light on the CE panel is turned on.

A redundancy check of the clock circuit is possible because only one trigger changes state with each drive pulse shift. The total number of triggers on will be alternately odd and even. The VRC circuit checks to see that:

1. An odd number of triggers are on when the + read clock drive is plus.
2. An even number of triggers are on when the + read clock drive is minus.

Clock checking is described and illustrated in the "objectives" section of Figure 206 (upper right).

Write Clock

- **Four stage counter used to control data flow and function during write and WTM operations.**
- **Stepping pulses are drive pulses which originate from various oscillator timings.**
- **Oscillator timings selected by model of TU and density desired.**
- **For each drive pulse the clock will step two counts.**
- **Continuously running clock; once it is gated on, its cycles repeat until all data is written.**
- **Write clock is also checked for correct stepping sequence (VRC).**
- **VRC error will set equipment and unit checks.**

The write clock in Figure 205 is a circuit to provide timing pulses to write data on tape. Whenever bytes are written on tape, the write clock must run to provide gating and synchronizing pulses. Operations which write bytes on tape are:

1. Write
2. Write tape mark

Erase is a write type operation, however, no bytes are written. The write clock is not needed for erase tape.

Write Clock Circuits

Four binary triggers are the heart of the write clock on Figure 205. A negative shift pulse at the input of a trigger will complement that position. If it was off, it turns on. If it was on, it turns off.

Drive pulses to the write clock vary in frequency according to the model of tape unit (speed) and the byte density to be written. A byte is written on tape for each write clock cycle. The drive frequency has a direct bearing on the spacing of bytes (density) on tape.

Write clock drive frequencies are:

TU MODEL	DENSITY	OSC FREQUENCY
1	200	60.0 kc
1	556	166.8 kc
1	800	240.0 kc
2	200	120.0 kc
2	556	333.5 kc
2	800	480.0 kc
3	200	180.0 kc
3	556	500.0 kc
3	800	720.0 kc

The proper oscillator frequency is selected and gated to the write clock drive for each operation.

When not in operation, the binary triggers are held reset by a positive level (Figure 205). After the reset is deconditioned, the drive pulses begin stepping the triggers in a cyclic pattern. One trigger changes state for each half of the drive pulse cycle. One trigger is on the plus shift; one trigger on the minus shift.

If the circuits stepped in a binary pattern, it would be possible for all the triggers to change on the same drive pulse shift. When all the triggers change at once, a time period is wasted while the triggers "settle down." The cyclic pattern, with only one trigger changing at a time, provides "cleaner" outputs and faster stepping.

The timing chart at the bottom of Figure 205 shows the relationship of triggers and drive pulses.

Each time the write clock drive goes minus the A trigger changes state. Trigger A turns on and off alternately at every negative drive pulse shift.

The B trigger will switch when the A trigger is on and the drive goes positive.

The C trigger will be switched when the A trigger is off, the B trigger is on, and the drive goes positive.

The D trigger will be switched when the A and B triggers are off and the drive goes positive.

During operation the write clock steps up to 15 and then repeats from 0 to 15 again. One write clock cycle is from 0 to 15. Write clock cycle times in relation to trigger states are shown in the following table:

WRITE CLOCK TRIGGERS				CYCLE
D	C	B	A	TIME
0	0	0	1	WC-1
0	0	1	1	WC-2
0	0	1	0	WC-3
0	1	1	0	WC-4
0	1	1	1	WC-5
0	1	0	1	WC-6
0	1	0	0	WC-7
1	1	0	0	WC-8
1	1	0	1	WC-9
1	1	1	1	WC-10
1	1	1	0	WC-11
1	0	1	0	WC-12
1	0	1	1	WC-13
1	0	0	1	WC-14
1	0	0	0	WC-15
0	0	0	0	WC-0

The key to operation of the write clock is the trigger reset line. The clock can run only when the reset line is deconditioned. "Write trigger release" and "write condition" are the two conditions that control the reset circuit.

Write condition remains on during the time when information can be written on tape. Write condition is not on while tape is accelerating or stopping. Write condition also turns on the write trigger release trigger (Figure 335, coordinates 4E).

When write trigger release and write condition are both active (Figure 205), the reset line is deconditioned and drive pulses are gated to the triggers. If write condition falls after WC-7 time, the clock will continue stepping until the end of the clock cycle. The write clock continues running until write trigger release falls or write condition and the D trigger are off at the same time. Each time the write clock steps from 0 through 15, a byte is written on tape.

At the end of data transmission, write condition must be turned off. When the clock reaches WC-14 or WC-15 after stop data transfer comes up, write condition is reset. The clock finishes the cycle through WC-15. After WC-15 the reset line to the clock triggers comes up to stop the clock. The write clock triggers will be held reset until the next write or write tape mark operation.

Write Clock VRC

As the clock triggers step, the vertical redundancy check (VRC) circuit monitors the sequence. If the clock triggers do not step correctly, an error is signaled and a CLK ERR light on the CE panel is turned on.

A redundancy check of the clock circuit is possible because only one trigger changes state with each drive pulse shift. The total number of triggers on will be alternately odd and even.

The VRC circuit checks to see that:

1. An odd number of triggers are on when the WC drive is minus.
2. An even number of triggers are on when the WC drive is plus.

Triggers A, B, and C are checked as a group. Output of the A, B, C check circuit (point H on Figure 205) is minus when A, B, and C have an odd number of triggers on. Trigger D output is combined with the drive pulses to form the signal at point L. Point H and point L should not be plus at the same time. If a clock error occurs, point H and point L will be plus at the same time which turns on the WC VRC trigger. A delay circuit is included in the input of the WC VRC trigger to prevent false errors. As the clock triggers change state, a time duration is needed for the VRC circuit to reflect the change. During this "settle down" time a short duration spike could turn on the VRC error trigger. The delay allows only true error condition through to set the R/W VRC. A pulse shorter than about 600 ns will not set the error trigger.

Delay Counter

- **Nine trigger counter used whenever TCU must delay to gage start/stop timings of the TU, gage time between characters, or measure some significant time period during operations.**
- **Delay counter can be stepped in either microsecond mode or millisecond mode.**
- **Stepping pulses originate from various oscillator outputs which are gated by the model of the TU selected in millisecond mode or density and model in microsecond mode.**
- **Since the delay counter has various functions, it is gated on and off at various times, depending upon when it is used.**
- **Correct stepping of the delay counter is checked by a specific VRC circuit.**
- **VRC error will generate equipment and unit checks.**

The delay counter is a group of circuits to provide timed pulses for measurement of tape travel. It can be used to measure the spacing between bytes when reading or generate a time delay while the tape unit performs a mechanical operation such as start tape, stop tape, or turnaround. Delay counter timings are used to measure the spacing of check characters and tape marks in both reading and writing. Figure 405 shows delay counter circuits. Figure 204 shows delay counter VRC circuits.

Delay Counter Circuits

The basic counting of the delay counter is done by nine binary triggers on Systems 70.01.1 and 70.03.1. A negative shift pulse at the input of a trigger will complement that position. If it was off, it turns on. If it was on, it turns off.

Drive pulses to the delay counter vary in frequency according to the model (speed) of tape unit, the density of the bytes, and what type of measurement is to be made. Drive pulses to the delay counter are in two groups: millisecond mode and microsecond mode.

Microsecond mode is primarily used for measurement of data byte spacing. Millisecond mode is primarily used for measurement of tape motion, such as load point delay or gap delays.

In millisecond mode the delay counter is driven at one of three frequencies: 5.0 kc, 2.5 kc, or 1.25 kc. The 5.0-kc oscillator output drives a binary trigger. Output of the trigger forms the 2.5-kc signal. The

signal is again divided by a trigger to generate the 1.25-kc drive pulses. The delay counter, in millisecond mode, uses:

1. The 5.0 kc for Model 3 tape units
2. The 2.5 kc for Model 2 tape units
3. The 1.25 kc for Model 1 tape units

In microsecond mode the delay counter is driven by the same group of oscillators that drive the write clock. The frequencies are selected according to tape unit model and byte density. The frequencies used by the delay counter in microsecond mode are:

TU MODEL	DENSITY	OSC FREQUENCY
1	200	60.0 kc
1	556	166.8 kc
1	800	240.0 kc
2	200	120.0 kc
2	556	333.5 kc
2	800	480.0 kc
3	200	180.0 kc
3	556	500.0 kc
3	800	720.0 kc

When the delay counter is not running, the triggers are held reset by a positive level. If neither the millisecond control nor the microsecond control is active, the triggers are held reset. The reset can be brought up by a variety of conditions (Figure 405). Reset is activated by such events as the end of an operation or a change from millisecond to microsecond mode (or vice versa). If millisecond and microsecond mode are down at the same time, the reset comes up.

The delay counter uses a cyclic stepping pattern similar to the pattern used in the write and read clocks. One trigger changes state for each half of the drive pulse. Output of the delay counter is "cleaner" and faster than the binary type of counter because there is no delay time waiting for a group of triggers to "settle down."

Figure 204 contains a sequence chart of the delay counter triggers. Specific count timings can be decoded from the trigger positions. The maximum possible count is 511. A chart of count values is included in Figure 204.

Notice that the count chart in Figure 204 is divided into four major groups. The condition of triggers H and G determine the division of groups (along the bottom of the chart). For example, the delay counter value RDD169 (circled) falls in the third group because triggers H and G are both on.

Within each group, the F and E triggers form smaller groups (to the left of each major group). For RDD169, triggers F and E are both on.

Within the F and E groups are divisions formed by triggers D and C (immediately to the right of F and E). For RDD169, the D and C triggers are both on.

Follow the line (for D and C on) to the right. RDD169 is in the column under B and A triggers both on. Trigger conditions and count values can be determined for any delay counter timing up to 255. For values between 255 and 511, the J trigger turns on and the chart sequence repeats.

For example, the value 256 is actually 0 in the chart sequence with the J trigger on (top line of the first group).

Delay Counter VRC

A vertical redundancy check (VRC) circuit monitors the stepping pattern of the triggers. Because only one trigger changes state with each drive pulse shift, the total number of triggers on will be alternately odd and even. The VRC circuit checks to see that:

1. An odd number of triggers are on when the drive is positive.
2. An even number of triggers are on when the drive is negative.

The entire VRC circuit can be viewed as a parity checking circuit that looks for an even total number of lines including the drive pulse. One exception is that the J trigger is not included in the VRC inputs.

The set path to the VRC error trigger includes a delay to filter out transient pulses as the triggers change state. To signal an error, the faulty condition must stay on for about a microsecond or longer.

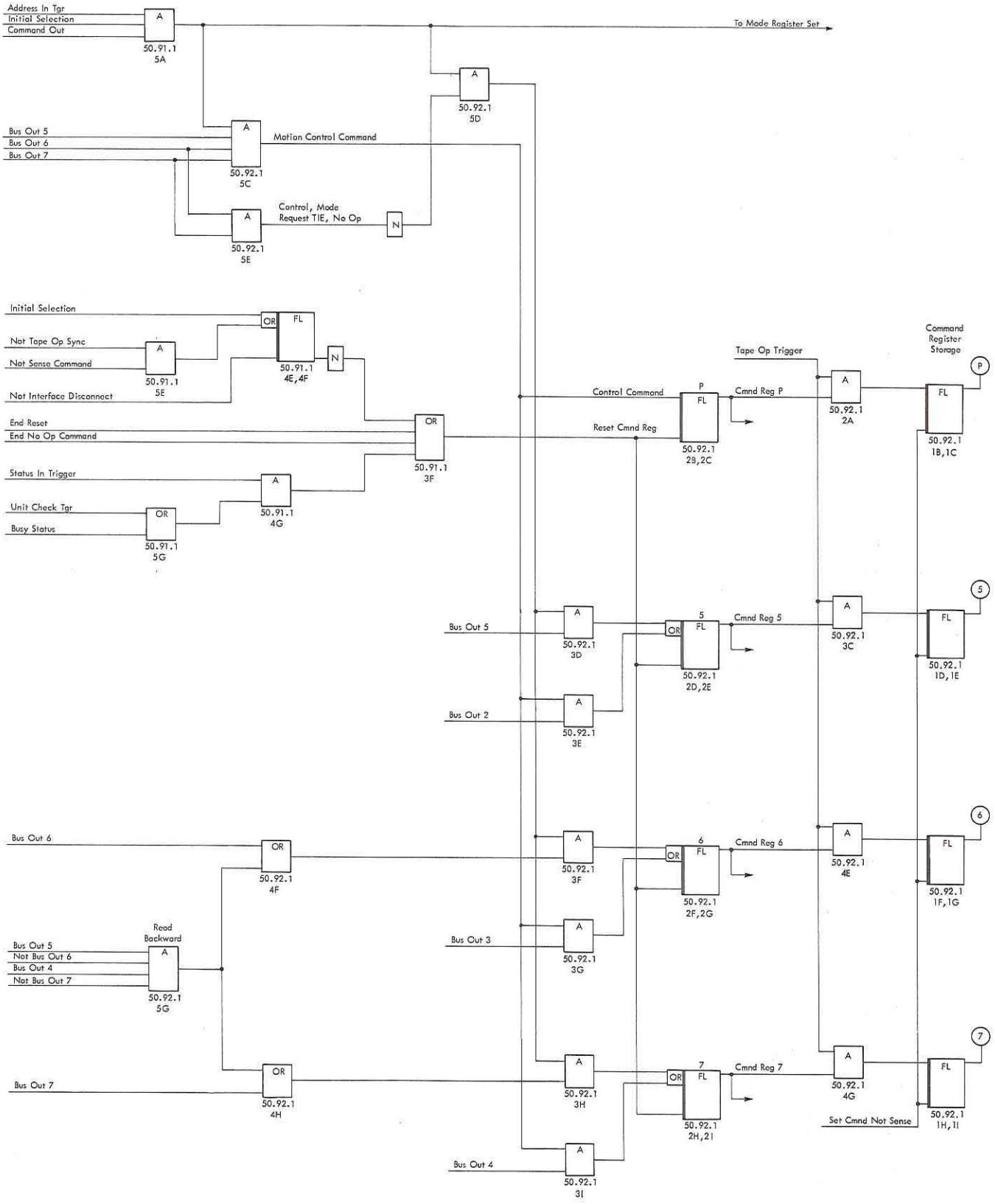
"Counter checking" is described on the bottom of Figure 204.

Command Register

- **Commands fit roughly in two categories: data transmission commands and control commands.**
- **Bits set into the command register may not correspond directly to the bits received in the command from channel.**
- **Commands which have 5, 6, and 7 bits on will set the P bit of the command register then set bus out bits 2, 3, and 4 into command register positions 5, 6, and 7.**

The command register contains bits P, 5, 6, and 7 of a command while it is being executed. Outputs from the command register are decoded to generate control lines for each operation. At the end of each operation the command register is reset. Figure 17 illustrates the command register set and reset circuits. When a command is received during initial selection, bits 5, 6, and 7 determine what will be done with the command byte.

If the command contains bits 6 and 7, it is a control command and will be handled differently than data



● Figure 17. Command Register Set

transmission commands. Commands which contain bits 5, 6, and 7 are tape motion control commands (a type of control command).

When a tape motion control command is received, the P bit is set in the command register. At the same time bits 2, 3, and 4 of the bus out lines (the command byte) are gated into positions 5, 6, and 7 of the command register.

Control commands which contain only the 6 and 7 bits do not set the command register. No op, mode set and diagnostic commands use only initial selection time and, therefore, do not require a place to store the command for execution. Control lines for these operations can be decoded directly from the bus out lines. Request TIE (track in error) requires execution time to obtain one byte from channel but because it uses the no-op circuits, no command storage is needed.

There is an exception to the command register bit pattern in the data transmission commands. All the data commands, except read backward, set the 5, 6, and 7 bus out bits into bits 5, 6, and 7 of the command register. Read backward has no 6 or 7 bit in the command but the 6 and 7 bits are forced into the command register along with a 5 bit from the bus out lines. A read backward command, when displayed, has 5, 6, and 7 on. This combination appears similar to a "forward space file" command; however, the P bit is not on for read backward.

When the command register is displayed in the selectable register lights on the CE panel, positions 0, 1, 2, 3, and 4 are the mode register. Positions 5, 6, ant 7 are the command register. Figure 101 provides a list of tape control command formats. Figure 101 also indicates which positions of the command byte are set into the command register for each format type.

Command register indicators P, 5, 6, and 7 are not attached directly to the command register. These indicators are driven by a group of command register "memory" latches. The indicator latches are shown on the right-hand side of Figure 17.

The indicator latches were added to retain the command register setting after the command has been completed. With the indicator lights connected directly to the command latches, the CE panel lights are reset at the end of each command. (The command register must be reset to drop gating lines to the tape control circuits.) The indicator latches retain the command bits until the beginning of the following command.

Note, in Figure 17, that the gating line to transfer the command into the indicator latches is the tape op trigger. If a command does not turn on the tape op trigger, it will not be displayed on the CE panel. Commands which do not set the tape op trigger are:

1. Test I/O
2. Sense
3. No op
4. Request TIE
5. Diagnostic
6. Mode set

The remaining commands normally set the tape op trigger and are displayed on the CE panel. In some cases, these commands may not set the tape op trigger because of unusual conditions. For example, if the initial selection status byte of a write, read, or read backward command is rejected by a command out response from the channel, the tape op trigger remains off. Tape op is not set if the channel does not send the first data byte for a write command (word count zero condition). Motion control commands are not displayed if a "busy" or a "unit check" condition occurs during initial selection.

Commands that are displayed, therefore, are:

1. Read, read backward, or write (with no unusual conditions).
2. Valid control commands (motion control commands with no busy or unit check condition).

The indicator latches are reset by the "set command not sense" line; this line is active only during command out time of initial selection. For this reason, the latches retain the command setting after completion of the command. The latches are not reset until just before a new command is accepted by the control unit.

Several other conditions can block the reset line and retain a command in the indicators. The same commands that do not set the tape op trigger also block the reset line:

1. Test I/O
2. Sense
3. No op
4. Request TIE
5. Diagnostic
6. Mode set

Since these commands block the reset line to the indicators, the previous command remains in the indicator lights during their execution.

Conditions for setting and resetting the command register indicators apply to both on-line and off-line operations. Commands performed off-line from the CE panel enter the tape control circuits from the bus out lines as if they were received from the channel. Therefore, commands appear the same from the CE panel or when operating on-line with a data channel.

Mode Register

- **The mode register establishes operating conditions or modes for seven-track operations.**

- **The mode register is not used for nine-track operation. A standard nine-track mode is forced.**
- **The mode register is set from bus out lines 0, 1, 2, 3, and 4 on a mode set command.**
- **Mode register is set to 800 BPI (position "0" on), data convert mode active, and odd redundancy (position "3" on) via system or machine reset.**

Bits in the mode register establish operating conditions, or modes, for seven-track operation. A mode set command sets the mode register during initial selection. Because no execution time is required, the mode set command does not set the command register.

Nine-track operations do not alter the contents of the mode register; they merely ignore it. Subsequent seven-track operations will again be under control of the mode register. Any number of operations, both seven- and nine-track, can be performed in any sequence. Bits will remain in the mode register until the next mode set command unless the control unit is reset.

Mode conditions are listed in Figure 101 under "mode set." Only a mode set command can set the mode register (Figure 18). During a mode set operation, bits 0, 1, 2, 3, and 4 of the bus out lines are gated into positions 0, 1, 2, 3, and 4 of the mode register.

Bits 0 and 1 of the mode register are decoded to determine the density of bytes on tape for seven-track operations. The decoded densities are:

BITS	DENSITY, BPI
0 0	200
0 1	556
1 0	800
1 1	800

Densities of 200 BPI and 556 BPI are gated by seven-track. In nine-track operation the only density used is 800 BPI.

Bit 2 of the mode register controls the data convert feature for seven-track operation. When position 2 is on, the converter is turned off. Either read backward or nine-track operation can also turn off the converter.

Bit 3 of the mode register specifies the parity of seven-track operations. A bit in 3 will bring up odd redundancy mode. No bit in 3 drops odd redundancy (even redundancy). Nine-track operations are all performed in odd redundancy mode.

The translator is controlled by the 4 bit of the mode register. A bit in 4 turns on the translator; no bit in 4 turns off the translator. Nine-track operations do not use the translator.

The mode set chart in Figure 101 illustrates how bus out positions 0, 1, 2, 3, and 4 of a true mode set command will affect seven-track operations. Only a true mode set command or reset can alter the contents of the mode register.

A system or machine reset will turn on positions 0 and 3 of the mode register. Seven-track operations with the mode register in the reset condition (0 and 3 on) will be performed in data convert on mode, odd redundancy, and 800 BPI density.

Longitudinal Redundancy Check Register (LRCR)

- **The LRCR is a circuit to check the longitudinal parity of each track in a block.**
- **The LRCR contains one binary trigger for each of the nine tracks.**
- **The triggers are complemented each time a bit is read in the corresponding track.**
- **At the end of a block all triggers should be off.**

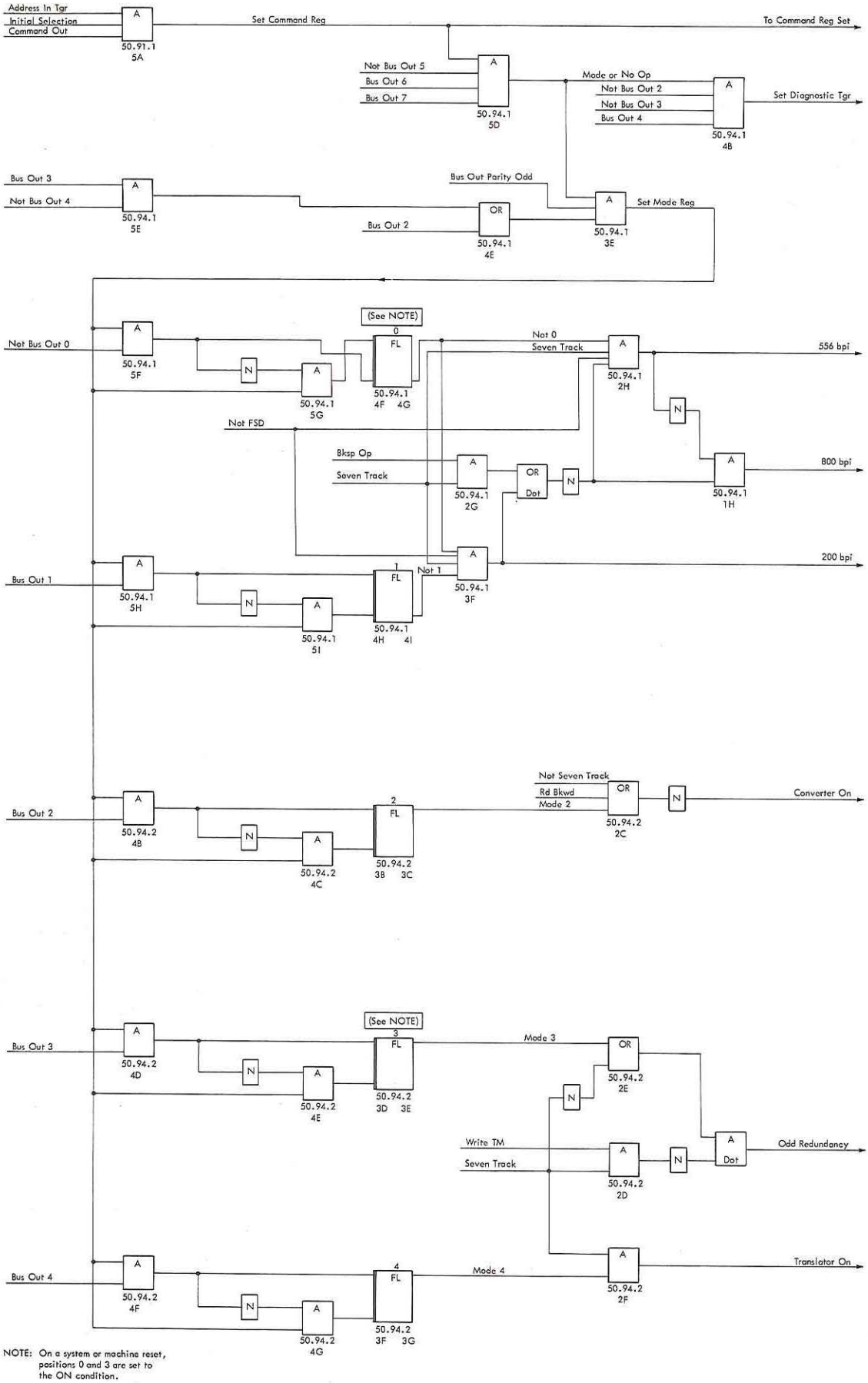
The LRCR is an error detection circuit to check the longitudinal (lengthwise to the tape) parity of each track in a block. The total number of bits in any track of a block should be even. LRCR triggers turn on or off for each bit in the corresponding track. At the end of the block all the triggers should be off. If an LRCR trigger remains on at the end of a block, an error is indicated.

The LRCR does not generate the LRC character during write operation. LRC characters are generated by resetting the tape unit write triggers left on at the end of each block. The LRCR checks bytes read from tape for errors on read, read backward, and write (read checking).

If the total number of bits written in a track (including the CRCC) is odd, the write trigger will be on at write LRCC time. That trigger will be reset to write a bit in the LRCC. If the bits in the track were even, the write trigger would be off at LRCC time and no bit would be written in the LRCC for that track.

As the block passes the read head, each bit complements a trigger in the LRCR. Since the triggers started from a reset condition, they should be off after an even number of bits. If each complete track including both check characters has an even number of bits, the LRCR will be blank at the end of the block.

A checking circuit (Figure 209) samples the LRCR at RDD169 time. If no bits are on, the LRCR error is not turned on; a bit left on in the LRCR will not always signal an error. A bit in the error pattern register (EPR) will prevent an error indication from the corresponding LRCR bit. This is done primarily to prevent false errors during error correction. In an error correction read operation, the LRCR receives bytes before they are corrected. The EPR bit prevents false LRC errors from



NOTE: On a system or machine reset, positions 0 and 3 are set to the ON condition.

● Figure 18. Mode Register Set

the track being corrected but the rest of the LRCR bits are checked normally.

When a true LRCR error is detected, the LRCR trigger turns on. The error condition also turns on data check and unit check.

Lost Character

- **Lost character detection is a circuit to detect loss of a complete byte while reading tape.**
- **Lost bytes can be detected only during seven- and nine-track, read-type operations.**
- **If the time between bytes is longer than normal but not long enough for check character spacing, a byte is assumed to be lost.**
- **Lost character indirectly causes a read/write VRC error by generating a shift data pulse when the read/write register is blank.**

The lost character detection circuit monitors seven- and nine-track read-type operations to detect a byte (character) loss. A single bit byte can be lost completely if the bit is dropped. The result is a greater than normal spacing between bytes. The lost character circuit detects this longer time and indirectly causes a read/write VRC error. If a read clock 0 pulse occurs during the time between RDD17 and RDD30, a lost character is signaled (Figure 331, coordinates 1B). After RDD36, the next byte is assumed to be a check character.

A lost character cannot be detected if it was the first or last byte in a block because the detection circuit relies on measurement of time (spacing) between normal bytes.

Recognition of a lost character condition causes four separate actions:

1. A shift data pulse, Figure 408, is generated, which causes a read/write register VRC error because the read/write register is empty.
2. A byte with only a parity bit is sent to the data channel.
3. The character count circuit is stepped to allow for the missing character.
4. The shift pulse also steps the error correction circuits.

The lost character circuit is primarily for error correction use. A single bit byte that is lost can be corrected if the loss is detected at the time it occurs and any other errors are in the same track within the block. Error correction circuits depend on a shift pulse and a read/write VRC error to locate the track in error. The lost character circuit provides the needed pulses even when a byte is completely missing.

Translator

- **For write operation the translator translates eight-bit code to six-bit code.**
- **For read operation the translator translates six-bit code to eight-bit code.**
- **The translator can be used only for seven-track operation.**
- **In seven-track operation, the mode register bit 4 controls translator operation.**

The translator is a group of circuits to translate back and forth between the eight-bit, IBM System/360 code (nine-track) and the six-bit (BCD) code used on IBM seven-track tape systems. In nine-track operation, the translator is automatically turned off. Tape control units without the seven-track feature will not have a translator.

When operating with a seven-track tape unit, the translator is switched on or off by bit 4 of the mode register. A bit in mode register 4 turns the translator on; no bit in 4 turns it off.

The translator is in the data path between the data register and the read/write register (Figure 100). When the translator is inactive, bytes transfer without modification. When the translator is active, bytes are translated by AND and OR circuits. Figure 407 includes the translation circuits and a chart of the six-bit and eight-bit codes.

During seven-track write operation, the translator changes eight-bit code bytes to six-bit (BCD) code bytes. During seven-track read operations, six-bit characters from the tape unit are translated into eight-bit bytes to be sent to the data channel.

Not all of the bits in a byte are modified by the translator. Bits 5 and 7 of the read/write register are connected directly to positions 10 and 12 of the data register. These positions enter the translator also. In the translator they are used for coding and decoding other bit positions. Bits transferred in either direction between read/write register 5 or 7 and data register 10 or 12 are not changed.

Read/write registers 5 and 7 contain bits 4 and 1 of a six-bit (BCD) character. A comparison of these bit positions on the code chart in Figure 407 will show that they remain the same in six- or eight-bit code. In seven-track operation, positions 0 and 1 of the read/write register are not used. Positions 0 and 1 of the read/write register remain in the circuit for seven-track and are gated out; however they should not contain data. Figures 307 and 308 illustrate the translator data flow.

Tape Unit Select Register or Device End Scanner

- The device end scanner will provide:
  1. One of eight TU select lines (TU addressing).
  2. A selected device end bit for the status byte.
  3. TU address to be inserted on "bus in 5-6-7" positions during "address in" time of a multiplex entry.
  4. Cause a multiplex interrupt when applicable.
- The device end scanner circuit allows the three TU select triggers to step or count as a counter which will then step through eight TU selections repetitively.
- When the scanner is *inactive*, one of the eight TU select lines is active.
- The device end scanner *cannot* step the three TU select triggers when TCU is performing a command with a selected TU or TCU has "stacked" status from a previous operation.
- Scanner circuits contain one "arming" latch for each tape unit which, when set, will allow the three TU select triggers to step *to* each TU address, stop stepping and provide a device end for that particular TU.
- Each "arming" latch can be set if the selected TU is:
  1. Not ready.
  2. Ending an operation.
  3. Busy due to rewinding.

The tape unit select register or device end scanner is a dual-purpose circuit. During an operation with a tape unit, the select triggers contain the address of the selected tape unit. A decoder circuit analyzes the three triggers (four triggers with SXT feature) to produce a select line for one particular tape unit. This tape unit address is retained or "locked" in the triggers by the "TU selected" trigger until the operation is complete.

The second purpose of this circuit is "scanning." When the triggers are not locked by an operation in progress, they begin to "scan" or step through all tape unit addresses. No stepping pulses are needed. Outputs of the triggers are connected to AND circuits which are gated to the inputs. The result is a circuit that runs continuously unless it is "locked" by the "TU selected" trigger or a device end latch (Figure 406).

Scanning is used primarily for rewind and rewind-unload operations. During rewind or rewind-unload, the tape control is free of the tape unit. When the operation is completed by the tape unit, a device end signal must be sent to the data channel, indicating which tape unit has finished rewinding. Since the control unit may have performed several other operations while the tape was rewinding, the tape unit address must be reconstructed. When stopped at the correct time by the TU device end circuit, the scanner provides the correct address.

The same group of AND circuits that feed the trigger outputs back to the inputs also stop the scanner. When a tape unit completes a rewind operation, one of the input AND circuits is degated to stop the scanner at that address. This operation is similar to placing a wrench in the spokes of a wheel to stop it at a given point.

While the triggers are stopped, the address is sent to the channel as part of an interrupt sequence. When the channel has accepted the information, the triggers are free to continue scanning or to accept an address for a new operation.

Scanner Functions

The device end scanner is a unique circuit arrangement that controls TU selection and the sequence in which any TU is allowed to signal a device end. The circuit is allowed to "scan" for device ends only if the channel and tape control are "free" of:

1. All tape units; that is, no TU address is held in the TU select triggers because no operation is in progress that requires retention of a TU address and,
2. TC has no stacked status; that is, no status latch is retaining some status from a previous operation performed by a particular TU.

"Scanning" is performed by the tape unit select triggers. These triggers step through tape unit addresses in a sequence similar to a counter or clock circuit. Outputs of three triggers are decoded to form eight tape unit addresses. If the control unit has the 16 address feature installed, four trigger outputs are decoded to form 16 tape unit addresses.

When the scanner is stopped, the decoded output will be used to supply the tape unit address information of the "address in" byte for multiplex entry on device end interrupts.

For operations which require both the control unit and the tape unit operating together, such as read, write, or WTM, the TU address must be "locked in" the TU select triggers so the scanner cannot scan. As soon as the operation is complete, a "selected device end" is generated by the scanner output circuits and "device end" is sent to the channel in the end status byte. There was no need to allow the scanner to "scan" on these operations.

For other operations which require only the TU to complete the command, such as rewind and rewind-unload, it is not necessary to retain the TU address in the select register. Thus the TU select triggers begin to scan (step) as soon as TC is free (TU begins to rewind). At this time the TC is free to perform another command on a different TU and now the TU select triggers are utilized to select another TU address. The device end status byte must be submitted to the chan-

nel via multiplex entry when:

1. the TU completes the rewind operation, or
2. an operator makes the TU ready after the rewind-unload is complete.

Scanning is necessary because there is only one set of TU select triggers, although more than one TU could have a "device end" condition pending.

Two other conditions utilize the scanning function of the select triggers: if a "not ready" TU is selected, or a newly loaded TU is rewinding back to load point when TC is attempting to select it for initial selection. Under these conditions, the TU address is lost when the select register is allowed to scan after initial selection. However, for both conditions, the TC must supply a device end status via multiplex entry when the TU becomes ready (available). Thus a "remembering" device knows to stop the scanner at the correct TU address when these tape units become ready. Again, the select triggers will supply the TU address information of the "address in" byte for the multiplex entry.

Once the channel takes the device end status, the scanner advances and stops at the next TU address indicating "ready" (if any). This procedure is repeated until all "device ends" are accepted by the channel, or channel initiates a general reset.

Device End

"Device end" generally means an I/O device, in this case the TU, has completed (come to the end of) its part of an operation. The channel and system are advised of this with a "device end" bit in the end status byte. To simplify programming procedures, a "device end" bit is also included in the end status byte after tape non-motion operations. Specifically, a "device end" status bit is sent to the channel after:

1. Any command, other than test I/O. However, a rewind-unload causes "device end" when the TU begins the operation, also when (and if) the TU is manually reloaded and made ready again after the operation.
2. A TU becomes "ready" following its selection when it was switched, rewinding, or not mechanically ready. During initial selection, a TU might be switched to another TC (2816 Switching Unit), rewinding (manual), or not ready (relay 101 not picked). An attempted selection "arms" the device end circuits in TC to prepare for the ready (device end) condition.

A device end bit is usually generated by a line called selected device end; this line is produced by the device end scanner circuits. However, under some conditions, device end is "forced" by the on state of the channel end trigger and:

1. Rewind-unload trigger (in TC) for the first device end status byte of a rewind-unload operation.
2. "Not selected and ready" line (model line inactive from TC). For example, a sense command must designate a TU address even if the addressed TU is not ready; a "device end" is generated anyway at the end of the operation.

Scanner Circuits Description

In the broadest sense, the device end scanner (Figure 406) consists of the following circuits:

1. Three *tape select triggers* (four triggers with SXT feature) must "lock" to select a TU.
2. One TU selected trigger.
3. Eight device end triggers (16 with SXT), one for each TU.
4. Eight rewind triggers (16 with SXT), one for each TU.
5. TU select (decoder) circuit, which translates the tape select triggers output to one of eight (or 16) possible TU addresses.

Tape Select Triggers: These triggers are the "heart" of the scanner; scanning actually means these triggers are stepping at a 10 megacycle rate, that is, each trigger is rapidly set and reset in a specific sequence. Each TU address is "selected" once during each complete continuous cycle.

TU Selected Trigger: This trigger is the "key" to the circuit; the scanner cannot begin to run unless this trigger is off. The trigger is always left on when the selected TU must remain selected as in the case of those commands requiring TC and TU operating together (read, write, etc.). Thus the TU select triggers (the heart of the scanner) retain the one address of the selected TU.

Device End Triggers: The eight device end triggers are the "remembering" devices for the control unit to remember which of its eight tape units it must prepare a device end for. Any device end trigger is set (armed) by any of three methods:

1. By an "end pulse" which will occur at the end of all commands which require TC and TU operating together, or when the TU begins to rewind on both rewind commands.
2. By a "not ready" condition if an attempt is made to select any not ready or switched tape unit during initial selection.
3. By a "busy" condition which is the result of the selected tape unit rewinding back to load point by a manual intervention. (Operator depressing load rewind and start pushbutton.)

Rewinding Triggers: These eight triggers (one per TU) work in conjunction with the eight device end triggers. As stated above, the device end triggers are "armed" to remember that TC must submit a device end status byte. These rewinding triggers basically indicate "when" the device end should be submitted. The OR half of a rewinding trigger has an output if its

associated tape unit is still rewinding or not mechanically ready. Thus, when the rewinding or not ready condition no longer exists and because the associated device end trigger is armed, a device end bit is generated immediately if the scanner is not stepping or immediately upon stopping the scanner at the correct TU address. In addition, these triggers are allowed to "latch up" only if the TU is "rewinding or not ready" during initial selection. This is to inhibit the TU status from changing while it is being interrogated during initial selection.

TU Select (Decoder) Circuit: This circuit consists of a number of AND circuits that translate the TU select triggers output into one of eight (or 16) TU addresses.

Selected Device End

A "selected device end" is generated at the output of the scanner circuit when three conditions exist:

1. A TU is selected (scanner stopped).
2. The selected TU device end trigger is armed.
3. The selected TU "rewinding" line (OR output of rewinding trigger) becomes inactive – the TU is ready.

For most operations, the scanner cannot step and the TU remains selected. The TU is not rewinding so "end pulse" arms the device end trigger to generate "selected device end." However, if the scanner had been stepping, the dropping of the TU rewinding (ready) line generated the "selected device end" as the scanner stopped to select the TU; the device end trigger was armed previously.

Scanner Stepping

The scanner steps continuously if the TU selected trigger is off and no TU has a device end (ready) condition. The AND circuits that gate the sets and resets to the tape select triggers are conditioned by the rewinding and device end trigger outputs, as well as the tape select triggers themselves. As soon as an armed device end trigger output AND's with "not rewinding" for a particular TU (Figure 406), either the set or reset of a tape select trigger is degated and the scanner stops. For example, if the device end trigger for TU2 is armed (set) and the TU2 rewinding line drops, these conditions AND to decondition the set to tape select trigger 5 (AND at 1D, Systems 50.42.1) with the 5 trigger off, the scanner stops with only the 6 trigger on which selects TU2. The scanner selection (stepping) sequence is 0, 1, 3, 2, 6, 7, 5, 4, 0, and so on.

The unit address in the TU select triggers supplies unit address bits for the address sent to the data channel. When channel has accepted the "address in" sequence and status byte it responds with a service out line. Status in and service out form a reset line to the device end scanner circuits on Figure 406. The device end trigger for that particular tape unit address will be reset.

The only device end trigger reset will be the one corresponding to the address in the TU select triggers. Other device end triggers remain on waiting for a device end sequence for their address.

Simultaneous Command Interference Circuits

- **The 2404/2804 simultaneous R/W tape control units connect one group of tape units to two data channels.**
- **Command interference circuits prevent conflict between data channel commands.**
- **If there are no conflicting conditions, the command interference circuits generate gate lines to "lock" the sections of the tape control in a particular configuration.**
- **When a command attempts to select a section in use, the interference circuits generate a busy signal to the interface that attempted to break in.**

Command interference circuits in the 2404 and 2804 tape controls prevent conflict between commands from the two channel interfaces. The heart of the command interference circuits is shown in Figure 19. For identification purposes, the two interfaces have been designated A and B.

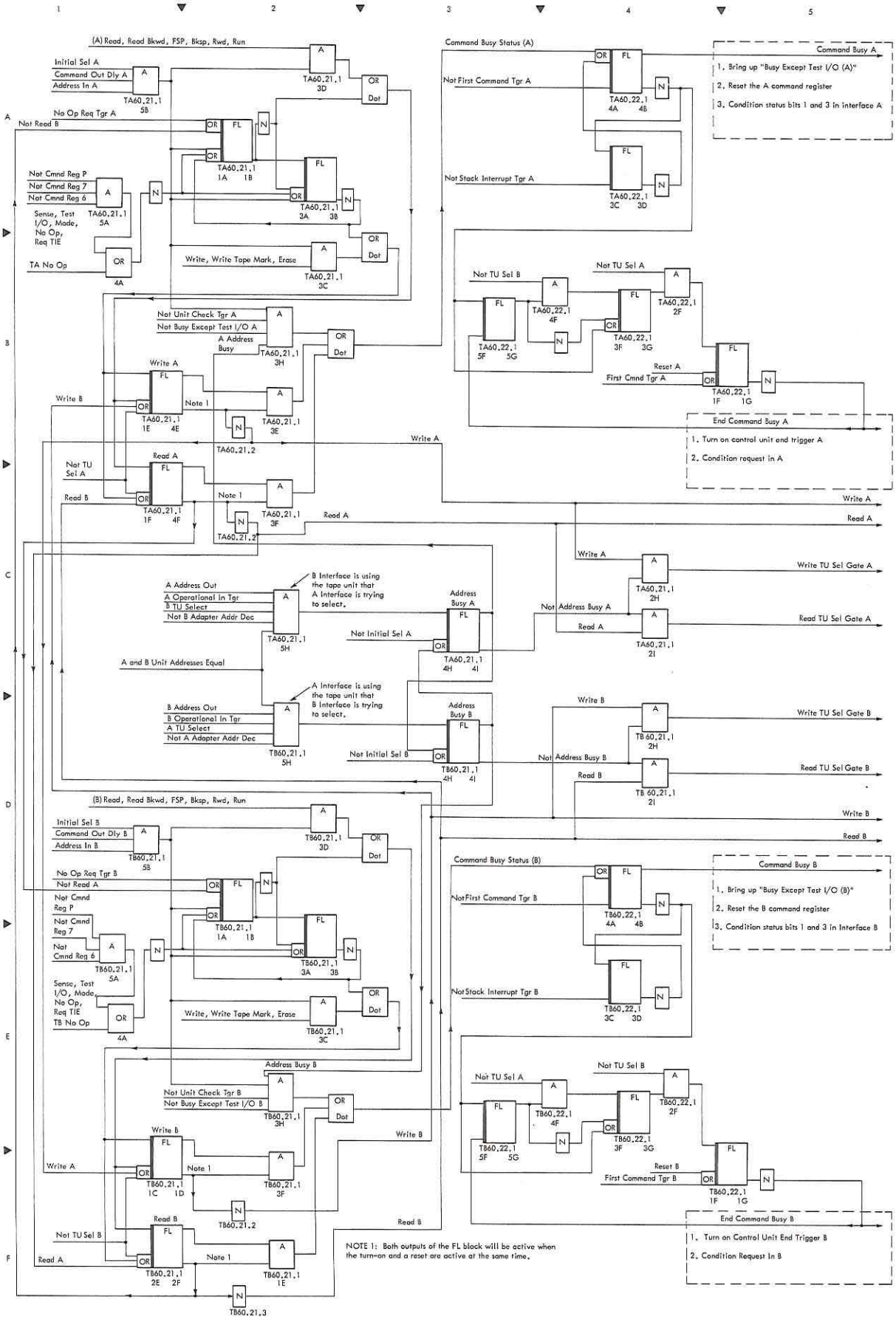
A read or write command from an interface will attempt to set the correct trigger on Figure 19. If no interfering conditions exist, the trigger will turn on and remain on until the operation is completed. Outputs of the triggers control gating circuits to connect the proper interface lines to the desired operating section (read or write).

Both interfaces can operate at the same time provided they do not try to select the same control section or tape unit. If a command attempts to select a section or tape unit being used by the other interface, the selection will be answered with a busy status. When the section and tape unit are no longer busy, the channel that attempted to break in will be notified with a "control unit end" bit in a status byte.

Selection

Command interference circuits for interfaces A and B are almost identical. The top half of Figure 19 shows the A circuits and the bottom half shows the B circuits. For illustration purposes, assume that a write command is received from data channel interface A while interface B is inactive.

Initial selection of interface A brings up the AND circuit in 1A of Figure 19. Output of this AND, along with write, brings up another AND circuit in 2B. Out-



● Figure 19. Command Interference A and B 2404/2804

put of the second AND circuit will turn on the write A trigger and turn off the read A trigger, 1B. The write A trigger remains on because the reset line (write B) from the B interface is inactive. An output line, 5C, from the write A trigger gates the circuits to connect the A interface to the write control section.

Along with the write command, a tape unit address is specified. The AND circuit in 2C checks to see if the B interface is using the tape unit. If the tape unit is available, the address busy A trigger, 3C, remains off. The write A trigger line and the not address busy A line bring up the write TU select gate A, 4C, which gates the tape unit select information from the A interface to the tape units.

Connections are now complete to write information on a particular tape unit from the A interface. The command interference circuits will keep the control unit "locked" in this configuration until the write operation is completed on interface A. When the operation is ended the reset line (not TU select A), 1C, resets the write A trigger. The write A line will drop to disconnect interface A from the write control section and the tape unit.

Command Busy

While interface A is performing a write operation, the B interface is prevented from selecting the write control section. During the A write operation the write A trigger is on, 1B. An output line from the write A trigger holds the write B trigger reset, 1E. If the B interface attempts to do a write operation through the AND circuit in 2E, the command busy B line, 5D, will be activated.

Both the "in phase" and "out of phase" outputs of the write B trigger feed the same AND circuit, 2F. At first glance it seems the AND circuit could never be activated. Because the FL block is actually an OR block and an AND block in a "latched" configuration, both outputs can be active at the same time. Both outputs are active when the B interface tries to select the write control section and the A interface is using the write control section. The AND circuit in 2F activates the command busy status B line, 3D, which generates a busy response to the B interface at the end of initial selection.

Command busy B activates:

1. Busy except test I/O (B).
2. Reset to the B command register.
3. Status bits 1 and 3 in interface B.

When the B command register is reset, the B interface drops the line that is trying to turn on the write B trigger. Only the off output of the write B trigger will be active, so the command busy status B line will drop.

The command busy B trigger in 4D retains the command busy B conditions until the B first command trigger is reset by a status in sequence. If the channel rejects the status byte by returning a command out, the stack interrupt trigger will be turned on. A trigger in 4E retains the command busy B condition until the stack condition is cleared.

The read B, write A, and read A triggers provide corresponding busy responses the same way the write B trigger does. These four triggers are the heart of the command interference circuits.

Address Busy

Command busy (A or B) can be signaled if the two interfaces attempt to use the same tape unit. If the A interface is using a tape unit (read or write) and the B interface tries to select the same tape unit (read or write), the address busy B trigger, 3D, is turned on by an AND circuit in 2D. Output of the address busy B trigger blocks the TU select gate "B" and activates the "command busy B" line, 5D.

Simultaneous R/W Operation

Both interfaces can operate at the same time if they do not try to use the same control section or tape unit. Refer to Figure 19 and assume that interface A is processing a write-type command. The write A trigger is on in 1B. The write A and write TU sel gate A lines are active to "lock up" the configuration.

A read-type command from interface B activates an AND circuit in 2D. Output of the AND circuit will try to reset the write B trigger (should not be on) and set the read B trigger. None of the resets to the read B trigger are active at this time, so the read B trigger will be turned on. Since both the set and reset to the read B trigger were not active at the same time, the AND circuit in 2F did not bring up the command busy line.

If the tape unit address from the B interface is not the same unit that A is writing on, the address busy B trigger, 3D, remains off. Under these conditions the output lines, read B, and, read TU select gate B, will "lock" the read control section and the tape unit to the B interface without disturbing the "locked" condition of the A write operation.

End Command Busy

The command busy trigger also activates a circuit to generate "end command busy" (A or B). End command busy sends a control unit end signal to the channel when the busy condition is cleared. Three triggers in each interface check the busy condition and generate the end command busy signal.

The first of the three triggers, 3E, is turned on by "command busy" (A or B). Output of the first trigger

goes to one leg of an AND circuit. The other leg of that AND circuit is "not TU select" from the opposite interface. Output of the AND circuit turns on the second trigger, 4E.

After a "command busy B," 5D, the AND circuit in 4E will turn on the second trigger when TU select A drops. If TU select B is down, the third trigger turns on and generates the "end command busy B" signal, 5F.

If another command busy B signal occurs while the second trigger is on, the sequence will start over with the first trigger. Output of the third trigger resets the first trigger and turns on the control unit end trigger. The third trigger output also conditions request in to notify the channel that the control unit has a status condition to transmit.

Sense, Test I/O, Mode Set, No Op, or Request TIE

Some commands can be performed on either the read or write control section. Test I/O, mode set, or no-op commands involve only initial selection time of the interface section. The select status of the read and write control sections has no bearing on these operations.

Sense

A sense operation must select a control section because part of the information sent to the data channel must come from the control sections. Assume that a sense command is received from interface A while the B interface is performing a read command. The sense command (not P, not 7, and not 6) activates an AND circuit in 1A of Figure 19. Output of the AND circuit allows the two triggers in 2A to latch on if set.

The first trigger will not be set because the B interface is performing a read operation. Not read B or no-op request is needed to set the first trigger. The second trigger turns on because the first trigger is off. Output of the second trigger holds the first trigger reset and turns on the write A trigger, 1B.

If the read control is not busy, the sense command turns on the read (A or B) trigger. If the read control section is in use, the sense command turns on the write (A or B) trigger unless "found track" has held the read section because of errors.

If, during a read operation, errors occur and the TIE is found, then the found track trigger will keep the read control section attached to the interface by bringing up a "chain" condition. This is to keep the other interface from destroying the TIE information in the EPR. The read control section will remain connected to the interface until a sense command is performed to move the TIE information to the data channel. Another command on the same interface can also break the "chain," however, the TIE information may be lost.

Information bits for the five bytes of a sense command (Figure 8) come from three possible sources: interface circuits, read control, or write control. A sense command on interface A will use interface A circuits for bytes 1, 4, and 5. These bytes represent the conditions left by the previous operation. Byte 3 can only come from the read control section because it has the only EPR for TIE information.

Byte 2 bits 1 through 6 provide tape unit information regardless of which control section is selected. Bit 0 or byte 2 is from the interface circuits and does not depend on control section or tape unit selection. Bit 7 of byte 2 can only come from the write control section.

Request TIE

A request TIE instruction must select the read control section to put a TIE bit in the EPR. The read section must be used because error correction can be performed only during read operations. The write section has no EPR to accept a TIE bit.

To insure that the read section is selected, the no-op request trigger turns on the trigger (Figure 19) which, in turn, sets the read A or read B trigger. Whichever interface decodes the request TIE command will be connected to the read control section provided the other interface is not using the read control section. If the read section is busy, a "command busy" condition will be generated. When the control unit is no longer selected, a device end bit is sent to the channel, in an interrupt status byte, to indicate that the request operation can now be performed.

Simultaneous CE Panel Light Switching

A 2404 or 2804 has only one CE panel for four groups of circuits. CE panel indicator lights must be switched from one circuit to the other to display all of the desired information. In the lower right corner of the CE panel (Figure 20) are two jacks labeled A and B. A plug in the A or B hub will connect the corresponding interface to the CE panel. Not all of the panel lights are controlled directly by the A and B plugs.

Those indicator lights which are connected directly with interface circuit operations are switched by the A and B plugs. Indicators connected with the read and write control sections are switched by a latch circuit controlled by the A and B jacks and interface operations.

Lights marked A/B in Figure 20 are switched by the A and B jacks. Lights marked R or W are associated with the read or write control sections. Lights marked R/W can represent read or write control section conditions, depending on the condition of the indicator latch circuit. When the latch is in the read state, indicators in Figure 20 marked R or R/W display read control section information. When the latch

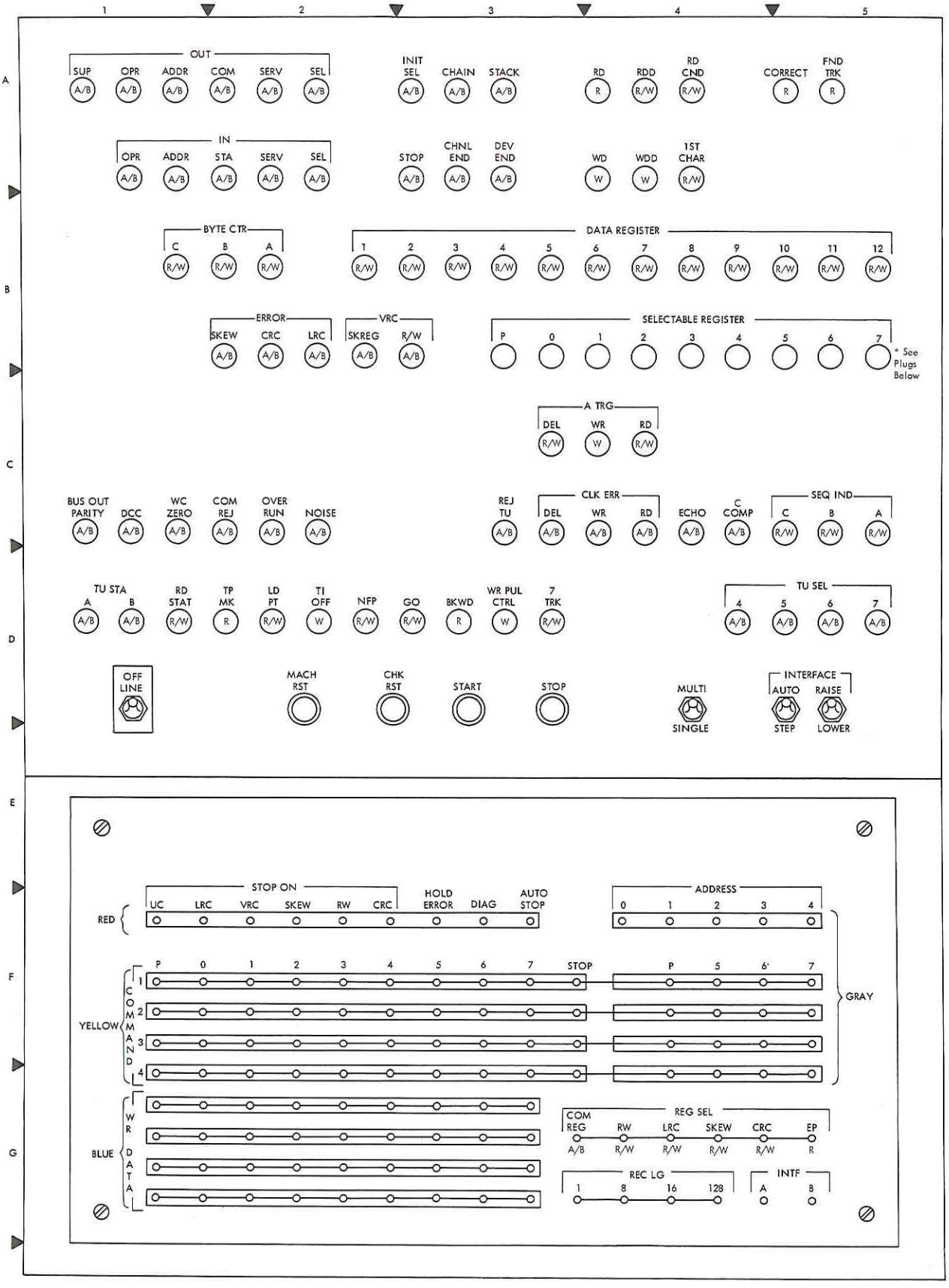


Figure 20. CE Panel Lights 2404/2804

is in the write state, indicators marked W or R/W display write control section information. Indicators marked R or W can be on only when the latch is in the corresponding state.

Control of the latch circuit is the key to indicator displays. The state of the indicator latch can be changed only if an operation occurs while the corresponding interface (A or B) is jack plugged. A read operation on the A interface can change the indicator latch to the read state only if the A jack is plugged. Moving the jack plugs has no effect on the indicator latch until an operation occurs to switch its condition. Moving the jack plugs will change the A/B indicators, however.

When a jackplug is moved from A to B or from B to A, the A/B indicators change condition but the R/W indicators remain connected to the control section used by the previous operation on the jackplugged interface. The R/W indicator latch circuit remains in one state until an operation is performed on the interface that has its jackplug in place on the CE panel. If the new operation selects the same control section, the indicator latch does not change state.

During CE panel operations on the A or B interface, the opposite interface may still be on line. If the A/B jackplug is moved, the R/W indicators may be indicating on-line operations that happen to use the control section left by the indicator latch circuit.

Off-Line Command Interference

When either the A or B interface is placed off line from the CE panel, the command interference circuits are still operating. Operations performed on the off-line interface can directly affect the opposite interface which is still on line. An off-line (CE panel) operation selects the read or write control section and a tape unit preventing the on-line interface from selecting either the control section or that tape unit. The off-line interface also has access to the same group of tape units that the on-line interface is using. CE panel operations can alter or destroy tape information written by on-line operations.

Tape Mark Detection

- **Any true tape mark character, seven- or nine-track, will contain two and only two characters with identical bit configurations.**
- **A tape mark character can only be recognized if the *first* character appears to possess a tape mark configuration.**
- **Tape mark recognition is active both on read and read backward operations.**
- **Tape mark recognition circuits check for:**
  1. **Proper bit configurations.**
  2. **Data character and *one* check character.**
  3. **No more than two characters.**
- **Recognition of a true tape mark will set "unit exception" (bit 7) of status byte during:**
  1. **Read forward or read backward.**
  2. **Backspace or forward space *record*.**

Tape marks are written on tape to separate groups of data blocks. During subsequent read operations, tape marks must be detected to identify data boundaries. Forward and backward reading and spacing operations must be able to correctly identify the difference between tape marks and data or check characters that appear to be tape marks because of their bit configuration. Most of the tape mark circuits are used to separate true tape marks from false ones.

First Character Tape Mark in Forward Read

The first character transferred to tape control in a read operation causes final amplifier tracks to produce high- and low-clip outputs that set corresponding high- and low-clip skew register positions. The first high-clip output from any final amplifier track sets the first bit trigger, causing the read clock to start.

The first character trigger, set during the initial selection sequence, is on only during the first read clock cycle in the operation, allowing the read clock output at RC-6 time to check the character in the high-clip skew register for the tape mark bit configuration (Figure 333, coordinates 4 and 5C).

A tape mark, in nine-track, is composed of bits in positions 3, 6, and 7 and no bits in P, 0, 1, 2, 4, and 5. A tape mark, in seven-track, is composed of bits in positions 4, 5, 6, and 7, and no bits in P, 2, or 3. Tracks 0 and 1 are not used in seven-track operation.

If the character in the high-clip skew register has the bit structure of a tape mark, the tape control sets the first character tape mark trigger at read clock 6 time. The tape control must now wait for the next character to see if the record is a true tape mark. The record may be data with the same bit configuration as a tape mark.

A true tape mark record contains only a tape mark followed by an LRC character with the same bit structure. Tape control holds the first character in the read/write register until the next character is received. The character spacing and bit configuration determine whether the record is a true tape mark or data (Figure 501).

At RC-7 time in the first read clock cycle:

1. Tape control sets the RDD trigger. "RDD" conditions microsecond delay counter drive circuits, and the delay

counter steps in the microsecond mode at the rate determined by the tape unit reading the record.

2. Tape control checks the character in the high-clip skew register for parity in seven-track to determine "gate out hi" on "gate out lo" to the R/W register. A nine-track tape mark is always gated from the high-clip skew register to the R/W register. A seven-track tape mark is always an even parity character; a nine-track tape mark is always an odd parity character.

At RC reset time, the tape control:

1. Resets the first character trigger.
2. Resets the first bit trigger, causing the read clock to reset.
3. Resets the high- and low-clip skew registers.
4. Sets the R/W control trigger.

Because the first character tape mark trigger is set, the R/W control trigger output does not condition "start shift" to initiate a read/write register to data register character transfer, as shown on Figure 408.

If the delay counter advances to 36 before the next tape character produces a high-clip output from final amplifiers, tape control sets the first check character trigger. The tape mark character stored in the read/write register on the previous read clock cycle does not transfer to the data register.

If the second character in the tape record arrives at the final amplifiers and causes the read clock to start before the delay counter advances to 36, the record is not a true tape mark record. Although the first character in the record (now stored in the read/write register) has the bit configuration of a tape mark, tape control must transfer that character to the data register. Tape control must also check and transfer the second character in the record (now in the skew registers) to LRCR and through the read/write and data registers to channel before the tape unit transfers the third character to tape control. If this is a data conversion read operation, tape control transfers only one byte to channel during this character cycle. If this is a standard read operation, tape control transfers two bytes to channel during this character cycle. Tape control resets the first character tape mark trigger at RC-0 as quickly as it detects that the record is not a tape mark, and begins action to transfer the first character from the read/write register to the data channel.

The character in the read/write register is assumed to be data because the record is not a tape mark. The first character was not shifted to the data register because the tape mark trigger blocked the start shift circuit (Figure 408). When the tape mark trigger is reset at read time clock 0 time, a shift data occurs to transfer the first character.

Tape control processes the second and succeeding characters in the tape record at normal times. The condition in which the first character in the record has the bit configuration of a tape mark but is not followed by a check character is one case that requires tape control to execute two character transfers in the second read clock cycle; one at RC-0, and the other at normal read clock reset time.

Tape Mark-Read Backward

During read backward, check characters can appear to be tape marks. The bit configuration of a check character could be the tape mark configuration and check character spacing is the same as tape mark spacing. To detect a true tape mark condition, tape control must rely on detection of a third byte or data spacing between bytes.

The second check character trigger is the key to detecting false tape marks because of data spacing during read backward. If a read clock 0 pulse occurs while the second check character trigger is on, the tape mark trigger is reset. The second check character trigger remains on from read clock 2 until RDD36. A read clock 0 pulse occurring during this time indicates that the bytes on tape are too close together to be a tape mark and will reset the tape mark trigger. Detection of a third byte via the unlabeled latch (Figure 333, coordinates 5C) definitely indicates false tape mark and will also reset the first character tape mark trigger.

At the end of a tape mark record with "file search" inactive, the unit exception trigger is turned on. First character tape mark, RDD-169, and not file search, turn on unit exception. The file search line is brought up only for backspace file or forward space file operations. Unit exception indicates that a tape mark was detected when the tape control was not expecting to find a tape mark.

Initial Selection

- **All commands performed in the tape control units begin with an initial selection sequence.**
- **Initial selection accomplishes four basic functions:**
  1. **Selects the correct control unit and tape unit from the available group.**
  2. **Confirms, from control unit response, that the correct unit has been selected.**
  3. **Transmits a command to the control unit.**
  4. **Status of the control unit and tape unit is transmitted back to the channel.**
- **The status byte, returned at the end of initial selection, informs the channel whether the selected units are able to perform the command.**

An initial selection sequence is the beginning of all commands performed in the tape control units. When a data channel has a task for a control unit to perform, an initial selection sequence transmits instructions to the control unit. Initial selection also transmits status of the control unit and tape unit back to the data channel (Figure 500).

Figure 500 is the flow chart for initial selection. Figure 7A illustrates the chronological sequence of interface lines. "Objectives" on Figures 300, 301, and 302 provide a guide to the description of the circuits.

Address Out and Select Out

- **The channel seeks the I/O control unit via "address out" tag and an address byte which contains the desired address.**
- **Control unit address occupies bus out positions 0 through 4.**
- **Tape unit address occupies bus out positions 5, 6, and 7.**
- **Control unit checks parity of address byte for odd number of bits.**
- **To determine correct TC address, a match is made between bus out 0 through 4 and pre-wired address card in TC.**

The first objective of initial selection is to pick out the correct control unit and tape unit. To accomplish this the channel activates the interface bus out lines with a byte containing the correct addresses. Bits 0, 1, 2, 3, and 4 designate the control unit; bits 5, 6, and 7 designate a tape unit attached to that control unit. If a control unit has the 16 address feature, bit 4 of the address byte is used as part of the tape unit address.

To identify the byte on the bus out lines, the channel activates an interface tag line called address out. When address out is active, control units attached to the channel examine the byte on the bus out lines. Address decoder circuits are shown on Figure 300. At coordinates 2C of Figure 300, the byte on the bus out lines is checked for odd parity. Address decoder circuits check the bit structure to determine which control unit is to be selected.

Address Decoders

Each tape control unit is assigned an address at the time of installation. No two control units have the same address on any one data channel. The assigned address is wired into circuit cards that can be changed easily if the system is reconfigured. Each bit position of the control unit address decoders is wired to accept an up or down level from the bus out lines. If the but out bits match the decoder configuration, an active output gates the control unit circuits to respond.

Control Unit Addressed

During address out of initial selection the address decoder is sampled to see if the bus out bits match the assigned control unit address. An AND circuit, at coordinates 3C of Figure 300, can be activated if all of the following conditions are active:

1. Select in circuit is down.
2. Address out tag is active from the channel.
3. Bus out parity is correct (odd).
4. Bits on the bus out lines match the address decoder configuration.

If all four conditions are active, the AND circuit produces a line called "control unit addressed." Control unit circuits have determined that the channel is trying to select this specific control unit. If the address bits did not match, the control unit would not respond. The address byte on the bus out lines would be examined by all of the attached control units but only one could respond to an address out.

Select Out

Shortly after the rise of address out, the channel activates select out. The select out line is sent to each control unit in turn. If a control unit is not active, the select out signal is propagated to the next control unit in sequence. Select out establishes a priority sequence

to separate control units that try to communicate with the channel at the same time. During initial selection, priority is not needed because only one control unit can respond. Priority is used primarily during multiplex mode operations when several control units can be operating at the same time. If a control unit is ready to communicate with the interface, it will accept select out and respond to the channel. If the control unit is not ready to communicate, the select out signal is sent to the next lower control unit in the priority sequence.

In the case of initial selection, the select out signal arrives shortly after address out. The tape control has decoded its own address and therefore is ready to begin operations.

Tape Unit Address

Bits 5, 6, and 7 of the address byte designate which tape unit is to perform the command. These bits must be stored in the tape unit select triggers. Outputs of the TU select triggers are decoded into one of eight (16 with the SXT feature) possible select lines to activate a tape unit.

A line called "adapter address decoded" is generated (Figure 300, coordinates 4D). "Adapter address decoded" is used (Figure 301) to gate the tape unit address bits into the tape unit select triggers. In Figure 301, coordinates 2A, the "set tape unit address" line gates the bus out bits into the TU select triggers. The same line also fires the "TU lines valid" singleshots. For 10 microseconds the singleshots block control unit operations. The delay is to allow the tape unit time to respond to the select line and generate a "start reset" to reset control triggers and latches.

Address In and Operational In

- **The control unit "locks" into the channel via its "operational in" tag.**
- **Address in tag is sent to the channel with an address byte to confirm that the I/O device is the sought device.**
- **The address byte is transferred via bus in lines.**
- **Bus lines must always contain odd parity.**
- **Address in byte originates from a second group of pre-wired address cards for TC address (0-4) and from tape unit select register for TU address (5, 6, 7).**

Adapter address decoded (Figure 301) also turns on the address in trigger (coordinates 4A). Address in is eventually going to be used to respond to the data channel and confirm that the correct control unit has been selected. The address in trigger first gates the turn on circuit for operational in.

Adapter address decoded also sets the initial selection trigger and the tape unit select trigger (Figure 301, coordinates 5A). The initial selection trigger establishes conditions so the control unit can accept a command from the channel. The tape unit select trigger stops the device end scanner to lock the tape unit address in the tape unit select triggers (Figure 301, coordinates 3B).

Three conditions gate the turn on circuit of the operational in trigger:

1. Not reset.
2. Select out.
3. Address in trigger.

During initial selection, the operational in trigger is gated on because all three conditions are active. Operational in line is a signal to the channel that a control unit is active and will now respond with address in (Figure 301, coordinates 3B).

The address in trigger places a byte on the bus in lines (objective 6 in Figure 301). The address in byte contains the address of this control unit and the address of the selected tape unit. Bits in the address in byte should be the same as the address out byte received from the channel. Data channel circuits will examine the returned address to make sure the correct control unit and tape unit have responded.

The address in byte is constructed by circuits in the tape control unit. A group of pre-wired jumpers provide the bits for the control unit address portion of the byte (Figure 301, coordinates 5B). The tape unit address is taken from the TU select triggers, which should contain the bits just received from the channel. As the address in byte is placed on the bus in lines, the correct parity is inserted in the P position.

After a time delay to allow the bus lines to settle down, the address in tag line is raised to identify the byte on the bus in lines. Even though the address in trigger is turned on before the operational in trigger, the address in tag is not sent to channel until after operational in (Figure 301, coordinates 3B and 3C). The channel drops address out when operational in arrives. Address in cannot be transmitted back to the channel until address out from the channel has dropped.

Command Out

- **Channel informs the control unit what to do via the command byte.**
- **The command byte is checked for odd parity on the bus out lines.**
- **Either the mode register or the command register or neither is set as a result of the command.**
- **The command register bits are decoded into the type of operation to be performed.**

- **The mode register decodes lines to inform the TCU of the mode in which seven-track operations will be executed.**

After the channel has accepted the address in byte, a command byte is transmitted from the channel to the control unit. The data channel places a command byte on the bus out lines, then activates the command out tag line. Bus out and command out enter (Figure 301, coordinates 1D, objective 8). Parity of the command byte is checked as the bits arrive on the bus out lines. Command out together with three other conditions will generate a line called set command register. The three conditions are:

1. Operational in
2. Address in
3. Initial selection.

Set command register gates the bus out lines into the command register (Figure 301, coordinates 6D). During most commands the command byte remains in the command register until the operation is completed. Some types of operations do not require the command after initial selection. When the command is no longer needed, the command register is reset.

Figure 101 is a chart of command bit configurations. Different types of commands are handled differently when received. The functional unit entitled "command register" in this manual explains how bus out positions are placed in different positions of the command register. Explanations of individual command operations will be found in the "Principles of Operations" section.

Status In

- **Status in tag is the TCU response to channel's command out tag during initial selection.**
- **The status byte can be any combination of its eight bits, depending upon TCU status, tape unit status, bus out parity errors, etc.**
- **Status byte is gated to the bus in lines when status in tag is activated.**
- **Bus in circuits assign odd parity to the status byte to ensure correct parity.**
- **The channel indicates acceptance of the status byte (service out) or rejection of the status byte (command out).**

The fourth major objective of initial selection is transmitting control unit and tape unit status back to the data channel. Status information is assembled into a byte and transmitted to the channel on the bus in lines. The status in trigger, after a time delay, brings up a status in tag line to identify the status byte (objective 5 on Figure 302).

One of the requirements to turn on the status in trigger is that the first command trigger must be on (Figure 302, coordinates 2A and 3B). The first command trigger acts as a status in gate. Three conditions are required to turn on first command during initial selection:

1. Address in
2. TU lines valid (after 10 microsecond delay)
3. Command out delayed.

Address in has not been reset so it remains active. TU lines valid will become active after the 10 microseconds allowed for the tape unit to respond. A delayed command out signal supplies the third condition to turn on the first command trigger.

First command, in turn, gates the circuit to turn on status in. The status in tag line rises after a time delay to allow status bits to settle down on the bus in lines. The status in trigger assembles the status information and gates it onto the bus in lines starting during the time delay (objectives 6-8, Figure 302).

Status Bits

A status byte represents conditions in the tape control and tape unit. Status bits 0 through 7 represent the status conditions as listed in the description of I/O interface in the "Introduction."

As a general rule, a status byte is sent to the channel twice for *most* commands, at the beginning and the end. The rewind commands can be an exception to this rule. However, *all* commands do submit a status byte at the end of the initial selection period. The contents (bit configuration) of the status will vary depending upon the type of command.

Figure 21 shows:

1. When status bytes are sent to the channel per command.
2. The content of the *normal* status byte.
3. How long the control unit is locked to the channel per command under burst mode. This is indicated by the "operational in" line.

Note, on Figure 21, that the commands are categorized into five groups. The status byte content, at the end of initial selection, is forced active for groups 3, 4, and 5. The circuits to generate these status bits are shown on Figure 302, objective 6.

An important point, relative to status bytes, is that the channel must *eventually* accept (service out) any status byte activated by *status in* trigger.

Valid Control Commands

Any control command that requires TU operation (P trigger in the command register) is a valid control command, provided no unit check or busy condition

for the selected TU is detected during initial selection. The line "valid control command" *forces* a channel end status bit into the initial status byte (Figure 302, coordinates 4D and 5D).

These commands do not require channel operation during execution (Figure 21, groups 4 and 5). After initial selection is complete, channel disconnects (operational in line inactive). The TC and/or the TU will complete the operation without channel. After the operation is complete, TC must "interrupt" channel (via multiplex mode) with the end status byte, by raising "request in" to channel.

The following are valid control commands if no unit check or busy condition exists.

1. Erase gap (ERG).
2. Write tape mark (WTM).
3. Forward space record (FSR).
4. Forward space file (FSF).
5. Backspace record (BSR).
6. Backspace file (BSF).
7. Rewind TU (REW).
8. Rewind-unload TU (RUN).

Any one of these commands is begun when "status in and service or command out" sets the tape op sync trigger at the end of initial selection (Figure 302, objective 11). When either "out" tag falls, the tape op trigger is set to begin the operation by bringing up:

1. Write op (ERG, WTM).
2. Read op (FSR, FSF).
3. Backward op (BSR, BSF, REW, RUN).

Channel Accepts Status

If the response to "status in" is "service out," this indicates that the channel has accepted the status byte and TC should proceed with the execution of the command if status conditions allow. The two status conditions that can affect the execution of a command are unit check and busy. Both can affect each command in a different manner. Refer to "Unit Check" and "Busy Status" for complete details.

Once the status byte has been accepted by channel (service out), the following action occurs to reset the initial selection period and progress on to the "operate" time if allowable or applicable.

1. Reset "status in" trigger by first resetting the "first command" trigger (Figure 302, coordinates 2-3A).

2. Reset initial selection trigger and status bit latches, via objective 9a on Figure 302.

3. For commands which require tape unit motion, set the "tape op sync" trigger as described in objective 9b.

4. Set the control immediate trigger if this is a valid control command (objective 9c).

Control Immediate Trigger

The prime purpose of the control immediate trigger is to function as a remembering device for TC when handling a valid control command. Since all valid control commands submit a forced channel end bit to the channel at the end of initial selection, TC must insure that this bit is accepted. This trigger can only be set by a "service out" response. Thus, if this status is rejected, the control immediate trigger will not be set, thereby informing TC to resubmit this channel end bit at a later time. These effects are described in the following section.

Channel Rejects Status

When channel responds to status in with command out, the channel has not accepted the status byte. This response will affect the tape control unit operation as follows:

1. Reset status in trigger by first resetting the first command trigger (Figure 302, coordinates 2-3A).

2. Terminate the initial selection period by resetting the initial selection trigger (Figure 301, coordinates 5A) after command out and first command are inactive.

3. *Will not* reset status bit latches so as to preserve the status bits. "Status bit reset" cannot be activated (Figure 302, coordinates AE).

4. Possibly inhibit the execution of particular commands.

5. Resubmit the rejected status byte either by stack trigger active or control immediate trigger inactive. (A submitted status byte must eventually be accepted.)

The following discussions are relative to the various conditions associated with 4 and 5, above.

Stack Trigger

The stack trigger serves as a dual "remembering" device for the TC. It can only be set by a rejection of status (command out) for all commands except a valid control command (objective 10 on Figure 302). In its first role, the stack trigger will originate a "request in" to channel for the re-submission of the previously rejected status byte if "suppress out" from channel is not active. This type of entry is definitely under the multiplex mode and is described under "Control Unit Request In Sequence" in this manual.

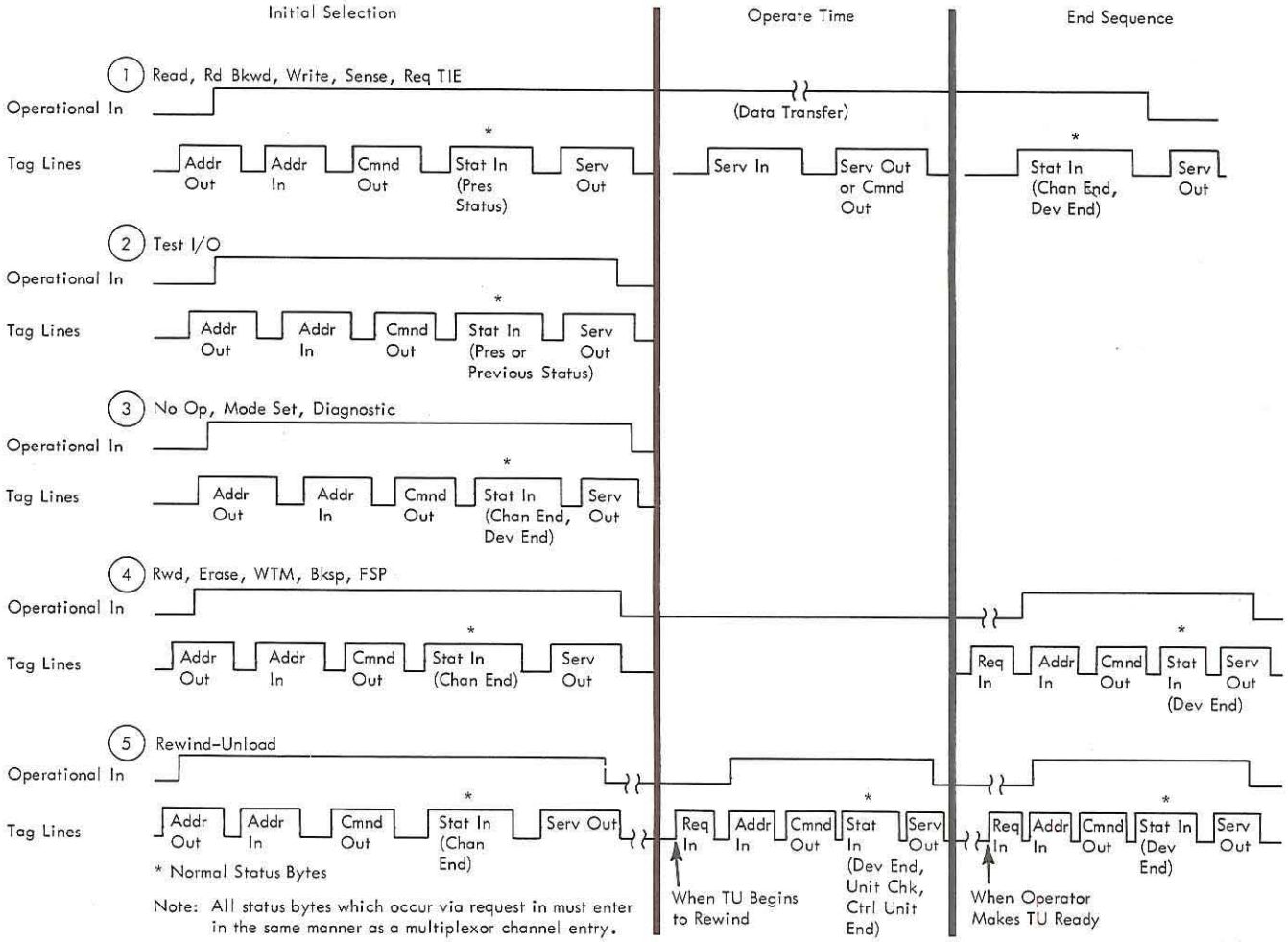
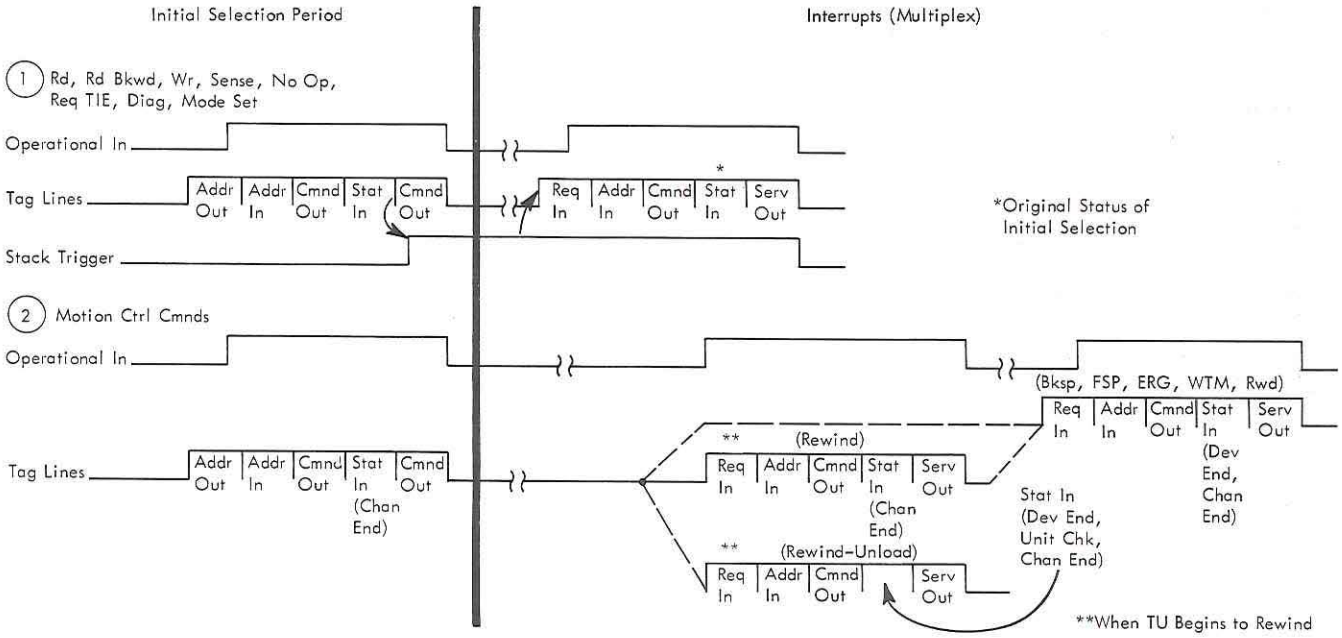


Figure 21. Channel Interface Lines



● Figure 22. Status In/Command Out

Figure 22 illustrates how the "stack" trigger causes the re-submission of the previously rejected status bytes for all those commands in Group 1. The contents of the status byte should be the same as the original status byte. Figure 21 indicates the original contents of each status byte per command type.

In its second "remembering" role, the stack trigger can cause a "busy" condition as described later under the "busy" sections. Since the active state of the stack trigger indicates the preservation of rejected status, the tape control unit cannot allow the execution of another command if the status latches are holding status. As shown in Figure 23 the tape control unit possesses one latch each for bits 2, 4, 6, and 7. Any command which would normally use one of these latches (groups 1, 2, and 3 on Figure 21) would set stack if this status was rejected. Stack would prevent the execution of another command since one or several of these status latches are retaining information. On the other hand, a valid control command normally utilizes a forced channel end bit (latch not set) and later only a device end bit. Figure 23 indicates that there are eight device end latches (one per TU). Thus stack is not set on valid control commands. This allows execution of another command if the new command uses one of the seven remaining tape units whose device end latch is not set. If a unit check occurs (unit check latch active) on a motion control command, it is no longer considered "valid." A "status in" "command out" response (rejection of status) will allow stack trigger to be set (Objective 10a on Figure 302). TC is again busy to the receipt of any new command because one of the "one-of-a-kind" status latches is set.

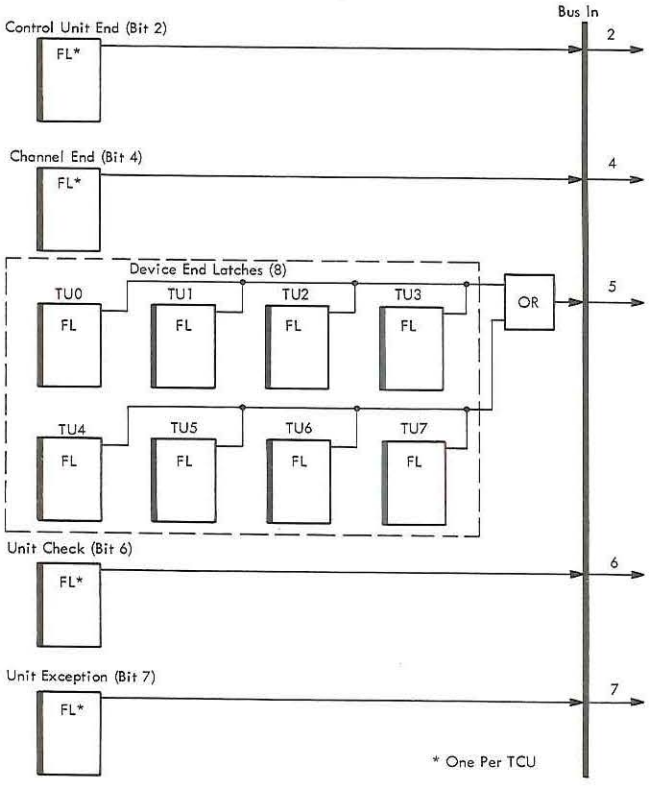


Figure 23. Status Bit Latches

Control Immediate Trigger

If the *forced* channel end bit in the valid control command status byte is rejected, the control immediate trigger cannot be set (objective 9c, Figure 302). The inactive state of the control immediate trigger will produce the results shown on Figure 22, Group 2. This *forced* channel end bit must eventually be accepted by the channel and, dependent upon the type of motion control command, it is re-submitted at various times. Note on Figure 22, Group 2, that all entries are in multiplex mode. Therefore, if "suppress out" from channel is *inactive*:

1. On the rewind-unload operation, channel end bit is substituted for the *normal* control unit end bit and re-submitted at the normal second status byte interruption. This occurs when TU begins to rewind. TC will continue to interrupt (re-submit) until channel accepts (service out).

2. On the rewind operation, TC will begin to interrupt with this channel end status byte as soon as the tape unit begins to rewind. If channel continues to reject (command out), TC will continue to interrupt until channel accepts or the tape unit completes its rewinding. On the latter, the channel end bit will be included in the normal end status accompanying the device end bit.

3. For the remaining motion control commands (WTM, ERG, BSP, FSP), the channel end bit replaces the normal control unit end bit in the end status byte along with the normal device end bit.

If the control immediate trigger is off after initial selection for both rewind and rewind unload, the channel end trigger is allowed to be set by "end pulse" as the tape unit begins rewinding. The on state of the channel end trigger causes the TC to "request in" with channel end status.

Command Execution

When a status byte is rejected (command out) by the channel at the end of initial selection, the command may or may not be executed depending on conditions. The following chart summarizes the effects of "command out" for each type of command:

COMMAND	STATUS BYTE	IS STACK TRIGGER SET	COMMAND EXECUTED	COMMENTS
Request TIE, Read, Read Backward, Write, Sense	Present conditions	Yes	No	Set stack (Figure 302, coordinates 5G) causes end reset to reset command register (Figure 301, coordinates 5D)
No-op, Diagnostic Mode Set	Channel end, device end	Yes	Yes	End reset is again generated, but command is normally executed during initial selection period.
Motion Control Commands	Forced channel end bit	No	Yes/No	Yes—if "valid ctrl cmd" (No unit check or busy) No—if busy or unit check; either causes reset to command register (Figure 301, coordinates 5D)

Control Unit Busy

- **TCU can perform one command at a time; therefore, if the command register contains a command, the TCU is busy.**
- **If the end status byte from a previous operation has been stacked — not yet accepted — TCU is retaining this status and cannot perform any command until cleared of this status.**
- **If an operation has just completed and is preparing to send its end status byte, the TCU is busy to any other command from channel.**
- **Control unit busy is recognized when the channel initiates another initial selection and any of the above conditions exist.**
- **Control unit busy is indicated by a "short sequence" — address out, select out, status in.**

The term "busy" implies that the selected control unit is unavailable to perform another command if it is:

1. presently occupied performing a command, or
2. presently "preserving" (stacked) a previously rejected status byte.

Item 1 is self-explanatory, since it is logical that the control unit cannot perform another command if it is already executing a previously issued command. However, the reasoning involved in item 2 is the same as discussed in the "stack" section, that is, the philosophy that most of the status bit latches are singular in this unit and can only preserve status for one operation at a time. Thus, if some status byte was previously rejected (command out), this control unit is busy to the receipt of another command until the previous status is accepted.

During the initial selection period, there are two distinct times that a "busy" condition can be experienced. The first busy is sampled at "address out/select out" time at the beginning of initial selection. This condition is labeled "control unit busy" or "short sequence." The second busy condition is sampled at the end of initial selection and is labeled "busy."

Consider the control unit busy (CUB), or short sequence. On Figure 300, coordinates 3C, there is a latch labeled "performing command." If this latch is active, it will inhibit the activation of the line "adapter address decoded." This line is necessary to proceed any further into the initial selection sequence. Thus, with "performing command" latch active, initial selection will terminate at "address out/select out" time and the AND circuit at coordinates 4C will generate the line "control unit busy." The functions of this line are described in the objective on Figure 300 labeled "CU busy sequence." They state that at select out time, if performing command latch is active, the control unit will:

1. set control unit end trigger which will cause a control unit end (bit 2) to be submitted in the next status byte to signal channel that TC is no longer busy,
2. force the status in tag line to the channel and force a "P" bit (parity), "1" bit (status modifier), and "3" bit (busy) in this prefabricated status byte. This unique status byte is used as a flag to channel that a CUB exists and it is the "short sequence" type. Since the status in trigger was not set, a channel response (service out or command out) to this type of status byte is not necessary.

The "performing command" latch is the heart of the "short sequence" busy conditions. Analysis of this latch on Figure 300, coordinates 3C illustrates that it can be set if any one of three conditions exist:

1. If the command register contains a command (P, 5, 6, or 7 bit) or TC is executing a request TIE command (no-op req latch on). Naturally if TC is presently performing a command, it cannot accept another.
2. End pulse active implies that TC is presently terminating a command and the end status has not yet been submitted.
3. Stack trigger is set and the tape unit addresses do not compare. This condition implies that the rejected status byte (stacked) was from a tape unit other than the tape unit the channel now wishes to select. The address of the former TU, in the TU select triggers, is compared to bus out positions 5, 6, and 7 (new desired TU). This condition must inhibit any further progression into initial selection; otherwise the new TU address will destroy the former TU address in the select triggers. The former address is necessary

to identify the rightful owner of the stacked status byte when it is sent to channel.

The CUB (short sequence) is relatively short in duration as illustrated by the timing chart on Figure 7D. This is necessary to inhibit any disturbance to the control unit if it is either executing or terminating a command, or preserving a status byte for a tape unit other than the tape unit that the channel is presently seeking. If a status byte were stacked from an operation performed by the same TU as the channel is now seeking, CUB *would not* be generated and initial selection would progress. The second type of "busy" would now be generated under these conditions.

Busy

- **This busy condition, unlike the short sequence, involves a *complete initial selection period.***
- **Busy bit occupies bit 3 of the status byte.**
- **Busy bit *without* a status modifier bit (bit 1) indicates either that the tape unit or the TCU is busy.**
- **Busy status without modifier bit, caused by:**
  1. **Tape unit is busy rewinding.**
  2. **TCU has outstanding status from an operation performed by the tape unit *presently* being selected.**
- **Busy, due to rewinding, will be indicated for any command type issued except test I/O.**
- **Tape unit rewinding is indicated by TU status B active.**

The second type of "busy" is interrogated at the end of the initial selection sequence. This type of "busy" can occur only if:

1. any command is issued and the selected TU is rewinding or switched to another TC (2816 tape switching feature) or,

2. any command, other than a test I/O, is issued to the same tape unit address for which an outstanding status byte exists.

To further amplify this point, assume the condition where TU 5 completed an operation and its end status byte was rejected (command out). If channel attempted to issue another command, other than test I/O, to TU 5, CUB (short sequence) would not occur because the tape unit addresses would compare. However, the second type of busy would occur since both the channel and TC were involved with the same tape unit. As a result, a busy bit (bit 3) would be submitted along with the outstanding status byte at the end of initial selection. The new command would not be executed and the control unit would now be completely freed if channel accepted this status byte. On the other hand, if this were a test I/O command that the channel issued to the same TU, a busy bit (3) would not be included in the status bit and the original contents of the status byte would be sent to channel. This is logical, since the prime purpose of the test I/O command is to seek status. Therefore, the outstanding status does not represent a busy condition to the operation of a test I/O command.

On Figure 302, coordinates 2D, are the circuits involved with the detection of the "busy" condition. The conditions which constitute a previously rejected status byte are AND'ed with "no test I/O" command to generate busy. End status conditions are:

1. selected device end, the selected TU has completed a control command operation and "device end" status is waiting,

2. channel end trigger alone, if selected tape drops "rdy" when end status is about to be sent via multiplex entry, but channel begins initial selection on same TU before the status can be transmitted,

3. stack interrupt trigger – status for a previous operation was refused by channel. *In most cases* busy status is caused by a combination of these conditions. When a busy status is generated, the status byte at the end of initial selection will contain a bit in position 3 and *no status modifier* (bit 1) included with the original status bits.

One point should be stressed concerning busy status. If end status does exist during initial selection, the designated command will not be executed as indicated by the command register reset by "busy" in Figure 301, coordinates 5E. As initial selection is completed and channel accepts the initial status byte, the end status conditions that brought up busy in the first place are cleared. There is good reason for this. If end status is waiting, then actually the previous operation is incomplete because channel does not yet have the end status. By "wasting" an initial selection sequence, the preceding operation is completed.

To summarize the differences between the two types of "busy":

1. Control unit busy (short sequence) checks for TC in process of performing or ending a command or, outstanding status from a tape unit *other* than the one now being addressed.

2. Busy (end of initial selection) checks for the *selected* tape unit rewinding or, outstanding status from the *same* tape unit that channel is now attempting to select.

Unit Check

- **During initial selection, a unit check can be generated by:**
  1. **Tape unit not ready.**
  2. **Command reject.**
  3. **Command parity error.**

- **Unit check occupies bit 6 of the status byte.**
- **A unit check generated via command parity error during initial selection will reset the command register and abort the command, with the exception of a test I/O command.**

Unit check is the 6 position of a status byte. Unit check indicates that some unusual condition has occurred which requires attention. In most cases a sense command is performed by the channel after a unit check status byte. Bits in the sense bytes provide more information about the cause of the unit check.

Unit check indicates error conditions which might prevent execution of the command by resetting the command register during initial selection. The circuit of unit check is shown on Figure 302, coordinates 2-3C. Three possible causes during initial selection are:

1. Command parity error, gated by "set command register" if the command byte has even parity (Figure 301, objective 8b).

2. Command reject, set if a write, write tape mark, or erase gap operation is designated and the selected TU is file protected, or a mode set "data converter on" indication when the TC does not have the data convert feature installed. The checks are indicated by three AND circuits (5G, 5F, and 4H) on Figure 302, coordinates 2B.

3. Not TU status A. The selected TU is not mechanically ready (R101 not picked). Notice that the interrogation of a "not ready" tape unit is not made on a sense command. This is because a sense command can be issued to either a not ready, non-existent, or ready TU without inhibiting the execution of this command.

Figure 24 is a chart showing what effect unit check, during initial selection, has on each type of command depending upon how it is generated. Unit check does not always prevent execution of the command because the reset to the command register via unit check cannot affect commands that require only initial selection to be executed. Figure 24 shows which types of commands can proceed and which types are aborted because of unit check conditions. Figure 24 also shows the status bits that can be expected from each type of command with and without a unit check condition.

TU Status A and B (Figure 25)

A *selected* tape unit indicates its status to the tape control unit by submitting two lines: "model (N)" and "not ready" line. These two lines are interrogated by the control unit and then transposed into two different line names: "TU status A" and "TU status B."

TU Status A: The tape unit is selected and ready. This line is activated by the "model (N)" line from the TU when the TU start relay 101 is picked (mechanically ready) and the TU is selected by TC.

TU Status B: TU is not ready (R101 is dropped) or rewinding or switched. The TU status B line is active when the "not ready" from TU is active and TC is selecting that tape unit as illustrated on Figure 25. From the tape unit logic the "not ready" line is activated if:

1. the start relay 101 is dropped or,
2. the rewinding latch in TU is set or
3. tape switching (2816 feature) indicates that the tape unit is switched to another TC.

The chart on Figure 25 summarizes the status of the selected tape unit under the various conditions of TU status A and B.

Turnaround

- **If the selected TU is in the opposite directional status to the new command being issued, a turnaround sequence is initiated.**
- **"Backward memory status" will always be active when the selected TU is in a backward status.**
- **The delay counter will run in millisecond mode as the turnaround sequence is executed.**
- **At a delay count of 96, the backward trigger in the selected tape unit will be either set or reset to place it in the proper directional status.**
- **An additional delay time is added to the delay count if the selected tape unit is a model 1.**
- **If the selected TU is in a "write-forward" status and must be reverted to a "read-backward" status, the TU will be allowed to go forward first to turn off its write triggers. This will allow any "noise" splash to be placed further away from the gap.**
- **All turnaround sequences are performed prior to initiating the command and achieving tape operating speed.**

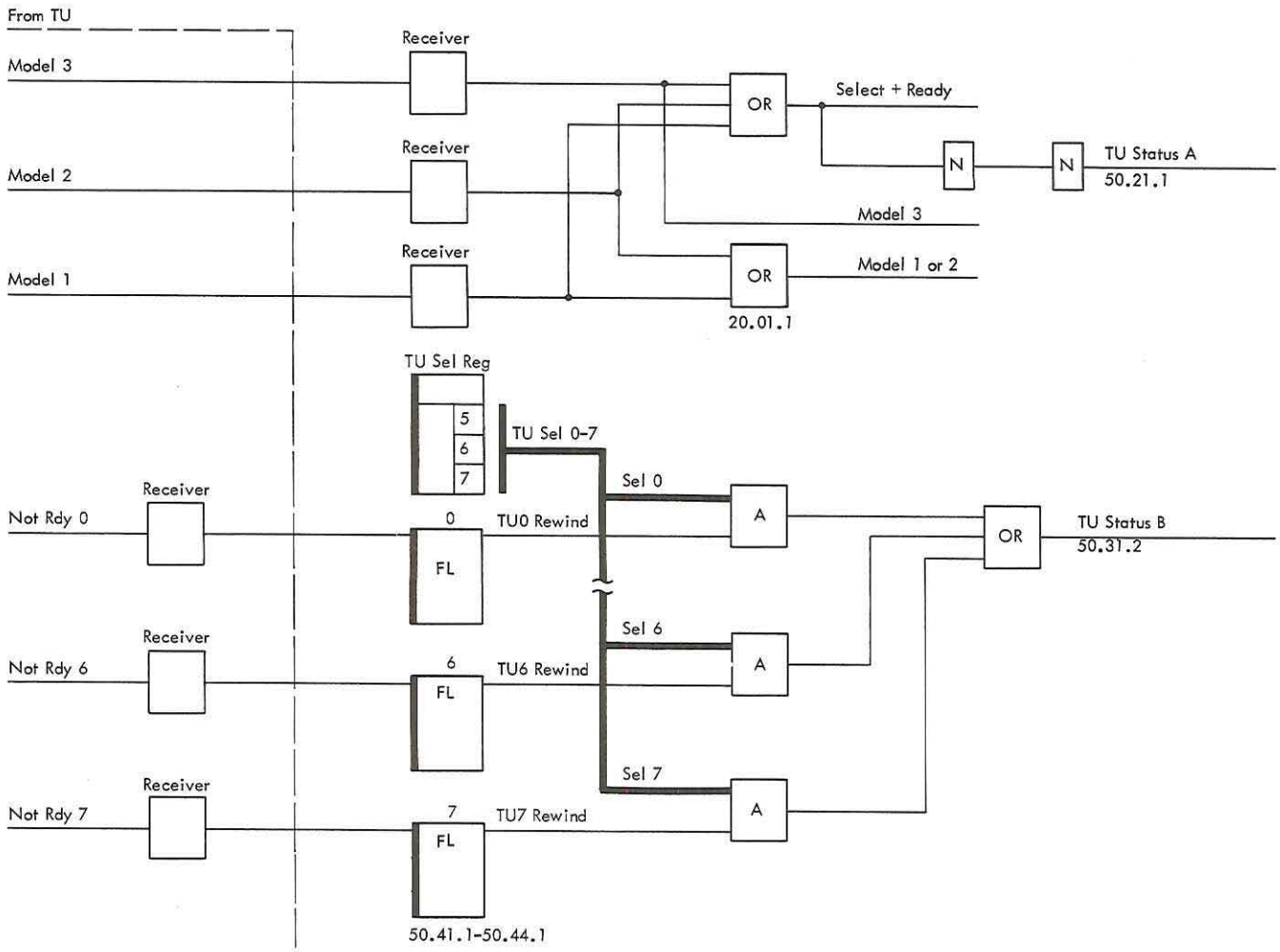
Turnaround means the tape unit (TU) just completed an operation moving tape in one direction and the operation just beginning designates that tape be moved in the opposite direction. Because the TU is partly mechanical, it requires some time to reverse its mechanical functions, such as the reel drives. The amount of turnaround delay varies with the TU model. The amount of delay is timed by the delay counter which advances at a count frequency determined by the "TU (N) model" line, Systems 20.01.1, and millisecond control.

After initial selection is complete, the designated operation is begun by activating the operation line

Command	Normal Status Byte	TU Not Ready		Command Parity		Command Reject	
		Command Executed	Status Byte	Command Executed	Status Byte	Command Executed	Status Byte
Test I/O	P 0 1* 2 3 4 5 6 7	Yes	P 0 1* 2 3 4 5 6-1 7	Yes	P 0 1* 2 3 4 5 6-1 7	N/A	N/A
Sense	P-1 0 1 2 3 4 5 6 7	Yes	P-1 0 1 2 3 4 5 6 7	No	P 0 1 2 3 4 5 6-1 7	N/A	N/A
Diagnostic No Op	P-1 0 1 2 3 4-1 5-1 6 7	Yes	P 0 1 2 3 4 5 6-1 7	Yes	P 0 1 2 3 4 5 6-1 7	N/A	N/A
Control Command (Requires TU)	P 0 1 2 3 4-1 5 6 7	No	P 0 1 2 3 4 5 6-1 7	No	P 0 1 2 3 4 5 6-1 7	Only Effect WTM and Erase act the same as command parity error.	
Read Read Bkwd Write Request TIE	P-1 0 1 2 3 4 5 6 7	No	P 0 1 2 3 4 5 6-1 7	No	P 0 1 2 3 4 5 6-1 7	Only Effect Write command acts the same as command parity error.	
Mode Set	P-1 0 1 2 3 4-1 5-1 6 7	Yes	P 0 1 2 3 4 5 6-1 7	No	P 0 1 2 3 4 5 6-1 7	Only Effect Set mode to converter ON. Command is executed with unit check.	

* Current status plus indicated bits.

Figure 24. Initial Selection Unit Check Conditions



TU Status A	TU Status B	Tape Unit Status	Response to Initial Selection
Not Active (0)	Not Active (0)	Non-Existent Tape Unit	Unit Check
Not Active (0)	Active (1)	Not Ready	Unit Check
Active (1)	Not Active (0)	Ready and Not Rewinding	Clear Status
Active (1)	Active (1)	Ready and Rewinding or Switched	Busy Status

Figure 25. TU Status A and B

such as "backward op" or "write op" (Figures 311 and 510.) Any operation that begins with the tape operation trigger on requires conditioning of a beginning delay such as "read delay" or "write delay." Before any tape operation can begin, the designated delay circuit must be conditioned by "not turnaround."

Backward Memory Status

The key line in the tape control (TC) that determines whether a turnaround delay is necessary is the "backward memory status" line brought up by "backward status" from TU. If the TU backward trigger is on, indicating the preceding operation moved tape backward, "backward memory status" is active in TC.

Any command that brings up "read op" or "write op," designating forward tape movement, will set the turnaround trigger (Figure 311, coordinates 3A) if "backward memory status" is active. Any command that brings up "backward op," designating backward tape movement, will set the turnaround trigger if "backward memory status" is not active.

"Turnaround" degates normal delay circuits and gates millisecond mode to the delay counter which advances to begin the delay sequence.

Advance and Erase Tape Forward

If the tape unit is in "write status" (not "read status" from TU) and "backward op" is designated, the go trigger is turned on at delay counter 3. Write status indicates the last operation wrote a record, tape mark (TM), or erased tape. In the case of the data record or TM, the erased area after the last check character must

be free of any recorded information. To be certain this area is "clean," the tape is advanced in write status (erasing) so that the write head is farther away from the last good record.

When the TU switches from write to read status, a "noise" splash is recorded on tape under the write heads. By moving the tape further ahead for a few milliseconds, the weak noise splash ends up farther away from the last good record; this prevents a possible read error later on.

Tape Movement

With the go trigger on at "3" time, "go" is sent to the TU and the prolays begin moving tape with the erase head energized. After the necessary tape movement, the delay counter outputs reset the go trigger at:

1. "D 30" for Model 1.
2. "D 32" for Model 2.
3. "D 53" for Model 3.

The selected tape unit is now prepared to be turned around.

Turnaround Tape Unit

When the delay counter advances to 96, the line "D 96 turnaround" sets the TU into the status designated by the "op" line. These "op" lines are conditioned by three different groups of commands. Note 1 on Figure 510 shows all the commands and the particular "op" line each conditions. The following occur:

1. "Read op" gates "set read status" to TU, to reset the R/W trigger and bring up "read status." "Set read" also resets the TU backward trigger to drop "backward memory status" to TC.
2. "Write op" gates "set write status" to TU to set the R/W trigger and bring up "write status" ("not read status"). "Set write" also resets the TU backward trigger to drop "backward memory status" to TC.
3. "Backward op," gated by "D 96 turnaround," sets the backward trigger in TC; The on state of this trigger gates "backward" to TU, to set the TU backward trigger, which:
   a. resets the R/W trigger to bring up "read status,"
   b. sends "backward memory status" back to TC.

Because the Model 1 TU is mechanically slower, additional turnaround time is needed. "D 96 turnaround" gated by "Model 1 TU" fires the Model 1 reverse delay singleshot, which deconditions the delay counter drive for 200 ms (Note 3 on Figure 311).

End Turnaround

After "D 96" or "model 1 reverse delay," the delay counter advances to "D 160 TA" to complete the turnaround delay sequence by generating a 925-nanosecond delay counter reset, which:

1. resets the delay counter,
2. resets the turnaround trigger to drop the delay counter drive (ms mode).

"Not turnaround" now conditions the delay circuit designated by the command and the new operation proceeds.

Read Forward Operation

- **Read operation moves data from the tape unit through the tape control to the data channel.**
- **If the previous operation left the tape unit in backward status, do a turnaround sequence.**
- **Start read delay and turn on go trigger to start tape motion.**
- **At the end of read delay, turn on read condition to look for data from tape.**
- **Read and check the data:**
  1. **Final amplifiers**
  2. **Skew register (high or low)**
  3. **Read/write register**
  4. **Data register**
- **Generate a service in to transmit each byte to the channel.**
- **When the end of data is reached, read the check characters.**
- **Stop tape motion and reset the control unit.**
- **Start an end status sequence.**

Initiate Operation

During initial selection (refer to "initial selection" Figures 300, 301, 302, and 500) tape control receives the command byte over channel bus out lines and sets the command into the command register. Before the operation can begin the TC must:

1. Check for a busy condition (TC and/or TU).
2. Check for a unit check condition.
3. Receive a "service out" response to "status in."

Busy or Unit Check

A busy condition exists if:

1. TU is rewinding. or switched (to another TC).
2. TC has an end status condition.

Unit check error conditions possible are:

1. Command parity error, bus out parity even.
2. Not TU status A, TU not mechanically ready (not relay 101).

In either case, the initial status in byte is not "clean" – contains more than just the P bit. The command register is reset and the read operation cannot be executed.

Channel Response to "Status In"

Channel might refuse the initial status byte with a "command out" response to "status in." The tape control, in turn, "stacks" the status byte by turning on the stack interrupt trigger.

"Set stack" resets the command register and the operation is ended at the end of initial selection. Tape control then attempts to interrupt with the rejected status unless "suppress out" is active from channel.

Condition TC for Read

- **TCU interrogates the "backward memory" status from the TU. If on, TCU will first initiate a turnaround sequence.**
- **A "read delay" is initiated to allow the selected TU to accelerate to operating speed.**
- **Delay counter steps in millisecond mode during "read delay."**
- **After "RD," read condition is set to allow data to transfer into skew registers and allow stepping of the read clock.**

If the channel accepts the initial selection status byte with a service out, the tape op sync and tape op triggers are turned on in sequence. Tape op sync turns on at the rise of service out and tape op turns on at the fall of service out (Figure 331, coordinates 1A and 2A). Tape op gates the read op line to begin the read operation. The read op line, in turn, generates read delay at 3A if turnaround is not necessary. The flow chart of read operation is on Figure 501. Figures 331 and 501 will be useful in understanding read operation.

The first major action of a read operation is generation of read delay. Figure 26 is a simplified timing chart of a read operation. Read delay accomplishes two operations:

1. Sets the go trigger. The active go trigger output causes the selected tape unit to move tape forward.
2. Conditions millisecond delay counter drive circuits.

The delay counter advances in millisecond mode at a rate determined by the model of tape unit designated to read the record. While "read delay" is active, tape on the selected tape unit accelerates to proper operating speed. Because the read head on the tape unit was positioned either in an interrecord gap or in the gap between load point and the first tape record when the operation began, characters do not transfer to the tape control immediately after tape movement begins.

If tape on the selected tape unit was not at load point when the read delay started, the read condition trigger turns on when the delay counter advances (objective 7, Figure 331):

1. to 15, if a Model 3 tape unit is selected.
2. to 17, if a Model 1 or 2 tape unit is selected.

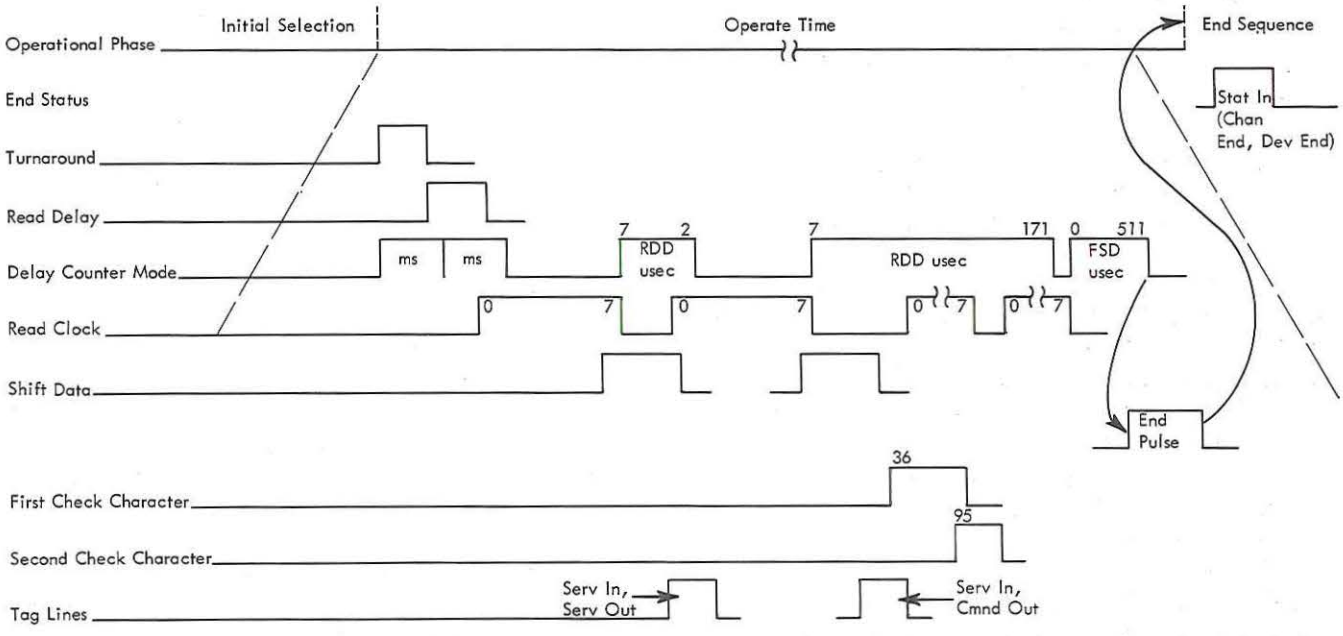


Figure 26. Simplified Read Operation

If tape on the selected tape unit is at load point when the read delay begins, the load point delay trigger is set, blocking the set path to the read condition trigger until the delay counter advances to 103 or 160 (Figure 311, coordinates 1B). Regardless of the original position of tape on the selected tape unit, read delays are shorter than corresponding write delays to ensure that read circuits are conditioned soon enough to read the first character in the tape record. At the end of read delay, the "read condition" trigger is set and produces a gating line to allow information to pass from the final amplifiers into the skew registers (Figure 311, coordinate 1D).

The read condition trigger blocks read delay, which resets the delay counter. Read condition also activates the first bit and read clock circuits to look for data from the tape unit. Between the times that tape control sets the go and read condition triggers, tape accelerates to proper operating speed.

Accept Characters

Each input character from the tape unit enters tape control final amplifiers. Final amplifiers contain a separate track for each bit position. A final amplifier track has two outputs, a high- and low-clip output. Input bits to corresponding final amplifier tracks must meet predetermined minimum amplitude requirements to produce high- and low-clip outputs. The high-clip output from each final amplifier track sets the corresponding high-clip skew register trigger; the low-clip output from each final amplifier track sets the corresponding low-clip skew register trigger. Theoretically, all bits in a character should arrive at the input to the final amplifiers simultaneously; however, all bits may not arrive at exactly the same time. The first high-clip output from any final amplifier track sets the first bit trigger to start the read clock.

It is possible to have a character in the low-clip skew register and nothing in the high-clip skew register because of low amplitude pulses from the tape unit. During seven-track read operation, the first bit trigger can be turned on without a high-clip pulse.

The high- and low-clip registers are compared. If they are not alike, the compare line is activated (not the C-Compare line). If the high-clip register has no bits and the compare line is active, the low-clip register must have at least one bit. On Systems TC 60.21.1, the AND circuit at 4G will turn on first bit and start the read clock when these conditions exist:

1. Read command is active.
2. RDD trigger is on.
3. The tape unit is seven-track.
4. The check character triggers are off.
5. Compare is active (high and low skew not alike).
6. The delay counter equals 9-12.

For any other conditions the signal from the tape unit must be large enough to set the high-clip skew register before the read clock can be started.

Read Clock Cycles

- **Read clock completes one cycle per character read.**
- **Between read clock cycles, the delay counter steps in RDD microsecond mode.**
- **RDD is checking for the large spacing between the last data character and the first check character.**

Spacing between bytes is checked by the read clock and delay counter. The output of the read clock at 7 time sets the read disconnect delay (RDD) trigger. "RDD" conditions microsecond delay counter drive circuits, and the delay counter steps in microsecond mode at the rate determined by the tape unit reading the record. If the next character from the tape unit is not a check character, the read clock output at RC-2 time resets the RDD trigger, stopping the delay counter (Figure 26). The next tape character must arrive at tape control soon enough to start the read clock before the delay counter advances to 36, or tape control will turn on the first check character trigger and process the character as a check character.

During seven-track operation, RC-7 controls the gate pulse that allows data to transfer from either the low- or high-clip register to the read/write register and LRCR. If the high-clip register has a character containing incorrect parity during a seven-track operation, read clock 7 generates a pulse called gate out low-clip. If the high-clip has no parity error, read clock 7 generates a pulse called gate out high-clip.

In nine-track operation the high-clip skew register is used for every character. The low-clip skew register can be gated out only during seven-track operation at RC-7 time. After RC-7 time, the read clock generates "RC reset" to:

1. Reset the first character trigger (only at the end of the first read clock cycle).
2. Reset the first bit trigger, blocking drive pulses to the read clock; the clock resets and cannot start again until the next character from the tape unit sets the first bit trigger.
3. Reset the high- and low-clip skew registers.
4. Set the R/W control trigger. The "on" status of the R/W control trigger indicates that the read/write register contains a byte of data and that action to unload the read/write register has not begun.

Read/Write Register to Data Register Transfer

The read/write control trigger remembers that the read/write register contains a byte which must be transferred to the data register. Output of the R/W

control trigger conditions a circuit to generate a shift data pulse (Figure 408). The primary functions of the shift pulse during read are:

1. Reset the R/W control trigger.
2. check the character in the read/write register for a parity error.
3. initiate a character transfer from the read/write register to the data register.
4. shift the error correction circuits to compute the cyclic redundancy check character.

Nine-Track Operation

Tape controls not equipped with the seven-track feature operate only in data converter (DC) off mode and cannot process six-bit characters. During each read clock cycle, a byte is set into read/write register positions P and 0 through 7. At the end of the clock cycle, "shift data" causes tape control to transfer bits in read/write register positions 0 through 7 to data register positions 5 through 12. Because the data register does not store parity bits, tape control does not gate the output of the read/write register P position. The transfer from the read/write register to the data register is unconditional and direct. Refer to Figure 306.

Seven-Track Operation

If the mode register specified both "DC off" and "translator off," tape control changes each six-bit character stored in the read/write register to an eight-bit character by adding two blank positions (data register 5 and 6) when transferring the character to data register positions 5 through 12. If the mode set specified translation, bits in the six-bit character in the read/write register feed through the BCD to eight-bit code translator, then set data register positions 5 through 12.

Because read/write register positions 0 and 1 are not set in seven-track operation, data register positions 5 and 6 always contain zeros when the translator is off. Only data register positions 7 through 12 may contain bits. Regardless of the status of the translator, the character gated from the data register to the channel in DC-off mode is unloaded from positions 5 through 12. Refer to Figure 308 for seven-track read data flow.

Cyclic Redundancy Check Register

In nine-track operation, each data character and the CRC character is shifted into the CRCR from the read/write register. Because of the shift pattern within the CRCR, a particular pattern is generated. The result should be the match pattern (111 010 111).

If the record has error characters in it, the erroneous pattern in the CRCR is used in conjunction with the error pattern register (EPR) to analyze and, if possible, correct the errors. Detailed information about the actual error correction operation is in the section entitled Error Correction.

Character Transfer from Data Register to Channel

In nine-track operation each shift data signal conditions tape demand to initiate a byte transfer from data register to channel. Because shift data is not activated when the first character tape mark trigger is set, and demand is not activated after channel indicates that it will not accept more characters from tape control, byte transfers to channel are not executed when one of these conditions exist. Under all other conditions, tape control unloads an output byte from data register positions 5 through 12 after each read clock cycle.

Because bytes transferred across the interface must be in odd parity, tape control checks the bit count in each byte unloaded from the data register and adds a P bit to the byte when necessary to make odd parity on the bus in lines.

"Tape demand" sets the "service in" trigger (Figure 408). The service in trigger allows tape control to place a data byte on the bus in lines and condition "service in" to channel 1 microsecond after "shift data" is conditioned. Because "shift data" also initiates a character transfer from the read/write register to the data register, the 1-microsecond delay allows tape control to complete the read/write register to data register transfer before signaling channel.

An overrun signal will occur if the control unit requests service and finds either the service in or service out line still active (Figure 408). Overrun condition will stop data flow.

Channel responds to the data byte and "service in" with either "service out" (indicating that it has accepted the character and will accept at least one more character from tape control) or "command out" (indicating that it has accepted the character but will not accept another data byte in the read operation). The "command out" reply sets the tape control stop trigger, holding demand inactive until the read operation is complete. Because the tape unit must read the complete record, tape control accepts all characters in the record from the tape unit and executes a read clock cycle for each input character regardless of the stop trigger status. If the stop trigger is set, however, no attempt is made to transmit characters across the interface to channel.

Check Character Cycles

- **First check character denotes the anticipated receipt of the CRC character in nine-track or the LRC character in seven track.**

- **Second check character denotes the anticipated receipt of the LRC character in nine-track.**

Normal character spacing in the tape record allows the tape unit to transfer at least one bit in a character to tape control to start the read clock and stop the delay counter before the counter advances to 36.

Because the check characters are spaced further from the last data character in the tape record than normal character spacing, the delay counter advances past 36 after tape control processes the last data character. RDD-36 sets the first check character trigger, indicating that the next character that the tape unit transfers is the first check character. The tape record might have blank check characters. If no bits are received between RDD-36 and RDD-95, the CRC character (nine-track only) is assumed to be blank. In seven-track operation the LRC character should be received between RDD-36 and RDD-95.

At RDD-95 the second check character trigger is turned on. In nine-track operation the second check character is the LRC character. In seven-track operation there is no second check character.

Tape control examines check character bits stored in the high-clip skew register for a parity error, but the state of the odd redundancy trigger does not unconditionally indicate whether the check character should contain an odd or even number of bits.

Correct parity for the LRCC and CRCC is described in the Functional Units section of this manual under the heading "Skew Register VRC"—"Skew Register Parity Conditions."

If a check character parity error is detected in the high-clip skew register during seven-track operation, the low-clip skew register will be gated to the LRCR and the read/write register. For other conditions the high-clip skew register is gated out.

During second check character time, RDD-95 to RDD-158, the skew register data cannot enter the read/write register. The nine-track LRCC will be gated to the LRCR but not to the read/write register.

At RC reset time, tape control resets the first bit trigger and skew registers. The first bit trigger "off" resets the read clock. Because the LRC character is the last character the tape unit transfers to tape control, the read clock should not run again in the read operation.

Forward Stop Delay

- **Forward stop delay (FSD) circuits check for noise at end of the record.**
- **"TAU end reset" will initiate ending sequence to send "end status byte," with "channel end" and "device end," to the channel after FSD.**

At the end of a read or forward space operation the tape should be blank after the check characters. Forward stop delay detects character or noise bits where tape should be blank.

At RDD-171 the forward stop delay trigger turns on (Figure 331, coordinates 2E). Forward stop delay runs the delay counter in microsecond mode. Since the read clock circuits are still conditioned to operate, if bits are detected before RDD-511 the delay counter is reset and FSD noise is signalled. Because the RDD trigger and forward stop delay are still on, the delay counter starts counting again from zero. Tape motion will continue as long as bits are detected on the tape.

If no bits are received and the delay counter reaches FSD 511, a TAU end reset pulse will be generated to reset the go trigger and the read condition trigger. TAU end also causes an end pulse which turns on the channel end trigger and initiates an end status sequence.

Another example of FSD usage: if, for any reason, the tape unit stops in the middle of a long record, the FSD circuit will allow easier recovery. Tape cannot physically move at full speed immediately. When a restart is attempted from the middle of data, the slow speed makes normal character spacing appear to be check character spacing and FSD becomes operative again. Thus, the tape control unit will keep the tape moving to the next record gap and turn on the FSD noise trigger.

Read Backward Operation

- **Read backward senses data from the tape, in reverse order, and transmits the data to the channel.**
- **A "read backward" into or at load point will cause a "unit check" to generate an end pulse and end the operation.**
- **Backward trigger is set at beginning of the operation and reset when TCU has decided "data character time" (check characters completed).**
- **Delay counter is set at RC-7 and reset at RC-2 and operates in microsecond control during check characters; operates in millisecond control during data characters.**
- **Shift control circuits can generate two "shift data" pulses during one read clock cycle when the decision is made that this is data.**
- **On "read backward" operations, all characters will enter the skew, LRC, *and* read/write registers.**
- **Data characters are the *only characters* checked for VRC in read/write register.**

- **At the end of data, stop tape motion and reset the control unit.**
- **Start an end status sequence.**

Read backward provides the ability to read data from a tape in reverse sequence. If a tape record must be read more than once, the read backward command eliminates time-wasting backspace operations. An entire tape can be read in reverse to eliminate rewind delay.

The major difference between read and read backward operations is sensing of check characters before data is received. Read backward data handling is the same as read forward except for determining when check characters (if any) have been read and when data is being read. Differentiating between check characters and data is an important part of read backward.

Read backward begins at the end of a standard initial selection sequence. Service out response to the status byte gates on the tape op sync and the tape op triggers. The decoded read backward command AND's with tape op to generate a line called "backward op" (Figure 332, coordinates 2A). Backward op sets up tape control circuits to perform the backward read operation.

Backward op generates backward read delay to turn on the go trigger and start tape motion. Backward read delay may be held off until a turnaround sequence is completed. If the tape unit is at load point, a line called "backward at load point" causes a unit check condition. Backward at load point brings up a backward disconnect and TAU end, which generates an end pulse (Figure 332 coordinates 7E, 2D, 3D, and 4E). The go trigger is reset, channel end and device end are turned on, and unit check is set. These conditions are transmitted to the data channel in an end status byte.

The tape unit may be at load point during the initial selection status sequence; however, the "backward at load point" line cannot come up until after initial selection. For this reason, an end status byte is sent to the channel after initial selection.

If the load point trigger is not on, the tape control will proceed to execute the read backward operation. Figure 507 is a flow chart of read backward. Figures 507 and 332 will be helpful in understanding a read backward operation.

Backward op brings up backward read delay which turns on the go trigger and runs the delay counter in *millisecond* mode. Backward op and backward read delay turn on the backward trigger to condition the backward line to the tape unit. The backward line turns on backward in the tape unit and also resets the tape unit read/write trigger to read status.

At the end of read delay the read condition trigger (Figure 332, coordinates 2B, objective 8) is turned on by:

1. RD-5 for a Model 1 tape unit.
2. RD-8 for a Model 2 tape unit.
3. RD-12 for a Model 3 tape unit.

Read condition activates the first bit and read clock circuits to look for the check characters from the tape. Read condition also turns off read delay which resets the delay counter.

The data path from the final amplifiers to the skew registers is conditioned to accept bytes from the tape (Figure 332, coordinates 2D). The tape control waits until physical motion of tape detects information. The first information should be the check characters followed by data.

Read Check Characters

- **Nine-track formats will always contain an LRC character.**
- **If the CRC character is even, it may not contain any bits, thus a blank CRC.**
- **Seven-track formats may or may not contain an LRC character.**
- **Backward trigger "off" indicates data character time.**

There are four possible combinations of check characters and data that must be correctly identified by read backward circuits.

1. Standard nine-track format with two check characters, then data.
2. Standard nine-track format with one check character (CRCC blank), then data.
3. Seven-track format with no check character (LRCC blank), only data.
4. Seven-track format with one check character.

Timing charts for each of the four possible formats are in Figure 27. The objective is to determine when the check characters (if any) have been read and when data bytes are being read. The backward trigger is used as a signal to indicate when to start processing bytes as data. Backward is turned on by backward read delay at the beginning of the operation. When tape control circuits determine that the check characters have been read, the backward trigger is reset. A better name for the backward trigger might be "backward check character time trigger." The backward trigger is reset by one of the four conditions in Figure 27.

Byte Timing

The basic factor used by the tape control is the spacing between bytes on the tape. Physical spacing is measured by using the delay counter to check timing. Read clock

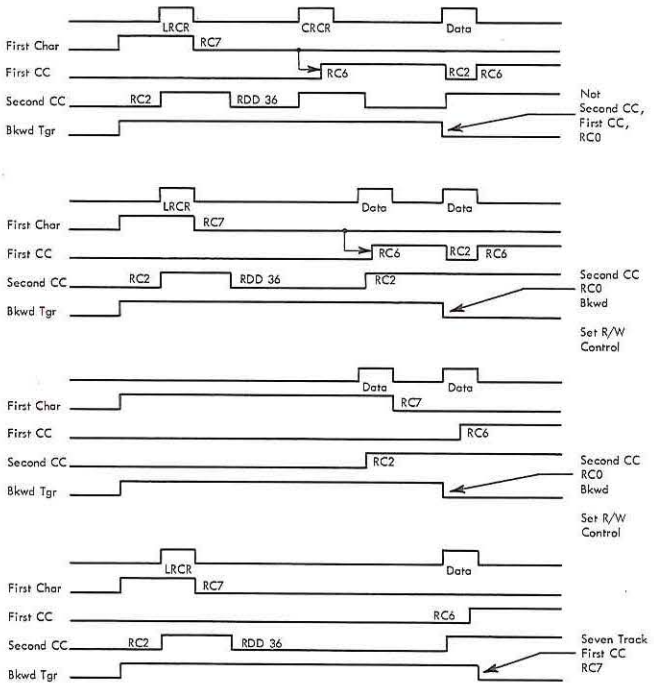


Figure 27. Read Backward Check Character Timing

pulses, associated with each byte, indicate when bytes are read. The delay counter measures timing between bytes.

Two triggers (first check character and second check character – Figure 332, coordinates 4-5D), are the key conditions to determine when check characters have been read. The second check character trigger is turned on by a read clock 2 pulse when a byte is sensed. It will remain on until the delay counter reaches RDD-36. If another byte is received before RDD-36, it indicates that the bytes were close enough together to be data spacing. After RDD-36 the spacing is considered long enough to be check character spacing. Time between bytes then can be checked by whether the second check character trigger is still on or has been reset by RDD-36.

The first check character trigger cannot be turned on by the first byte received. It is turned on by read clock 6 of each byte after the first, and reset by RC-2 of each following byte. In reading backward the second check character (LRCC) is read from the tape before the first check character (CRCC). The first check character trigger on indicates that at least two bytes have been read.

Check character format can be determined by analyzing the condition of these two check character triggers at RC-0 time, in relation to bytes received from the tape. The four formats are illustrated in Figure 27.

Nine-Track with Two Check Characters

Three conditions are needed to reset the backward trigger in a standard nine-track read backward operation:

1. Second check character trigger off.
2. First check character trigger on.
3. Read clock 0 pulse.

The first time that these three conditions are active at the same time occurs at read clock 0 of the first data byte (top of Figure 27). The second check character trigger is off at that time because the timing between the CRCC and the data byte is greater than RDD-36. The first check character trigger was turned on by read clock 6 of the second character (CRCC). At read clock 0 of the first data character the first check character trigger is still on because it is not reset until read clock 2. These three conditions activate an AND circuit to reset the backward trigger (Figure 332, coordinates 2B). The backward trigger off signals tape control circuits that the check characters have been read and the rest of the information is data. The *off* state of the backward trigger will allow the activation of "RC reset not write" (Figure 206) which will initiate shift control circuits at the end of each data byte read clock cycle.

Nine-Track with Blank CRCC

In nine-track operation it is possible to generate a blank CRC character. When this occurs, the block is written with only one check character (LRCC). Because of the CRCC it is impossible for a nine-track LRCC to be blank. The second section of Figure 27 represents this condition.

If the second check character trigger is on when a read clock 0 pulse occurs, the backward trigger is reset. Second check character trigger on indicates that the time between bytes was less than RDD-36 (data spacing). The tape control now knows that the CRC character was missing because data has been detected.

A blank CRCC presents a problem to the tape control circuits. This condition is not detected until RC-0, as the third character is being read. By that time, TC is retaining two data bytes, (one in the R/W register and the other in the skew register). In this situation the tape control must transmit two bytes to the channel in the same read clock cycle to catch up. The circuit that turns off the backward trigger, "set R/W control," also turns on the read/write control trigger to generate an extra shift data pulse (Figure 332, coordinates 3B and 4C). The first shift occurs at read clock 0 to shift the first data byte from the R/W register on to channel. The regular shift for the second data byte now occurs at read clock reset time of the same clock cycle because backward trigger was reset.

Seven-Track with Blank LRCC

In seven-track the LRCC may be completely blank, leaving only data characters in the block. Detection of this condition is very similar to detection of a missing CRCC in nine-track. The third section of Figure 27 shows the sequence used. Since there is no check character, the first two bytes are spaced close together. The second check character trigger is still on (less than RDD-36) when the second data character is read on the next read clock 0. Tape control now knows that both bytes are data. The two bytes must be sent to the channel in the same read clock cycle. In Figure 332, coordinates 3B and 4C, the same circuits are used that generated two shift pulses for nine-track with a blank CRCC. The read/write control trigger is turned on at read clock 0 and again at read clock reset of the same clock cycle.

Seven-Track with One Check Character

Standard seven-track format with one LRC character is illustrated in the fourth (bottom) section of Figure 27. The time between the LRCC and the data is longer than RDD-36 so the second check character trigger is reset. With the second check character trigger off, the backward trigger cannot be reset at read clock 0 of the data byte.

The tape unit has informed the control unit that this is a seven-track operation which has only one check character. After the check character has been read, the rest of the block must be data. The first check character trigger turns on at read clock 6 of the second cycle (first data byte). With seven track and first check character both active, the tape control knows that the check character must have been read. These two conditions, along with read clock 7, will reset the backward trigger (Figure 332, coordinates 2B and 3A.) There is no need to generate an extra shift pulse because only one data byte has been read.

Look for Tape Mark

- **Tape mark recognition also applies to read backward operation.**
- **Tape mark recognition must also check for more than two characters going backwards.**

A true tape mark, in either seven- or nine-track, consists of a one character record with an identical check character. When bits in either character are not a tape mark or if more than two characters are sensed, the record is not a true tape mark. If it is not a true tape mark record, the first character tape mark trigger is reset and the characters are assumed to be check characters or data.

When a true tape mark is detected, no data is transferred to the channel and the unit exception trigger is turned on at RDD-169. Unit exception indicates that a tape mark was read when the control unit was not expecting one. A complete description of tape mark detection during read backwards is in the "Functional Units" section.

Data Transfer Cycles

- **Data handling — shift control, etc. — is identical to read forward operation.**
- **RDD checks for end of data characters by setting at RC-7 and resetting at RC-2 between data characters.**
- **Delay counter will step in millisecond control during RDD of read backward.**

During read backward data cycles (backward latch reset) the RDD trigger gates millisecond control to the delay counter to time the spacing of characters. This is in anticipation of locating the last data character in the record. Figure 28 is a simplified timing chart on the read backward operation. During seven-track and nine-track operation, RC-7 controls the gate pulse which allows data to transfer from either the low- or high-clip register to the read/write register and LRCR. If the character in the high-clip register has an incorrect parity during a seven-track operation, read clock 7 generates a pulse called gate out low-clip. If the high-clip has no parity error, read clock 7 generates a pulse called gate out high-clip.

In nine-track operation the high-clip skew register is used for every character. The low-clip skew register can be gated out only during seven-track operation at RC-7 time.

After RC-7 time, the read clock generates "RC reset" to:

1. Reset the first character trigger (only at the end of the first read clock cycle).
2. Reset the high- and low-clip skew registers.
3. Reset the first bit trigger, blocking drive pulses to the read clock; the clock resets and cannot start again until the next character from the tape unit sets the first bit trigger.
4. Set the R/W control trigger. The "on" status of the R/W control trigger indicates that the read/write register contains a byte of data and that action to unload the read/write register has not begun.

Read/Write Register To Data Register Transfer

The read/write control trigger remembers that the read/write register contains a byte which must be transferred to the data register. Output of the R/W control trigger conditions a circuit to generate a shift data pulse. The primary functions of the shift pulse during read and read backward are:

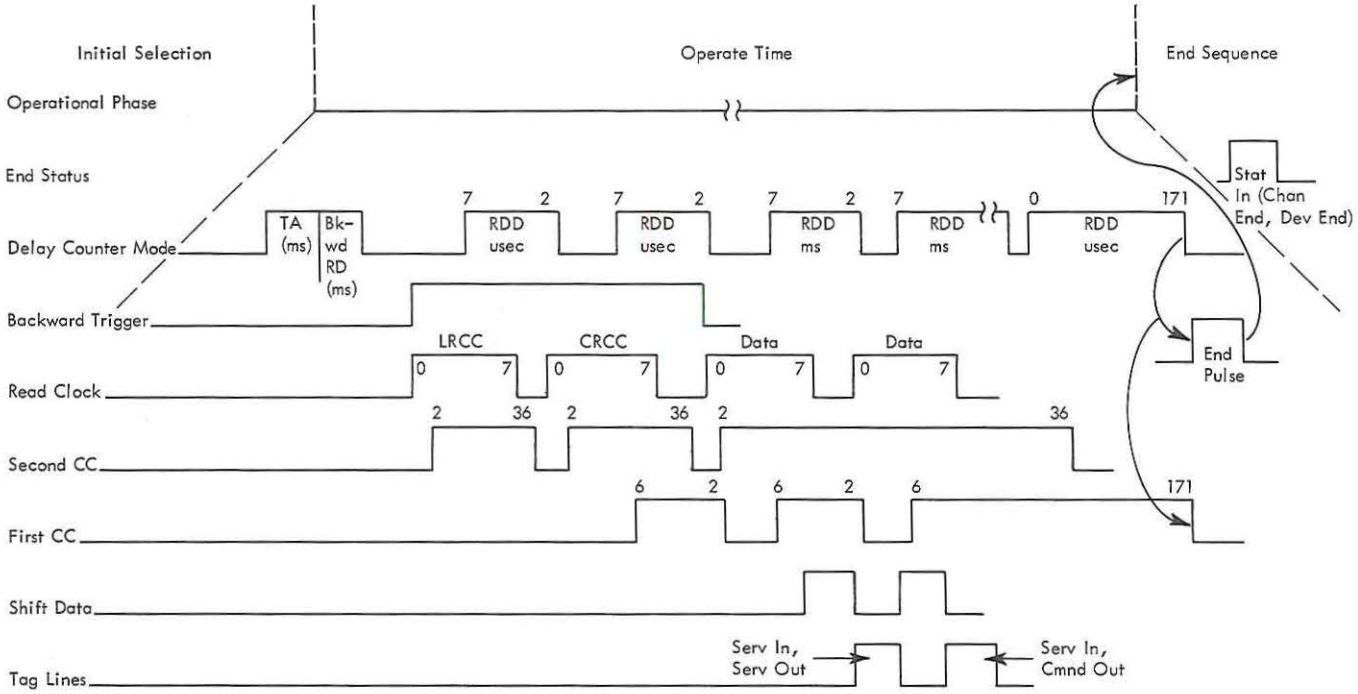


Figure 28. Simplified Read Backward Operation

1. reset the R/W control trigger,
2. check the character in the read/write register for a parity error,
3. initiate a character transfer from the read/write register to the data register,
4. shift the error correction circuits to compute the cyclic redundancy check,
5. generate tape demand to prepare "service in" circuits.

Nine-Track Operation

Tape controls not equipped with the seven-track feature operate only in data converter (DC) off mode and cannot process six-bit characters. During each read clock cycle, a byte sets read/write register positions P and 0 through 7. At the end of the clock cycle, "shift data" causes tape control to transfer bits in read/write register positions 0 through 7 to data register positions 5 through 12. Because the data register does not store parity bits, tape control does not gate the output of the read/write register P position. The transfer from the read/write register to the data register is unconditional and direct.

Seven-Track Operation

In DC-off mode, tape control changes each six-bit character stored in the read/write register to an eight-bit character by adding two zeros and transfers the eight-bit character to data register positions 5 through 12. If the mode set specified translation, bits in the six-bit character in the read/write register feed through the BCD to eight-bit code translator, then set data register positions 5 through 12. If the mode set did not activate the translator, bits in read/write register set data register positions 5 through 12 directly. Because read/write register positions 0 and 1 are not set in seven-track operation, when the translator is off, data register positions 5 and 6 always contain zeros; only data register positions 7 through 12 may contain bits. Regardless of the status of the translator, the character gated from the data register to the channel in DC-off mode is unloaded from positions 5 through 12.

Data convert on mode is described in the "Features" section of this manual.

Cyclic Redundancy Check Register

In nine-track operation, each data character is shifted broadside into the CRCR from the read/write register. Because of the shift pattern within the CRCR, a cyclic redundancy check pattern is developed. At the end of a read backward the CRCR should have a match pattern of 111 010 111, even though the CRC character is read before the data.

CRCR operation during a read backward is similar to read forward operation. The major differences are in the sequence of characters received from the tape, and inversion of the characters as they are shifted from the R/W register into the CRCR. R/W register P position goes to CRCR position 7; R/W 0 goes to CRCR 6, etc.

Inversion is required by the CRCR shift pattern when the data is received in reverse sequence.

If the record has error characters, the CRCR is used in conjunction with the EPR to analyze and if possible correct the errors. Detailed information on error correction operation is in the section entitled "Error Correction."

Character Transfer from Data Register to Channel

In DC-off mode operation, each "shift data" signal conditions "tape demand" to initiate a byte transfer from data register to channel. Because "shift data" is not activated when the first character tape mark trigger is set and because demand is not activated after the channel indicates that it will not accept more characters from tape control, byte transfers to channel are not executed when one of these conditions exist. In other cases, however, tape control unloads an output byte from data register positions 5 through 12 after each read clock cycle. In some special cases the tape control may be required to send two bytes to the channel in one read clock cycle.

When the first character tape mark trigger is on, no data can be sent to the channel from the data register. If the second character indicates that the record is not a true tape mark, the tape control may need to send both bytes of data to the channel in one read clock cycle. The first character tape mark trigger can be reset at RC-0. If the read/write control trigger indicates that data must be transferred, a shift pulse will be generated and a byte sent to the channel in the first half of the read clock cycle. At the end of the read clock cycle the read/write control trigger is set again and another shift pulse is generated.

Because bytes transferred across the interface must be in odd parity, tape control checks the bit count in each byte unloaded from the data register and adds a P bit to the byte when necessary to make odd parity on the bus in lines.

"Tape demand" sets the service in trigger. The service in trigger allows tape control to place a data byte on the bus in lines and condition "service in" to channel 1 microsecond after "shift data" is conditioned. Because "shift data" also initiates a character transfer from the read/write register to the data register, 1-microsecond delay allows tape control to complete the read/write register to data register transfer before signaling channel.

An overrun signal will occur if the control unit requests service and finds either the service in or service

out line still active. Overrun condition will stop data flow.

Channel responds to the data byte and "service in" with either "service out" (indicating that it has accepted the character and will accept at least one more character from tape control) or "command out" (indicating that it has accepted the character but will not accept another data byte in the read operation). The "command out" reply sets tape control stop trigger, holding demand inactive until the operation is complete. Because the tape unit must read the complete record, tape control accepts all characters in the record from the tape unit and executes a read clock cycle for each input character, regardless of the stop trigger status. If the stop trigger is set, however, tape control does not attempt to transmit characters across the interface to channel.

End Read Backward Operation

- **When the last data character is read, RDD will cause line "backspace reset read condition" to be activated at various time settings dependent upon model of TU.**
- **"Backspace reset read condition" line resets go, read condition, and delay counter millisecond control; RDD and delay counter microsecond control activated.**
- **During the ending sequence, the delay counter steps in microsecond control until "TAU end reset" at RDD-171.**
- **Normal "end status" byte contains channel end and device end bits.**

As each data character is read, the delay counter has been operating in millisecond control between read clock cycles – Figure 28. However, at RC-7 of the last data character, the delay counter again began to step in millisecond control but was not reset. It will be allowed to count higher to a designated point. When the delay counter advances to the point designated for the selected tape unit speed, the tape control generates "backspace reset read condition," indicating that the tape unit has reached the end of data (actually the beginning of the record):

1. 19 for a seven-track, Model 3 unit.
2. 21 for a seven-track, Model 2 unit.
3. 24 for a seven-track, Model 1 unit.
4. 7 for a nine-track, Model 3 unit.
5. 11 for a nine-track, Model 2 unit.
6. 14 for a nine-track, Model 1 unit.

"Go" and "read condition" are reset by backspace reset read condition line. Inactive state of "go" drops millisecond control to reset the delay counter (Figure 332, coordinates 4-5A). The delay counter is now conditioned to run in microsecond mode to prepare ending condition. At RDD-171, "TAU and reset" and "end pulse" are generated. The end pulse turns on *channel end* and initiates an end status sequence.

In a series of backward operations it is possible to stop tape motion in the gap between load point and the first block. When the next backward operation begins, the tape may move into the load point area without sensing data.

If load point is reached instead of a record, the tape unit automatically stops tape motion and sends an "at load point" signal to the tape control (Figure 332, coordinates 6-7C). "At load point" generates a backward disconnect and backward at load point (Figure 332, coordinates 7E, 2D, and 4E). Backward at load point turns on unit check and causes an end pulse to turn on channel end and initiate an end status sequence.

Write Operation

- **Write operation transmits data from the channel to the tape unit to be written on the tape.**
- **Request first data byte before starting tape motion. If channel responds with "command out" to first "service in," set word count zero and end operation. A "service out" response to the first "service in" means the first data byte is on the bus out lines.**
- **Condition tape control for write operation.**
- **If previous command was a backward operation, do a turnaround delay sequence.**
- **Start write delay and tape motion.**
- **Condition write and read circuits after write delay.**
- **Write and read check data record.**
  Send "service in" to request each data byte.
  A "service out" accompanies each byte sent by channel.
  Read circuits check byte skew and VRC as record is read.
- **Check for stop command to end write operation.**
  A "command out" to "service in" means stop writing.
- **Take disconnect delay and write check character(s).**
- **Stop tape motion and complete read check of record.**
- **Start end status in sequence.**

Initiate Operation

During initial selection, tape control (TC) receives the command byte over channel bus out lines and sets

the command into the command register. Before the operation can begin the TC must:

1. Check for a busy condition (TC and/or TU).
2. Check for a unit check condition.
3. Receive a "service out" response to "status in."
4. Get first data before starting tape.

Busy or Unit Check

A busy condition exists if:

1. TU is rewinding or switched (to another TC).
2. TC has an end status condition waiting.

Unit check error conditions possible are:

1. Command parity error, bus out parity even (command byte has bad parity).
2. Command reject, TU file protected.
3. Not TU status A, TU not mechanically ready (not relay 101).

In either case, the initial status in status byte is not "clean"—contains more than just the P bit. The command register is reset and the write operation cannot be executed.

Channel Response to "Status In"

There is the possibility that channel might refuse the initial status byte with a "command out" response to "status in." The TC, in turn, "stacks" the status byte by turning on the stack interrupt trigger.

"Set stack" resets the command register and the operation is ended at the end of initial selection. TC then attempts to interrupt with the rejected status unless "suppress out" is active from the channel.

First Data Byte

Figure 502 is a flow chart of write operation. Before writing can begin, the TU requires a mechanical (write) delay as it builds up speed. A higher priority TC can "tie up" channel before write delay is complete and channel could disconnect before sending the first byte. The TU would then move tape needlessly.

To ensure that at least a one character record is written, TC gets the first data byte *before* tape motion is begun. As soon as initial selection is complete, a special AND circuit (at coordinates 1A on Figure 335, objective 3) sets the service-in trigger.

For non-burst mode operation, the operational-in trigger is allowed to reset after initial selection is complete. TC has to begin a "request in" sequence for the first character.

Should channel respond with "command out," the write operation is terminated by setting the stop trigger and word count zero trigger with "stop command" (coordinates 1D on Figure 335) Word count zero brings up "no tape op stop" to generate "end pulse," unit check, and TAU resets, 4E. The word count zero condition is indicated in sense (operation) byte 6, bit 1. "End pulse" also sets the channel end trigger to begin the end status in sequence (objective 4, Figure 335).

A "service out" response to the first "service in" sets the tape op sync trigger, objective 5, indicating the first data byte is on the interface bus out lines and the write operation should continue. "Service out" gates only the eight data bits into the data register, 1C, the P bit is not gated. The P bit and eight data bits are parity checked. A bus out parity error is indicated if parity is even, though the operation continues.

Condition TC and TU to Begin Operation

Conditions for data flow, such as translate or convert were established previously by a mode set control command. Seven- or nine-track tape unit is distinguished by the "7 track" line from TU.

Write Delay and Tape Motion

Refer to the timing chart, Figure 29. After the fall of "service out" to the first "service in," tape op trigger is set to bring up the key line "write op" (Figure 335, coordinates 5A). "Write delay" is then brought up, objective 6, unless a turnaround sequence is required. The TU requires time to prepare for forward operation if the previous command moved tape backward (Refer to "Turnaround Operation"). "Write delay" sets the go trigger, 5B, to gate "go" to the TU. "Go" and "read select A internal" activate the TU prolays. "Write delay" also brings up "MS control" to gate "MS" mode, 4B, to the delay counter. The counter advances in millisecond (ms) mode at a frequency determined by the TU model number.

While "write delay" is active, tape on the selected tape unit accelerates to proper operating speed, but TC does not transfer characters to the TU.

If tape on the selected TU is not at load point when the write delay begins, the write condition trigger turns on, 3D, when the delay counter advances:

1. to 30 if a Model 1 tape unit is selected,
2. to 32 if a Model 2 tape unit is selected,
3. to 53 if a Model 3 tape unit is selected.

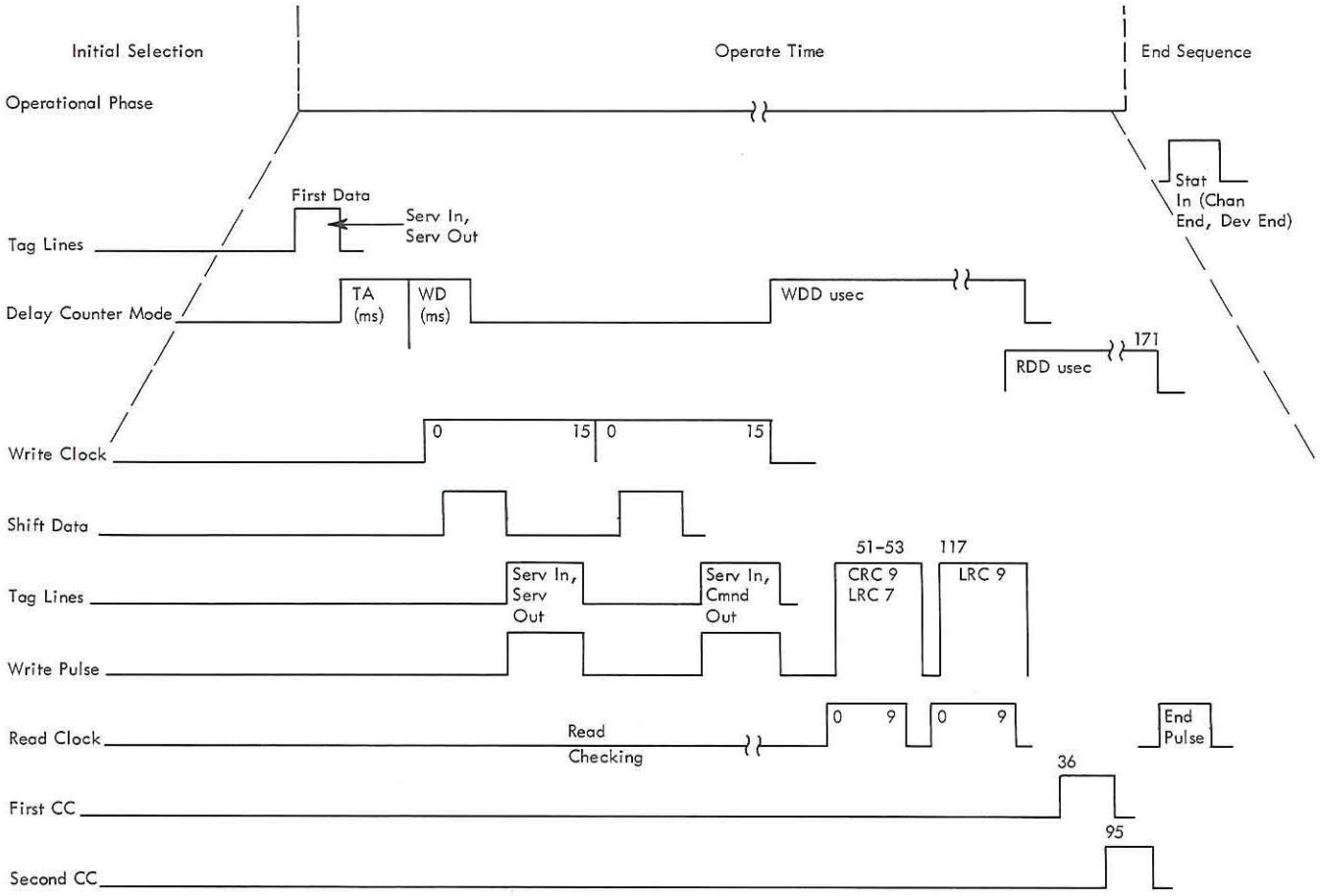


Figure 29. Simplified Write Operation

If tape on the selected tape unit is at load point when the write delay begins, the load point delay trigger is set, 2E, blocking the set path to the write condition trigger until the delay counter advances to 215 or 320. Because tape is not positioned at the same spot with respect to the write head after each load rewind operation, the longer write delay, when the load point delay trigger is set, causes the tape unit to erase a section of tape approximately 3¾ inches long between load point and the first character in the tape record. This erasure ensures that all previously recorded data on tape are destroyed before the new record is begun.

Write Delay Noise

If the first bit trigger turns on during write delay, the noise trigger is set. "First bit" indicates that flux changes have been detected on the tape. Tape should be blank during write delay; therefore, when the first bit trigger is on, the noise error trigger is turned on.

Write and Read Condition

Objective 10 of Figure 335 describes the following actions: The write condition trigger "on" drops "write delay" and stops the delay counter to permit actual writing of the tape record. However, between the times that the go trigger is set to start tape movement and the write condition trigger is set, tape on the selected tape unit reaches proper operating speed. In addition to ending the write delay, the "on" state of the write condition trigger further prepares the tape control to perform the write operation by:

1. Setting the write trigger release trigger, 4D, to allow write triggers in the selected tape unit to turn on and off as required to record characters on tape. The write trigger release trigger is reset after the tape control processes the last data character in the record to the tape unit. All tape unit write triggers on at that time switch to their off states. In returning to their reset conditions, write triggers record the LRC character at the end of the tape record. However, the resets to the off states are gated by an all one bit (special) character in the R/W register which uses the same path as normal data. This allows deskewing of the LRCC the same as for a data character.

2. Directly setting the read condition trigger, 2D, if the LP delay trigger is on. If LP is not on, read condition is set when the delay counter advances to 17 or 32 during the write delay. The on state of the read condition trigger conditions output gates for the final amplifiers to allow the tape control to check each character that the tape unit writes. Read condition also conditions read clock circuits to allow each character to cycle the clock.

3. Starting the write clock, 4-5D. Write clock outputs establish timings that control data flow and processing. The write clock cycles continuously as long as the write condition trigger is on.

Write the Tape Record

In accordance with objective 11, when the write clock (WC) advances to 2, it generates "WC-2" to reset the read/write (R/W) register, 3C, and set the R/W control trigger, 1B. The R/W control trigger indicates the status of the R/W register. When:

1. R/W control is on, the R/W register is empty during write,

2. R/W control is off, the R/W register is receiving or already contains a character (data byte) during write.

The "on" output of the R/W control trigger (and fall of "WC-2") brings up "start shift" to:

1. set the not stop data trigger,

2. bring up "tape demand" (depending upon data convert feature),

3. generate a 600 ns "shift data" pulse.

"Shift data" resets the R/W control trigger and gates the data register byte to the R/W register (Figure 408).

Data Register to Read/Write Register Transfer

Nine-Track Operation: Tape controls *not* equipped with the seven-track feature do not have a translator or data converter and cannot process six-bit characters. During each write clock cycle, tape control receives one eight-bit byte from channel and transmits the eight-bit byte to the selected tape unit. Each input channel byte is set into data register positions 5 through 12. "Shift data" initiates the direct transfer that causes the data register byte to set R/W register positions 0 through 7. Because the data byte in the data register does not have a check bit (not gated in), the R/W register P position is not set. (Refer to Figure 305.)

Seven-Track Operation: Tape controls equipped with the seven-track feature may have both a translator and data converter and, if so, can convert eight-bit bytes from channel to six-bit characters to be written on magnetic tape. The previous mode set command designated whether or not the data convert or translate feature is active. If "not data convert" (DC off) was specified, the mode set also indicated whether the translator is to be employed in the write operation. (Refer to Figure 307.) Data convert and translate cannot be used at the same time.

Request Character From Channel

Tape control receives a byte from channel and transmits a character to the tape unit during each write clock cycle in nine-track operation or in seven-track DC-off operation; each time that bits are unloaded from the data register, "shift data" brings up "tape demand" to set the service in trigger (Figure 335 coordinates 2A). "Service in" to the channel requests that another byte be transferred across the interface. With seven-track DC on, byte counter trigger states determine when another byte from channel is required. However, the sequence to condition the "service in" request to channel is the same whether the DC is on or off, or seven- or nine-track operation.

Not Burst Mode Operation

A multiplex channel not forced to operate in burst mode will have disconnected from TC as soon as the first write character was received. This occurred because channel dropped "select out" and allowed TC to reset the operational in trigger.

When each additional data byte is required, TC must execute a data interrupt. TC sends "request in" to channel when the service in trigger is set. Channel responds with "select out" causing TC to send "operational in,"

and its own address on the bus in lines accompanied by "address in" tag. After the channel "command out" response falls, TC can gate "service in" for the next data byte.

When the byte is set into the data register and "service out" falls, channels drops "select out" allowing the reset of the operational in trigger. Channel is again free of the TC. This procedure is repeated for *each data byte* into the data register. Refer to "Interrupt and/or End Status Interface" logic for details.

Read/Write Register to Tape Unit Character Transfer

The previous mode set command designates whether characters recorded on magnetic tape be written in odd or even parity. Even parity can be used only in 7-track operation. (Mode set designates whether characters in the record should contain an odd or even number of bits.) If odd parity is specified, the odd redundancy trigger was set at the beginning of the write operation. Vertical redundancy check circuits examine the bit structure of the character stored in the R/W register. If the R/W register character contains incorrect parity, a check bit (P bit) is added by the R/W VRC circuit to the character transferred to the TU.

A C-Compare check is also made at the R/W VRC circuits. Refer to C-Compare section of "Functional Units."

Cyclic Redundancy Check Register (CRCR): An additional error checking and correction feature is employed for nine-track operation *only*. As each character is gated into the R/W register at the fall of WC-2, the R/W register outputs condition the CRCR inputs. At WC-11, the character is set into the CRCR as each register position is "shifted" in a predetermined pattern. A shift occurs during each write clock cycle to create a special check character after all data characters are written.

At the end of the record, all positions except 2 and 4 of the CRCR are complemented and the resulting CRCC is written on the tape four character spaces after the last data character and four spaces before the LRCC. The LRCR considers the CRCC as a normal character when the LRCC is generated. Refer to the detailed explanation of "Error Correction" in the "Functional Units" section of this manual.

Echo Error Check

As mentioned earlier, WC-2 gates the data register character to the R/W register. At WC-5 the no echo (test) trigger is set in preparation for a no echo error test (Figure 211). The WC then generates a write pulse which rises at 6 time and drops with the fall of 10 time (Figure 335, coordinates 4C-D). Write pulse causes the TU to accept the character on the write bus lines fed by the R/W register outputs. As the TU writes the character, an echo pulse is sent back to TC from any track writing a bit. Any echo pulse resets the no echo (test) trigger (Figure 211, coordinates 2E-F). If no echo returns, the test trigger stays on and a equipment check is indicated in sense byte 1 while a unit check error is also indicated in the end status byte.

Stop Command

A "service in-service out" sequence is generated for each data byte of the record. When channel responds to a "service in" request for data with a "command out" instead, "stop command" is brought up (Figure 335, coordinates 2D) to set the stop trigger, 1D (objective 12, Figure 335). This indicates that the previous "service out" set the last character of the record in the data register during the preceding write clock cycle. During the last clock cycle (which began the "service in-command out" sequence), the last character is set into the R/W register at WC-2, gated to the CRCR, and written on tape at WC-6 to WC-10.

The stop trigger resets the not stop data trigger (Figure 408) to bring up "stop data transfer." This line gates WC 14-15 to reset write condition and reset the write clock at 15.

Write Disconnect Delay

Write disconnect delay (WDD) used to:

1. Control timings when writing check character(s).
2. Measure time before dropping "go" to TU to establish stop point in the gap.

- **WDD conditions delay counter in microsecond mode.**
- **WDD is conditioned when "write condition" is deactivated.**
- **Length of the WDD period is variable, depending upon model of TU and seven- or nine-track.**
- **Write clock does not run during WDD, but read checking continues through WDD.**

Not "write condition" brings up "write disconnect delay" (WDD), 3B, to condition delay counter microsecond drive circuits. The delay counter steps at a frequency determined by TU model, and the bit density previously designated by the mode set command. At "WDD-27," the R/W register is reset.

As the delay counter advances further, the following occur:

Nine-Track Operation:

1. WDD-35-42, shift CRCR once more.
2. WDD-47, set CRC character (CRCC) into R/W register.
3. WDD-53, generate write pulse to write CRCC.
4. WDD-95, generate write pulse control to set all one's

character into R/W register (to deskew LRCC) and condition TU write triggers to be reset (Figure 335 coordinates 5D).

5. WDD-117, (TU Models 1 and 2 only) reset write trigger release trigger to generate write pulse and write LRCC (coordinates 4D), also reset delay counter and gate set of RDD trigger (coordinates 3B).

6. WDD-117, (TU Model 3 only) reset write trigger release trigger to generate write pulse and write the LRCC.

7. WDD-403, (TU Model 3 only) set end WDD trigger, 5B, to reset delay counter and go trigger, drop WDD, and condition RDD trigger set with not "go."

Seven-Track Operation:

1. WDD-47, generate write pulse control to set all one's character into R/W register (to deskew LRCC) and condition TU write trigger to be reset.

2. WDD-53, reset delay counter and write trigger release trigger to generate write pulse and write LRCC.

3. WDD-53, (TU Models 1 and 2 only) gate set of RDD trigger at fall of write trigger release.

4. WDD-53, (TU Model 3 only) after reset of delay counter, continue WDD in millisecond mode (coordinates 3B).

5. WDD-18, (MS model, TU Model 3 only) set end WDD trigger (5b) to reset delay counter and go trigger, drop WDD, and condition RDD trigger set with not "go."

The various WDD delays allow the TU to space the check character(s) further apart than normal character spacing, and away from the last data character of the record. Except for seven-track Model 3 operation, WDD timings also allow tape movement at full speed for predetermined distances after writing the check character(s) and before go is reset. These delays compensate for the different speeds at which various tape unit models move tape and for the greater distance between the write and read heads on seven-track head assemblies than on nine-track assemblies. The objective is to write a standard size gap no matter what speed the tape runs.

Stop Tape Motion

Read condition, set at the beginning of the operation, is still on to complete read checking (discussed in the next section). "RC-7 or -9" sets the RDD trigger (Figure 335, coordinates 3B) after fall of write trigger release (Models 1 and 2), or at the fall of the go trigger (Model 3). RDD trigger conditions the delay counter in microsecond mode (read disconnect delay timing) to prepare for the end of the operation. At "RDD-36," the go trigger is reset for Model 1 and 2 operation. However, because of the slower speeds of the Models 1 and 2, "RDD-36" does not occur until just before the check character(s). This is discussed in greater detail in the next section. For a Model 3 TU, the go trigger is reset during WDD before RDD-36.

After the go trigger is off, the remainder of the record is checked during "full coast time" of the Model 3 TU.

Read Check of Write Operation

As the tape record is written, tape moves away from the TU write heads and passes over the read heads. Characters of the record are read, after traveling the distance between the heads, and sent back to TC over the TU read bus. As each character enters the TC read circuits it is examined to determine:

1. Whether all the bits written on magnetic tape are of sufficient strength to be read in subsequent read operations.

2. Whether the character contains an odd or even number of bits and in seven-track whether the bit count matches the state of the odd redundancy trigger.

3. Whether spacing errors (skew errors) occurred in recording the character.

4. The correct bit structure for the record's check character(s). The CRCC is checked for VRC (nine-track only). The LRCC is also checked for VRC.

Each input character from the TU enters final amplifiers in the TC. Final amplifiers contain nine tracks, one for each bit position. Each final amplifier track has two outputs, a high-clip and a low-clip output. Input bits to corresponding final amplifier tracks must meet predetermined minimum amplitude requirements to produce high- and low-clip outputs. The high-clip output from each final amplifier track sets the correspond ing high-clip skew register. The low-clip output from each final amplifier track has no function during a read checking operation, although low-clip is gated out for seven-track operation. Theoretically, all bits in a character should arrive at the input to the final amplifiers simultaneously; however, all bits need not arrive at exactly the same time. TC does make a check that all bits do arrive within a specific time; this is known as a skew check.

Skew Check

The first high-clip output from any final amplifier track sets the first bit trigger to start the read clock for each character.

The skew gate trigger (Figure 210) turns on when the read clock advances to:

1. 4 if the selected tape unit is not operating at 800 BPI (all models).

2. 5 if a Model 1 or 2 tape unit is selected and is operating at 800 BPI.

3. 6 if a Model 3 tape unit is selected and is operating at 800 BPI.

All bits in the character must be stored in the high-clip skew register before RC-4, -5, or -6 time, because

any high-clip output from the final amplifiers after the skew gate trigger is set causes tape control to indicate a skew error and data check. This, in turn, will present a unit check bit (6) in the end status byte.

Skew Register VRC

Another data check performed on the newly written information is the high-clip VRC circuit.

Vertical redundancy check circuits examine the character stored in the high-clip skew register for parity error. If the high-clip skew register character contains a parity error, TC sets the high- clip VRC and data check triggers at RC-7 time. See "High-Clip Skew Register" VRC description for complete details.

For a nine-track operation, the high-clip character is gated to the LRCR. For seven-track, the low-clip register is gated to LRCR only if a high-clip VRC occurs; an LRC check is made at the end of the operation.

Read Disconnect Delay

Tape control processes characters from channel and writes them on tape at the same time that it checks characters written on tape earlier in the operation. Tape control completes actions involved in transferring characters from channel to the tape unit before it completes the read check operation. When the tape unit has written the check character(s) and the read clock advances to 7, the RDD trigger is set and steps the delay counter in microsecond mode between each read clock cycle (Figure 335, coordinates 3-4B). The "on" state of the RDD trigger does not signal that this is the last read clock cycle, nor even the next to the last read clock cycle; it merely warns that only a few characters in the record are left to be read and transferred to tape control.

The RDD trigger is generally used to drive the delay counter while timing the distance between bytes. During read checking of a write operation, the RDD trigger is not used until after writing is completed. While writing is in progress, the RDD trigger does not check the spacing between bytes. When writing is completed the RDD trigger can be set at read clock 7 time and reset by the following read clock 2 time. The delay counter runs in microseccnd mode, when the RDD trigger is on, to find the check character spacing at the end of the block. If the RDD trigger was set during the previous read clock cycle, the next character from the TU should produce a high-clip output from the final amplifiers and set the first bit trigger to start the read clock. The read clock output at RC-2 time resets the RDD trigger, causing the delay counter to reset. The read clock output at RC-7 time again sets the RDD trigger to re-start the delay counter in microsecond mode.

Because a check character is spaced further from the last normal character in the record than normal character spacing, the delay counter advances to 36 after tape control checks the last normal character in the record. RDD-36 sets the first check character trigger, 4B, indicating that the next character transferred from the TU is either the CRCC (nine-track operation) or the LRCC (seven-track operation). The output of the first check character trigger AND'ed with "write op" finally resets the go trigger for Model 1 or 2 operation. The RDD trigger remains on during check character time.

The first high-clip output from the final amplifiers after RDD-36 sets the first bit trigger to start the read clock for a check character. For nine-track operation, the CRCC is checked for parity, and gated to the LRCR. Check character VRC is explained in the functional units section under "Skew Register VRC."

At RDD-95, the second check character trigger is set, though it has no major function during a write operation (read check). When the LRCC is sent from TU, it is also checked for parity and gated to the LRCR.

Operation for seven-track is similar, except that the first (and only) check character to arrive from TU is the LRCC.

The first and second check character character triggers are necessary because it is possible for the check character(s) to contain no bits. At RDD-169 the LRCR sample is generated and actions to end the write operation begin. With the first check character trigger on, the RDD trigger is not reset at RC-2 and the delay counter was allowed to advance.

LRCR Error

At the end of a write operation, the state of each tape unit write trigger makes up the LRC character. As all write triggers are reset, the LRCC is written. For seven-track operation, the LRCC may be odd or even; in nine-track, LRCC will always be odd.

As the tape record is read checked, each character, including the CRCC, is set into the LRCR. When the LRCC is read and set into the LRCR, all LRCR triggers should end up in the *off* state.

An LRCR error is signaled if *any* LCRC trigger is still on at RDD 169. A complete description of LRC checking is in the "Functional Units" section.

End Operation

At RDD 171-75, TAU end reset is brought up to generate an end pulse and end reset. End pulse sets the channel end (status) trigger and device end trigger for the selected TU. The on state of the channel end trigger begins the end status in sequence discussed in a separate section.

Motion Control Commands

- **Motion control commands are eight commands that do not require channel operation during execution.**

- **Command byte denotes motion control commands by placing all ones in bus out positions 5, 6, and 7.**
- **Command byte denotes the type of motion control command by contents of bus out positions 2, 3, and 4.**
- **Command register P position is always set for motion control commands.**
- **No data is exchanged between channel and TC for any motion control command.**
- **At the end of initial selection, a forced channel end bit is submitted via "status in."**
- **All motion commands except rewind/unload submit an end status byte, containing a "device end" bit, via multiplex entry upon completion of execution.**

The term motion control implies that this type of command requires tape unit motion and no data transfer to or from channel during execution. These commands are:

1. Erase gap (ERG)
2. Write tape mark (WTM)
3. Forwardspace record (FSR)
4. Forwardspace file (FSF)
5. Backspace record (BSR)
6. Backspace file (BSF)
7. Rewind (REW)
8. Rewind/unload (RUN)

Of these eight commands, six (numbers 1-6) require the joint effort of both the tape control unit and the selected tape unit during the execution of the command. The rewind and rewind/unload commands are executed by the selected tape unit itself after TC has instructed the TU to rewind or rewind/unload.

Figure 101 shows the command byte bit configuration for each of these eight motion control commands. Notice that bus out positions 5, 6, and 7 are equal to 111 for all; however, the command byte differentiates between each by the bit configuration of bus out positions 2, 3, and 4. Notes 1 and 2 of Figure 101 describe how the command byte is transferred into the command register.

Any motion control command (P trigger in the command register, Figure 101) is a "valid control command," provided no unit check or busy condition for the selected TU is detected during initial selection. The line "valid control command" forces a "channel end" status bit into the initial status byte (Figure 302, coordinates 4-5C).

The control immediate trigger is set during a control command initial selection if channel accepts the initial status byte that contains the "channel end" status bit "forced" by "valid control command" (Figure 330, coordinates 3B). If channel did not accept the "channel end" status (command out response to "status in"), control immediate is not set.

Motion control commands do not require channel operation. After initial selection is complete, channel disconnects (reset operational in). The TC and/or the TU will complete the operation without channel. After the operation is complete, TC must "interrupt" channel with "request in" for the end status byte. Figure 303, coordinates 4A, illustrates how motion control commands reset the "burst mode" latch which will allow the "operation in" trigger (Figure 301, coordinates 2C) to be reset when channel drops "select out."

Erase Tape (ERG)

- **The erase command erases 3¾ inches of tape.**
- **Bring up "write op" and "ERG" lines with no data.**
- **Do turnaround if tape moved backward for previous operations.**
- **Condition "write delay."**
- **Start tape motion in write status. (Energize erase head to erase tape.)**
- **Stop tape motion after long (LP) write delay.**
- **Generate resets and begin end status in sequence.**

The erase tape command provides the ability to erase 3¾ inches of tape. Timing used to measure the erased portion of tape is the same as for write load point gap. At the end of a "load point delay" the tape control merely stops tape motion without writing on the tape. Figures 336 and 504 illustrate erase operation.

After initial selection is complete, "status in and command or service out" sets the tape op sync trigger. The operation will proceed even if channel rejects the initial selection status byte with command out. The fall of either "out" tag sets the tape op trigger to bring up "ERG" and "write op." "Write op" brings up write delay (WD) if turnaround is inactive (Figure 336, coordinates 2A). A turnaround is indicated if "backward memory status" is active from TU. This means the TU just completed a backward operation and requires time to reverse mechanical functions. Figure 504 is a flow chart of erase command.

Not "backward memory status" gates "set write status" and sets the TU R/W trigger to energize the TU erase head and put the TU in write status.

Write Delay

Write delay gates millisecond control to advance the delay counter at a rate determined by the TU model. Write delay also sets the go trigger to send "go" to the TU. The prolays are energized and begin to move tape forward (objective 4-7 on Figure 336).

When the delay counter output reaches 215 (Model 1 or 2) or 320, a line called "WD-215 + 320" resets the first character trigger to drop WD and set the RDD trigger. The delay counter is also reset at 215 or 320. RDD gates microsecond control to the delay counter (objective 8-10).

Stop Tape Motion

When the delay counter reaches 36 in microsecond mode, RDD-36 resets the go trigger (objective 11). As the delay counter advances to 171, RDD 171-75 generates a TAU end reset, end pulse, and end reset to reset most of TC. Tape has been erased for approximately 3.75 inches.

End Status

"End pulse" sets the device end trigger for the selected TU. Because the TU not ready line from TU is inactive, a "selected device end" is generated immediately; this line begins the end status (request in) interrupt if the channel does not have "suppress out" conditioned.

The end status byte will contain:

1. Device end.
2. If control immediate trigger is off, channel end bit.
3. Unit exception (if TI is on in TU).
4. Unit check (if TU dropped "ready").
5. Control unit end (if control immediate is on *and* unit check or unit exception is active).

Write Tape Mark (WTM)

- "WTM" and "write op" lines are conditioned at initial period of operation.
- A write delay period is initiated but it extends out to WD-215 or WD-320 to produce a 3¾ inch erased area — commonly referred to as "load point skip."
- At end of WD, write condition and read condition are activated to allow running of both write and read clocks.
- Tape mark configurations (seven-track or nine-track, are forced in the R/W register.
- Seven-track WTM forces even redundancy. Tape mark character is written at write pulse time: WC-6 to WC-10.
- Stop trigger is set at WC-8-9 time; simulates a "command out" from channel during write command.
- At end of the only write clock cycle, WDD is conditioned in microsecond control to write check character (LRC).
- WTM cannot write a CRC character; it is blocked.
- Ending sequence, including read check cycle and dropping of "go" and error detection is identical to the write command.

Figures 335 and 503 should be used with the following discussions. The objective listed under WTM on Figure 335 supplements the description of the circuits utilized in this operation.

After initial selection is complete, "status in and command or service out" sets the tape op sync trigger. The fall of either channel out tag is gated by tape op sync to set the tape op trigger (4A on Figure 335). Command register outputs P, 6, 7, and not 5 gated by tape op trigger bring up WTM which, in turn, brings up "write op." "WTM" also sets the LP delay trigger, 2E. The LP trigger will cause load point write delay before the tape mark is written.

Seven-Track Tape Mark is a four-bit single character record with bits in positions 4, 5, 6, and 7. The "WTM" line forces 1 bits into positions 6 and 7 of the R/W register.

WTM AND's with seven-track to force bits 4 and 5 into the R/W register, and "even redundancy" at the mode register output (Figure 18). Only an LRCC check character is written after the tape mark, no CRCC.

Nine-Track Tape Mark is a single-character record. The WTM line forces bits 6 and 7 into the R/W register. The WTM line AND's with "nine-track" to bring up "nine-track WTM"; this line forces "odd redundancy," and a bit into position 3 of the R/W register. A normal record in nine-track operation writes two check characters, CRCC and LRCC. The nine-track WTM operation causes a CRCC to be generated, but only the LRCC is written.

Write op conditions write delay (WD) immediately if "turnaround" is inactive. A turnaround is indicated if "backward memory status" is conditioned from TU. Backward memory is brought up by the on condition of the TU backward trigger. A turnaround sequence delay allows the TU enough time to reverse its mechanical functions if the previous operation moved tape backward.

If backward memory status is inactive, write op gates set write status to the TU; otherwise, "D 96 turnaround" gates "set write status" during the turnaround sequence. Set write turns on the TU R/W trigger to set the TU in write status.

Write Delay and Tape Motion

Write op now brings up WD to gate millisecond control (Figure 335, 3B) and set the go trigger. "Go" is gated to the TU (5B) as millisecond mode advances the delay counter at a rate determined by the model of the selected TU. When delay counter output reaches 215 (Models 1 or 2) or 320 (Model 3), "WD 215-320" line (2D) sets the write condition trigger, which sets the read condition trigger (3D).

Condition Write and Read Circuits

The write condition trigger:

1. degates "WD"

2. sets the write trigger release trigger
3. gates start of the write clock (4D).

Write clock pulses are necessary to write the tape mark and gate the end of the operation.

Write the Tape Mark

Tape is now moving at full speed, the long write delay is complete, the tape unit is in write status, and the tape mark is in the R/W register which gates the TU write bus. At the first WC-6, a long write pulse is generated (falls at WC-10) to write the tape mark (4C). At WC 8-9 the stop trigger is set (1D) to gate the reset of write condition with the stop data transfer line. At WC 14-15 the write condition trigger is reset.

The fall of write condition stops the write clock which coasts to 15. "Not write condition" brings up WDD (write disconnect delay) to gate microsecond mode to the delay counter.

Write the LRCC

To deskew the check character, an all ones character is gated to the R/W register at WDD-47 for seven-track operation, or WDD-95 for nine-track operation. When the write trigger release trigger is reset, a write pulse is generated to write the longitudinal redundancy check character (LRCC).

Stop Tape Motion

Ending conditions for a WTM operation vary depending upon the number of tracks and tape unit model (the same as a normal write operation). A summary of these conditions follows:

Seven-Track Model 1 or 2 TU:

1. WDD-53 — reset write trigger release and write LRCC.
2. Condition RDD at fall of write trigger release.
3. Advance delay counter in microsecond mode
4. RDD-36 — reset go trigger to drop "go" to TU.

Seven-Track Model 3 TU:

1. WDD-53 — reset write trigger release and write LRCC
2. Reset delay counter, continue WDD in millisecond mode
3. Advance delay counter in millisecond mode
4. WDD-18 — reset go trigger to drop "go" to TU and condition RDD; reset delay counter
5. Advance delay counter in microsecond mode.

Nine-Track Model 1 or 2 TU:

1. WDD-117 reset write trigger release and write LRCC
2. Condition RDD at fall of write trigger release
3. Advance delay counter in microsecond mode
4. RDD-36 — reset go trigger to drop "go" to TU.

Nine-Track Model 3 TU:

1. WDD-117 reset write trigger release and write LRCC
2. WDD-403 set end WDD trigger to reset the go trigger and delay counter
3. Not go drops WDD, sets RDD trigger, and drops "go" to TU
4. Advance delay counter in microsecond (RDD) mode.

Read Check of the TM Record

In the preceding paragraphs, RDD was said to have begun and the delay counter was running in microsecond mode. This may not be true at this point, because the fall of write trigger release or go only *gates* set of the RDD trigger; RC-7 actually turns the trigger on. Though the TM and LRCC are written, they require time to travel the distance between the write and read heads. The read clock cannot run until a first bit is detected in the final amplifier high-clip circuits.

The final amplifiers and other read circuits are conditioned. When the TM is finally read at TU, it is sent over the read bus to the final amplifiers and set in the high-clip register (also low-clip for seven-track operation). "First bit" sets the first bit trigger to start the first read clock cycle. The TM is checked for skew (like any normal character) at RC-4, 5, or 6, depending on TU model and density. At RC-7 the RDD trigger is finally set and the TM is set into the LRCR. The delay counter advances in RDD microsecond mode. The TM is also checked for parity in the high-clip register.

When the delay counter advances to 36, RDD-36 will reset the go trigger for model 1 or 2 tape units. RDD-36 also sets the first check character trigger as the LRCC approaches the read heads. When the check character is read, it is also checked for skew and parity, and set into the skew register(s) because the read clock was again started by the first bit trigger (reset previously by the read clock reset). The LRCC is set into the LRCR at RC-7.

The delay counter continues to advance in RDD μS mode because the first check character trigger degates the normal reset to the RDD trigger. At RDD-169, a check character sample determines that all LRCR triggers are off, or an LRCR error and data check is indicated.

At RDD 171-75, TAU end reset is generated to bring up end pulse and end reset. The tape mark operation is complete except for the end status sequence.

End Status Interrupt

End pulse sets the device end trigger for the selected TU. Because the TU not ready line from TU is *inactive*, a "selected device end" is generated immediately; this line begins the end status (request in) interrupt if the channel does not have "suppress out" conditioned.

The end status byte will contain:

1. Device end
2. If control immediate trigger is off, channel end bit
3. Unit exception (if tape indicate is on in TU)
4. Unit check (if a data check occurred or the TU dropped "ready")

5. Control unit end (if control immediate is on *and* unit check or unit exception is active).

Forward Space Record and Forward Space File

- The forward space record will:
  1. Read over the next record on tape and then stop.
  2. In addition to the normal DE bit, cause a unit exception and control unit end bit in the end status byte if a tape mark character is read.
  3. Normally set the DE bit in the end status byte to indicate completion of the operation.
- The forward space file will:
  1. Continually read over records until a tape mark character is recognized.
  2. Send an end status byte with DE bit to channel upon completion of the operation.
  3. Reset "go" and "read condition" between records as it searches for the tape mark character.
- Forward space commands check for "noise" at the end of the record via "forward stop delay."
- Data flow for the forward space commands is from read bus to final amplifiers to skew register(s).
- Transfer data is deconditioned so data will not be sent to the R/W register.
- End status byte must enter via "request in," similar to a multiplex entry.
- During entire operation, channel is free, but TCU is occupied.
- Like all valid control commands, "operational in" was reset at end of initial selection.

In a forward space record operation, the selected tape unit moves tape forward through one tape record to the first interrecord gap, or to the gap between the last record and the end of a tape. In a forward space file operation, the selected tape unit moves tape forward to the first interrecord gap sensed after reading a tape mark record, or to the gap between the last record and the end of tape. The tape unit reads characters recorded on tape and conditions data lines to the tape control. Tape control does not execute data transfers either to channel or to the tape unit when performing the forward space record or forward space file operation. Figure 505 is a flow chart of forward space; Figure 334 is the second level diagram.

Condition Tape Control to Execute Forward Space Operation

After the initial selection sequence, tape control conditions "read operation and read delay" to:

1. Set the go trigger, causing tape on the selected tape unit to move forward.
2. Condition millisecond delay counter drive circuits; the delay counter advances in millisecond mode. If the load point delay trigger is set, the delay counter output at RD-103 or RDD-160 sets the read condition trigger. If the load point delay trigger is not set, a delay counter output sets the read condition trigger when the counter advances to:
   1. 15, if a Model 3 tape unit is selected.
   2. 17, if a Model 1 or 2 tape unit is selected.

   Read condition resets the delay counter and allows tape control to accept data that the selected tape unit transfers.

Process Data from Tape Unit

During forward space commands, TC merely interrogates the data read for:

1. tape mark records,
2. end of each record.

Bits in each character transferred from the tape unit enter corresponding final amplifier tracks (Figure 334, coordinates 1C). Each final amplifier track has a high-clip and a low-clip output; high-clip outputs set the high-clip skew register positions; low-clip outputs set the low-clip skew register positions. The first high-clip output from any final amplifier track in each character cycle sets the first bit trigger, causing the read clock to start. Tape control performs a normal read check operation on characters stored in the high-clip skew register and sets the LRCR at RC-7 time of each read clock cycle. However, results of read checks in forward space operations are ignored. The record being read has been checked previously, and errors in the record should have been detected in an earlier read or write operation.

The first character trigger, set during the initial selection sequence, is on during the first read clock cycle. If the first character received from the tape unit is a tape mark, the first character trigger on output allows tape control to set the first character tape mark trigger at RC-6 time.

The read clock output at RC-7 time sets the RDD trigger, causing the delay counter to advance in the microsecond mode.

The read clock output at RC reset time:

1. resets the skew registers,
2. resets the first bit trigger, causing the read clock to stop,
3. resets the first character trigger.

If the second character that the tape unit transfers to tape control is a data character, a final amplifier track produces a high-clip output to set the first bit trigger and start the read clock before the delay counter advances to 36. The read clock output at RC-2 time resets the RDD trigger and the delay counter. If the first character tape mark trigger was set in the

first read clock cycle, tape control unconditionally resets the trigger in this second cycle. The first character tape mark trigger functions to detect a two-character (tape mark character and check character) tape mark record. If the tape unit transfers two characters to tape control before transferring the check character, the record is not a tape mark record. The first character tape mark trigger has no purpose in the check operation on the record being read after tape control determines that the record is not a tape mark record.

Read Check Characters

The check characters are spaced further from the last normal character in the record than normal character spacing. The extended delay allows the delay counter to advance to 36 and 95 to set the first and second check character triggers.

The first check character trigger blocks the read clock 2 (reset) pulse to the RDD trigger. The delay counter runs, without stopping, throughout both check character cycles.

Forward Stop Delay

At RDD-171 the forward stop delay trigger turns on (Figure 334, coordinates 1E). Forward stop delay runs the delay counter in microsecond mode. If bits are detected before RDD-511 (about 40 character spaces), FSD noise is set and the delay counter is reset. Because the RDD trigger is still on, the delay counter starts counting again from zero. Tape motion will continue as long as bits are detected on the tape.

If no bits are received and the delay counter reaches FSD-511, a TAU end reset pulse will be generated to reset the "go" trigger and the read condition trigger. If the operation is forward space record, TAU end causes an end pulse which initiates an end status sequence.

File Search

During a forward space file operation, the tape control must continue reading records to look for a tape mark. At the end of each record the first character tape mark trigger is sampled. When the tape mark trigger is off, the reading sequence must be restarted to continue searching for a tape mark. The first character trigger (Figure 334, coordinates 1A) must be turned on by "file search and ungated TAU end" to restart.

If the tape mark trigger is off, the end pulse is blocked and the next record will be read. If the tape mark trigger is on, the end pulse is allowed to reset TC and initiate an end status sequence (Figure 334, coordinates 4D and 4E).

End Status Interrupt

"End pulse" sets the device end trigger for the selected TU. Because the "TU not ready" line from TU is inactive, a "selected device end" is generated immediately; this line begins the end status (request in) interrupt if the channel does not have "suppress out" conditioned.

The end status byte will contain:

1. Device end (FSF and FSR).
2. If control immediate is off, a channel end (FSF and FSR).
3. Unit check and control unit end if TU dropped ready (FSF and FSR).
4. Unit exception and control unit end if TM record (only FSR).

Backspace Record and Backspace File

- Backspace record will:
  1. Read back over one record or into load point.
  2. Set unit check and end operation if it reads into or is at load point.
  3. Normally send an end status byte containing a device end bit.
  4. Set control unit end and unit exception bit in addition to the normal end status if TM record.
- Backspace file will:
  1. Read backwards until a tape mark character is read or load point is sensed.
  2. Set unit check and end operation if TU reads into or is at load point.
  3. Normally send an end status byte to channel containing a device end bit.
- Backspace operations use the same data flow path as forward space.
- Delay counter will run in *millisecond* control during backward read delay and during entire record *including* check characters.
- 200 BPI is *forced* if backspace occurs on seven-track TU to allow for skew on records not written by 2400 tape system.
- Ending sequence for record and file which did locate TM is identical to read backward op. Determine "backspace reset rd cond" during millisecond control, then turn to microsecond control to end.
- Backspace file will *retain* "go" and "read condition" between records when file searching.

Condition Tape Control to Execute Backspace Operation

Figure 506 is a flow chart of backspace; the second level drawing is Figure 333. Consult the "objectives" on Figure 333 as the circuits are described. The backspace command brings up backward op. If the tape unit is at load point, the unit check trigger is turned on and backward disconnect generates a TAU end. Backward at load point causes an end pulse which turns on channel end and initiates an end status sequence.

The tape unit may be at load point during the initial selection status sequence, however, the backward at load point line cannot come up until after initial selection. For this reason, an end status byte is sent to channel after initial selection is finished.

If the load point trigger is not on, the tape control will proceed to execute the backspace. After the possibility of a turnaround sequence has been checked, backward op brings up backward read delay which turns on the go trigger and runs the delay counter in millisecond mode. Backward op and backward read delay turn on the backward trigger to condition the backward line to the tape unit. The backward line turns on backward in the tape unit and also resets the tape unit read/write trigger to read status.

At the end of read delay the read condition trigger (Figure 332, coordinates 2B) is turned on by:

1. RD-5 for a Model 1 tape unit.
2. RD-8 for a Model 2 tape unit.
3. RD-12 for a Model 3 tape unit.

Read condition activates the first bit and read clock circuits to look for data from the tape.

Process Data from Tape Unit

Bits in each character transferred from the tape unit enter corresponding final amplifier tracks. Each final amplifier track has a high- and low-clip output; high-clip outputs set high-clip skew register positions; low-clip outputs set low-clip skew register positions. The first high-clip output from any final amplifier track in each character cycle sets the first bit trigger, causing the read clock to start. Tape control performs a normal read check operation on characters stored in the high-clip skew register and sets the LRCR at RC-7 time of each read clock cycle. However, results of the read checks in backspace operations are ignored. The record being read has been checked previously, and errors in the record should have been detected in an earlier read or write operation.

At read clock 2 time the second check character trigger is turned on. The second check character (LRCR) should be the first data read from the tape in nine-track backspace operation.

The first character trigger, set during the initial selection sequence, is on during the first read clock cycle. If the first character received from the tape unit is a tape mark, the first character trigger on output allows tape control to set the first character tape mark trigger at RC-6 time.

The read clock output at RC-7 time sets the RDD trigger, causing the delay counter to advance in *millisecond mode*. During backspace the tape control does not check the spacing between bytes. It only looks for the end of the block (actually the beginning). In millisecond mode the delay counter does not reach RDD-36 so the second check character trigger stays on.

The read clock output at RC reset time:

1. Resets the skew registers.
2. Resets the first bit trigger, causing the read clock to stop.
3. Resets the first character trigger.

In the second read clock cycle, the tape unit transfers a character to tape control soon enough to produce a high-clip output from a final amplifier track, set the first bit trigger, and start the read clock before the delay counter advances to:

1. 19, if a seven-track Model 3 tape unit is selected.
2. 21, if a seven-track Model 2 tape unit is selected.
3. 24, if a seven-track Model 1 tape unit is selected.
4. 7, if a nine-track Model 3 tape unit is selected.
5. 11, if a nine-track Model 2 tape unit is selected.
6. 14, if a nine-track Model 1 tape unit is selected.

Read clock 2 of each character turns off the RDD trigger and resets the delay counter. Backward motion will continue until a record gap, tape mark, or load point stops the operation.

The tape control examines the second character bit configuration. If the bits equal a tape mark and the first character tape mark trigger is on, the record may be a true tape mark. If no more characters are sensed, the record is a true tape mark. A true tape mark record is a tape mark and tape mark check character. When the bits in either character are not a tape mark, or if more than two characters are sensed, the record is not a tape mark. If not a true tape mark, the first character tape mark trigger is reset and the data is scanned to find the record gap.

At RC-6 of the second read clock cycle the first check character trigger is turned on. Read clock 2 of the next character will turn it off again. The cycle of "on at RC-6 and off at the next RC-2" will be repeated until the end of the record. After RC-6 of the last character, the first check character trigger remains on until TAU end reset.

End Backspace Operation

Between each read clock cycle the delay counter has been stepping in millisecond mode, gaging when the last character would be read. Like the read backward operation, when the last character is read, the delay counter will be allowed to step to a higher count. If the delay counter advances to the time designated for the selected tape unit, ("Note" on Figure 506) the tape control conditions backspace reset read condition, indicating that the tape unit has backspaced through one record and has sensed either an interrecord gap or the gap between load point and the first tape record.

In a backspace record operation, the go trigger and read condition are reset by backspace reset read

condition (Figure 333, coordinates 4A). After read condition and go are reset, the delay counter runs in microsecond mode. At RDD-36 time the second check character trigger is reset. RDD-171-175 generates a TAU end reset and an end pulse. The end pulse, in turn, initiates an end status sequence.

At the end of a record, during a backspace file operation, the first character tape mark trigger is sampled. If the tape mark trigger is on, backspace reset read condition is allowed to stop the operation. If the tape mark trigger is off, read condition and go will not be reset (Figure 333, coordinates 5A). The first character trigger is turned on by "continue backspace file" (Figure 333, coordinates 2A); end pulse is blocked (Figure 333, coordinates 4D). When the tape mark trigger is off, the next record will be read to search for a tape mark.

If load point is reached instead of another record, the tape unit automatically stops tape motion and sends an "at load point" signal to the tape control (Figure 333, cordinates 4E). "At load point" generates a backward disconnect and backward at load point. "Backward at load point" turns on unit check and causes an end pulse to initiate an end status sequence.

When a tape mark is detected during a *backspace record* (file search down), the unit exception trigger is turned on at RDD-169 (Figure 333, coordinates 6C). Unit exception is sent to the channel as the 7 bit in the end status byte.

End Status Interrupt

End pulse sets the device end trigger for the selected TU. Because the "TU not ready" line from TU is *inactive*, "selected device end" is generated immediately; this line begins the end status (request in) interrupt if the channel does not have "suppress out" conditioned.

The end status byte will contain:

1. Device end (BSF and BSR).
2. If control immediate is off, a channel end bit (BSF and BSR).
3. Unit check and control unit end if error existed; error caused by TU dropping ready or backspace into or at load point (BSF and BSR).
4. Unit exception and control unit end if tape mark record (only BSR).

Rewind (REW) or Rewind-Unload (RUN)

- A rewind command moves tape to load point. A rewind-unload command moves tape to load point then unloads the tape.
- Channel is freed at end of initial selection period by resetting "operational in" and submitting a status byte containing a forced channel end (bit 4).
- TCU checks for possibility of a "turnaround" sequence following initial selection period.
- Using existing "backward op" circuits, TCU produces a "rewind" or "rewind-unload" line to the selected TU.
- As soon as TU begins to rewind, the TCU is informed via "TU status B" active. A "backward disconnect" is generated, which, in turn, fires the end pulse singleshot to reset and free the control unit.
- At "backward disconnect" time, the device end scanner is *armed* via end pulse, to provide a selected device end when:
  1. TU completes rewind operation or,
  2. TU is made ready by operator if rewind-unload.
- At "backward disconnect" time, TCU will submit an end status byte, via multiplex entry, for rewind-unload command. Status byte will normally contain:
  1. *Forced* device end bit.
  2. Unit check bit.
  3. Control unit end bit.

A rewind command instructs the tape unit to move tape backwards until load point is reached. A rewind-unload command rewinds the tape to load point and then unloads the tape. If the tape is at load point a rewind-unload only unloads the tape. Figure 30 is a simplified timing chart of rewind and rewind-unload.

After initial selection, channel is free to perform other operations. At the end of initial selection, "valid control command" forces a "channel end" status bit in the initial selection status byte (channel end trigger is not on at this time).

After the tape unit begins to rewind, the tape control is free to perform other operations. When the tape unit has accepted a rewind-unload and begins rewinding, the tape control transmits an interrupt status byte to the channel. A rewind-unload generates another interrupt (third status byte) when the operator manually reloads the tape and makes the tape unit ready again.

A rewind command does not transmit an interrupt until the tape reaches load point. When the tape unit completes the rewind, a selected device end interrupt (second status byte) is generated to inform the channel.

Initiate Rewind or Rewind-Unload

Figure 508 is a flow chart of rewind and rewind-unload. starting with initial selection. "Valid control command" forces a "channel end" bit into the initial selection status byte.

After initial selection is complete, "status in and command or service out" sets the tape op sync trigger which sets the tape op trigger. Tape op gates the command register P trigger (and not 6 trigger) to bring up "backward op" (Figure 330, coordinates 1A).

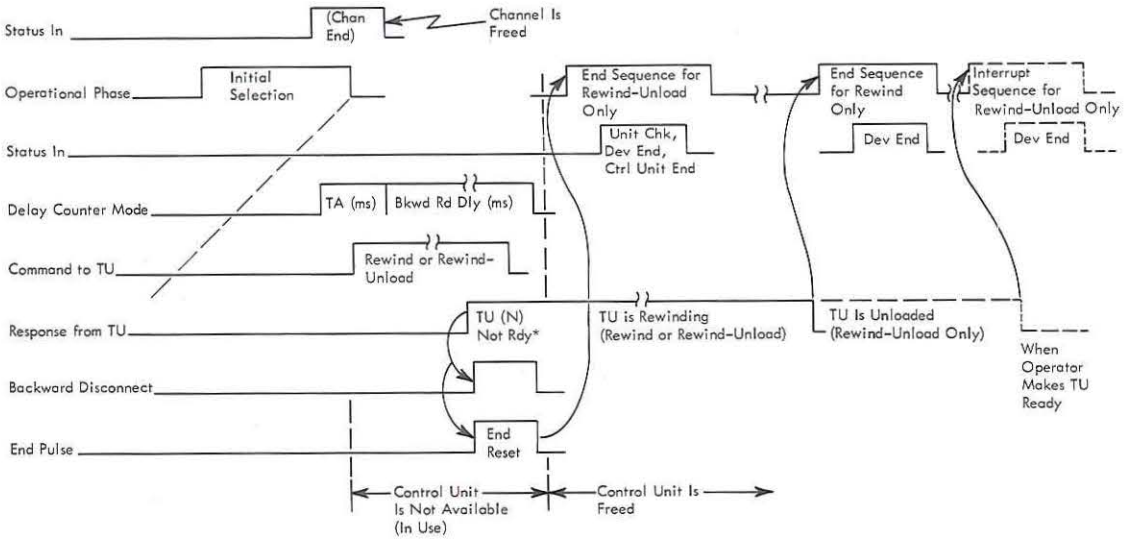


Figure 30. Simplified Rewind-Run Operation

Load Point

If the tape on the selected TU is already at load point (LP), and a rewind (REW) is designated, "backward op" brings up "backward disconnect." (Objective 13, Figure 330.) The rewind operation is unnecessary because the selected TU is already at load point.

Note that a rewind-unload is not complete until the TU is at load point and unloaded.

An important point to consider is the fact that the LP condition existed during initial selection. Programming procedures are simplified if the LP condition at selection time is treated no different than after a tape rewinds to LP. Therefore, the TC does not include "device end" in the initial selection status byte, but causes an end status (device end) interrupt after initial selection is complete.

Backward Read Delay

"Backward op" brings up "backward read delay" immediately if "turnaround" is inactive (Figure 330, coordinates 1A). A turnaround is indicated if "backward memory status" is not conditioned by TU and a backward operation is designated. Backward memory is brought up by the on condition of the TU backward trigger. A turnaround delay sequence allows the TU enough time to reverse its mechanical functions if the preceding operation moved tape in the opposite direction.

If the TU is in write status, the turnaround must erase tape forward (in write status) to lengthen the short gap area after the last good record; this erases old record information which could cause "noise bits" at the wrong time during a subsequent read operation.

The turnaround sequence (explained in another section) sets the TC backward trigger with "D 96 turnaround." The TC backward trigger sends "backward" to the TU and sets the TU backward trigger. TU then conditions the "backward memory status" line to TC. The TU backward trigger also resets the TU read/write trigger to bring up "read status."

Start Operation

All circuits used in the start operations are available on Figure 330 and described by objectives 6 through 11. After turnaround is complete (if it is necessary) "backward memory status" brings up "backward read delay," sets the go trigger, gates millisecond mode to the delay counter, and gates either "rewind" or "rewind-unload" to the TU. Relays are picked in the TU to begin the designated operation. "Rewind-unload" also sets the rewind-unload trigger in TC (Figure 330, coordinates 1B).

"Go" is not sent to the TU; it is blocked by REW or RUN (Figure 330, coordinates 5A). Other backward operations that use the same circuits do send "go" to the TU; however, from a design standpoint it was simpler to use existing backward circuits and just block the go line for a REW or RUN.

The delay counter is used to turn on the after DC-3 trigger, which is used to monitor for a TU reject (Systems 09.21.1). A reject is indicated if the TU should drop mechanical ready while the TC is initiating the operation.

Reset Tape Control

As soon as either the load-rewind or rewind-unload relay is picked in the TU, the rewind trigger is set in the TU. The on state of the rewind trigger brings up "not ready," which TU sends to the TC as "TU(n) not ready," "TU status B" (not ready) is brought up by the rewinding trigger to AND with "backward op" and bring up "backward disconnect" (Figure 330, objectives 12 and 13).

Backward Disconnect brings up "TAU end reset" to generate an end pulse and end reset. The 800-nanosecond end pulse sets the device end trigger for the *selected* TU. The eight device end triggers (16 with the SXT feature) make up what can be considered a device end register (Figure 406). Setting of the trigger is referred to as arming for device end, because the trigger cannot cause a selected device end interrupt until the TU has either *finished* the rewind operation or the operator manually loads and makes the TU ready *after* completion of the rewind-unload operation. The rewind operation is now complete except for the end status byte when the TU reaches load point. At this time, TC is free to execute another command on a different TU since TC has been completely reset by the end pulse. If this were the rewind-unload command, at this time TC is also free by the end pulse reset, but TC must now generate and submit the end status byte via multiplex entry.

End Rewind-Unload is brought up by AND'ing the outputs of rewind-unl
output of the channe
tive 17). The 800-nan
channel end trigger, t
AND half, if the contr
end trigger is *off* (Fi

Also - Look @ next page!

Because the end p
the channel end latc
has enough time to set unit check (Figure 330, coordinates 3C), which sets the CU end trigger (Figure 330, coordinates 4B) and degates the channel end trigger reset; channel end trigger latches up and stays on.

Rewind-Unload End Status Interrupt is begun by the channel end trigger (on state) as soon as "backward disconnect" completes the end reset sequence. An interrupt is begun for rewind-unload (if "suppress out" is inactive) because once the TU is unloaded, the TU might not be reloaded manually for some time. Therefore, programming requires two requests in "interrupts":

1. "Device end" status when the TU begins rewinding.
2. "Device end" status when the TU is reloaded and made ready.

The first request in status actually contains "device end," "unit check," and "CU end" status bits if the control immediate trigger is on; "channel end" bit 4 is blocked (Figure 330, coordinates 5C). If control immediate is off, the status byte will contain "channel end" instead of "CU end."

An interrupt does not occur at this point for a rewind operation unless the control immediate trigger is off (channel rejected initial channel end bit). Notice on Figure 330, coordinates 1C, that the channel end latch would be allowed to set at "backward disconnect" time (end pulse) if the control immediate latch were off. Channel end would then initiate a multiplex interrupt to re-submit the previously rejected channel end status (Figure 22, Group 2).

Device End Scanner

The TC completed its part of either operation when the end resets occurred. When the TU completes the REW or RUN (after manual intervention), the "TU not ready" line is dropped by the TU (TU is ready) and a "selected device end" is generated. However, the TU select trigger (Figure 406), set during initial selection, was reset at the fall of "end pulse" at backward disconnect time. With this trigger off, the device end scanner cycles and the TU select address is lost.

The device end scanner is running (cycling) to "pick up" a device end (not rewinding) condition from one or more other tape units which might also be rewinding. The first TU to complete its operation will stop the scanner at its address; after channel has accepted the device end, the scanner again runs continuously or advances and stops at the next sequential address of another TU with a device end, and so on.

The device end scanner consists mainly of the TU select register triggers, the device end (register) triggers, and the TU rewinding (register) triggers. As long as no TU has its device end trigger on (armed) and its rewinding trigger *off*, the TU register triggers cycle (turn on and off sequentially – ring fashion).

As soon as "TU not ready" line drops (and its device end trigger is armed), the scanner stops to indicate the TU address. The scanner cannot start again until a device end reset is generated when channel takes the "device end" status; this reset only resets the device end trigger of the address indicated by the scanner (the address of the now selected TU).

Chain Trigger

The device end scanner can cycle only if the TU selected trigger is off (Figure 406). If the chain trigger is on, the TU select trigger cannot be reset. With both triggers on, the TU select register (scanner) retains the original TU address giving the selected TU device end priority. That is, after the "end pulse" arms the device end trigger, the TC cannot accept a device end (not rewinding) condition from the selected TU. The TC is not really "free" unless the chain trigger is off.

The chain trigger is set only during a status in sequence if channel conditions "suppress out." Once set, the trigger is held "on" until suppress out drops or channel initiates a selective reset.

Selected Device End

The "TU not ready" line is dropped by a TU when a rewind operation is complete, or after an operator reloads a TU and makes it ready after a rewind-unload

Backward Disconnect brings up "TAU end reset" to generate an end pulse and end reset. The 800-nanosecond end pulse sets the device end trigger for the *selected* TU. The eight device end triggers (16 with the SXT feature) make up what can be considered a device end register (Figure 406). Setting of the trigger is referred to as arming for device end, because the trigger cannot cause a selected device end interrupt until the TU has either *finished* the rewind operation or the operator manually loads and makes the TU ready *after* completion of the rewind-unload operation. The rewind operation is now complete except for the end status byte when the TU reaches load point. At this time, TC is free to execute another command on a different TU since TC has been completely reset by the end pulse. If this were the rewind-unload command, at this time TC is also free by the end pulse reset, but TC must now generate and submit the end status byte via multiplex entry.

End Rewind-Unload is brought up by AND'ing the outputs of rewind-unload trigger set earlier and the OR output of the channel end trigger (Figure 330, objective 17). The 800-nanosecond end pulse tries to set the channel end trigger, but the trigger is held reset at the AND half, if the control immediate trigger is on *and* CU end trigger is *off* (Figure 330, coordinates 4C).

Because the end pulse brings up the OR output of the channel end latch for 800 ns, "end rewind-unload" has enough time to set unit check (Figure 330, coordinates 3C), which sets the CU end trigger (Figure 330, coordinates 4B) and degates the channel end trigger reset; channel end trigger latches up and stays on.

Rewind-Unload End Status Interrupt is begun by the channel end trigger (on state) as soon as "backward disconnect" completes the end reset sequence. An interrupt is begun for rewind-unload (if "suppress out" is inactive) because once the TU is unloaded, the TU might not be reloaded manually for some time. Therefore, programming requires two requests in "interrupts":

1. "Device end" status when the TU begins rewinding.
2. "Device end" status when the TU is reloaded and made ready.

The first request in status actually contains "device end," "unit check," and "CU end" status bits if the control immediate trigger is on; "channel end" bit 4 is blocked (Figure 330, coordinates 5C). If control immediate is off, the status byte will contain "channel end" instead of "CU end."

An interrupt does not occur at this point for a rewind operation unless the control immediate trigger is off (channel rejected initial channel end bit). Notice on Figure 330, coordinates 1C, that the channel end latch would be allowed to set at "backward disconnect" time (end pulse) if the control immediate latch were off. Channel end would then initiate a multiplex interrupt to re-submit the previously rejected channel end status (Figure 22, Group 2).

Device End Scanner

The TC completed its part of either operation when the end resets occurred. When the TU completes the REW or RUN (after manual intervention), the "TU not ready" line is dropped by the TU (TU is ready) and a "selected device end" is generated. However, the TU select trigger (Figure 406), set during initial selection, was reset at the fall of "end pulse" at backward disconnect time. With this trigger off, the device end scanner cycles and the TU select address is lost.

The device end scanner is running (cycling) to "pick up" a device end (not rewinding) condition from one or more other tape units which might also be rewinding. The first TU to complete its operation will stop the scanner at its address; after channel has accepted the device end, the scanner again runs continuously or advances and stops at the next sequential address of another TU with a device end, and so on.

The device end scanner consists mainly of the TU select register triggers, the device end (register) triggers, and the TU rewinding (register) triggers. As long as no TU has its device end trigger on (armed) and its rewinding trigger *off*, the TU register triggers cycle (turn on and off sequentially – ring fashion).

As soon as "TU not ready" line drops (and its device end trigger is armed), the scanner stops to indicate the TU address. The scanner cannot start again until a device end reset is generated when channel takes the "device end" status; this reset only resets the device end trigger of the address indicated by the scanner (the address of the now selected TU).

Chain Trigger

The device end scanner can cycle only if the TU selected trigger is off (Figure 406). If the chain trigger is on, the TU select trigger cannot be reset. With both triggers on, the TU select register (scanner) retains the original TU address giving the selected TU device end priority. That is, after the "end pulse" arms the device end trigger, the TC cannot accept a device end (not rewinding) condition from the selected TU. The TC is not really "free" unless the chain trigger is off.

The chain trigger is set only during a status in sequence if channel conditions "suppress out." Once set, the trigger is held "on" until suppress out drops or channel initiates a selective reset.

Selected Device End

The "TU not ready" line is dropped by a TU when a rewind operation is complete, or after an operator reloads a TU and makes it ready after a rewind-unload

operation. In either case, "not rewinding" (ready) turns off the rewinding trigger and conditions one leg of the selected device end AND circuit (Figure 406). The armed device end trigger conditions another input to the AND, and the last input is "TU select"; this line is brought up if the address is "chained," or the scanner has stopped at the correct TU address. When "made," the AND circuit brings up "selected device end" to begin the end status interrupt. The TC begins this sequence by generating a "request in" to channel if "suppress out" is inactive.

The end status byte will contain only a device end status bit unless channel does not yet have channel end status.

Non-Motion Control Commands

- Four non-motion control commands do not require tape unit motion for execution:
  1. No-op
  2. Diagnostic
  3. Mode set
  4. Request TIE.
- All four commands are identified by command byte positions 5, 6, and 7 equal to 011.
- Command byte positions 0 through 4 differentiate between each command.
- All non-motion control commands *except* request TIE:
  1. require only TC to complete operation
  2. complete execution of command during initial selection period
  3. force channel end and device end in initial selection status byte.
- Request TIE will require an "operate" time after initial selection to request *one* data byte from channel.

The non-motion control commands are commands which establish operating condition in TC for the execution of other types of commands. None of the non-motion control commands require a tape unit to be executed. These commands are:

1. No Op
2. Diagnostic
3. Mode set
4. Request TIE.

With the exception of the request TIE command, the commands are complete at the end of the initial selection period. Figure 21, Group 3, illustrates the basic time duration of these three commands and the contents of their status bytes. The request TIE command is used primarily to prepare TC to perform error correction on a succeeding nine-track read operation. Its functions are similar to a write command, writing one data byte.

Request TIE will require an operate time after initial selection. During this operate time, TC will request *one* byte of information from the channel (service in/out) and then terminate the operation with an end status byte containing channel and device end bits.

The command byte format for each non-motion control command is illustrated on Figure 101. Under the major heading of "Control" all of the commands are distinguishable by command byte positions 5, 6, and 7 equal to 011. Note that the mode set command byte can assume any of 15 different configurations to set the mode register to the various methods of handling seven-track data. During a mode set command, byte positions 0 through 4 actually transfer into the mode register and set it accordingly (Note 3). Each of these commands *does not* set the command register (last column, positions P, 5, 6, and 7). It is not necessary since most of these commands require only initial selection period to be executed.

No-Op, Diagnostic, and Mode Set Commands

- Any command byte with bus out 5, 6, 7 equal to 011, with the exception of request TIE, will set the no-op command latch.
- No-op command latch will:
  1. Force channel end and device end in status byte at completion of initial selection.
  2. Reset the command register.
- If channel responds to "status in" with "command out," no-op will set channel end trigger to resubmit status byte via "request in."
- No-op performs no operation in the tape unit or control unit.
- The diagnostic command sets the diagnostic trigger.
- Mode set bits 0, 1, 2, 3, and 4 enter mode register 0, 1, 2, 3, and 4.
- The no-op command sequence is also used during mode set and diagnostic.

The no-op command does not transmit data or cause tape motion. A no op performs no function in the tape control unit. No op can be used as a delay or it can be used to reserve a space in computer program. At a later time another command can replace the no op without disturbing the locations of other commands. If a command is deleted from a program, a no op can fill the space.

Whenever a non-motion control command is received during initial selection, a line called "mode or no op" is generated (top of Figure 18). The "mode or no op" line will be active for commands that have 6

and 7 bits and no 5 bit. This includes mode set, no op, request TIE and diagnostic (see Figure 101).

The no-op command trigger turns on only if the command is not a request TIE (3 and 4 bit, not 2 bit). The no-op command trigger will turn on for no op, mode set, and diagnostic. These three commands require only initial selection time but request TIE must transmit a byte from channel to the tape control after initial selection.

During initial selection the no-op command trigger blocks the no-op request TIE trigger, (Figure 301, coordinates 3E), and sets the "no-op reset command register" trigger, 4D. The no-op command trigger also forces channel end and device end (bits 4 and 5) in the initial selection status byte (coordinates 4D and 5D of Figure 302).

The no-op command trigger is reset by the same line that resets the command register. When the channel has accepted the initial selection status byte, the "no-op reset command register" trigger is reset (Figure 301, coordinate 4D). The tape control has then completed initial selection, transmitted channel end and device end, reset the command register, and reset the no-op command trigger. No tape motion or data transmission took place. All of these actions also occur on the diagnostic and mode set commands as shown on Figure 515.

Mode Set

- **Mode set command sets one of 15 different settings into mode register for seven-track operations.**
- **If command byte contains incorrect parity, mode set is not executed.**
- **A mode set command indicating "set data convert" on will generate a unit check via "command reject" if convert feature is not installed.**

The tape control mode register controls four operating conditions or modes for seven-track operation; bit density 200/556/800; data convert on/off; redundancy odd/even; and translate on/off. Operating mode is specified by the mode set command that contains modifier bits to set appropriate triggers in the mode register (Figure 18). Once the mode register is set, it will remain in the same status until another mode set is given or the control unit is reset. All succeeding seven-track operations are performed under the conditions set up by the last mode set. Nine-track operations are not subject to conditions specified by the mode register.

During initial selection, if the command is a mode set, a line called "set mode register" will gate bus out lines 0, 1, 2, 3, and 4 into mode register 0, 1, 2, 3, and 4 (Figure 18). Figure 515 is a flow chart of mode set.

The no-op command trigger (Figure 301, coordinates 4D) turns on during command out of the mode set initial selection. The no-op command trigger forces channel end and device end (bits 4 and 5) into the initial selection status byte (Figure 302, coordinates 4D and 5D). When channel end and device end are accepted in the initial selection status byte, the mode set operation is completed.

Diagnostic Mode

- **Diagnostic command will set diagnostic latch at "command out" time of initial selection period *if "diagnostic" jack on* CE *panel is plugged.***
- **With diagnostic latch set, R/W VRC circuits cannot activate P bit output to the TU write bus.**
- **Diagnostic latch is reset when TCU executes a "mode set" command *or* diagnostic jack on CE panel is unplugged.**

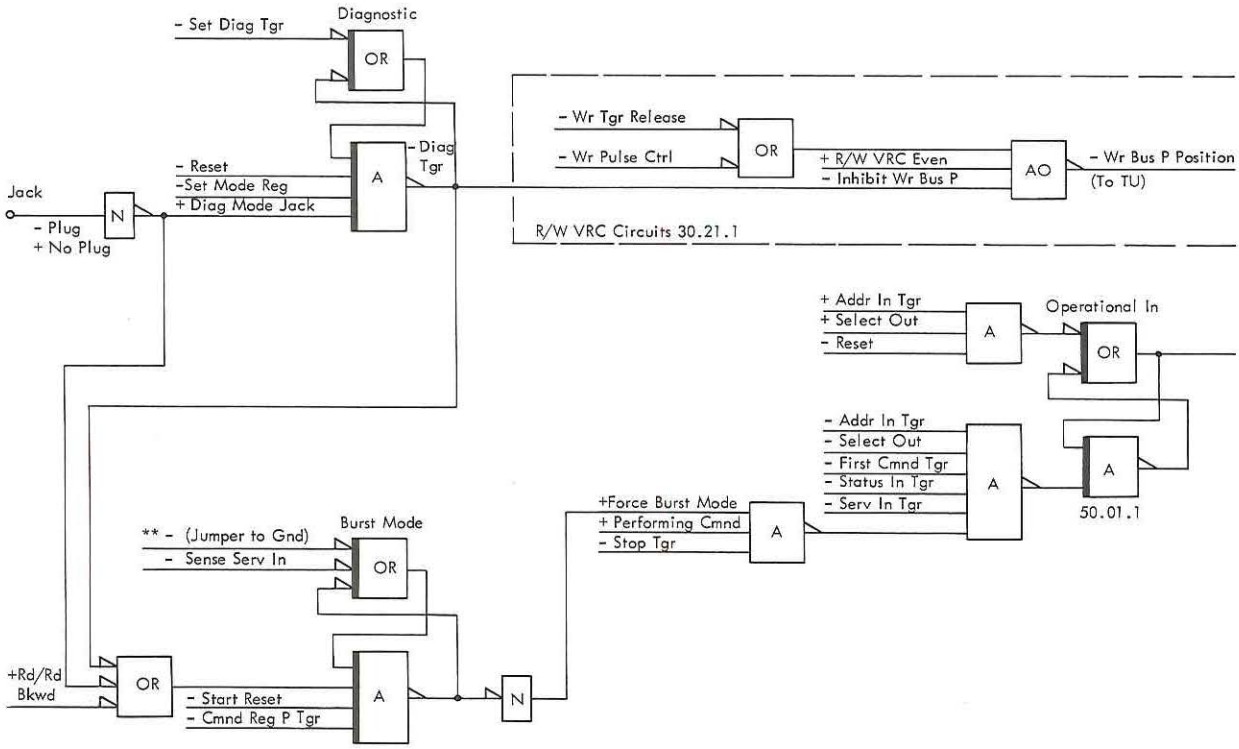
A diagnostic mode command sets the diagnostic trigger if the CE panel plug is in place (Figure 9). Diagnostic trigger blocks the write bus P line so that error bytes or blank bytes can be written for testing as shown on Figure 31. Diagnostic mode is intended for use with the CE panel and/or diagnostic programs to analyze error detection circuits as explained in the Appendix under "Diagnostic Mode."

On top of Figure 18, the "mode or no op" line will activate the "set diagnostic trigger" line during command out time of a diagnostic command. Notice the function of this line on Figure 31. If the diagnostic jack is plugged on the CE panel, the diagnostic trigger is set. Thus, on any following write commands, no P (parity) bit will be generated from the R/W register VRC to the write bus lines feeding the tape unit. Notice that the diagnostic trigger is reset if the jack is unplugged or if a mode set command is executed. Therefore, a programming stipulation is that a mode set command should not follow a diagnostic command before executing a write command.

In the flow chart on Figure 515 and the I/O diagram on Figure 301, as in the no-op and mode set commands, the no-op command trigger is turned on during initial selection command out. The no-op command trigger forces channel end and device end into the initial selection status byte to complete the operation.

Request TIE

- **Objectives of a TIE command are:**
  1. **Request one byte of data from channel (SI and SO) and set this byte in the EPR register.**
  2. **Set the "correct trigger" if the data byte does not contain a 6 and 7 bit.**



** Gnd Jumper	Diagnostic Tgr	Diagnostic Jack	Burst Mode Latch
Not Installed	N/A	N/A	Only set for sense command; reset for remainder of commands.
Installed	Off	Not Plugged	Set for ALL commands EXCEPT motion control commands.
Installed	Off	Plugged	Set for ALL commands EXCEPT motion ctrl commands and rd/rd bkwd command.
Installed	On	Plugged	Set for ALL commands EXCEPT motion control commands.

● Figure 31. Diagnostic and Burst Mode

3. **Provide an end status byte with channel end and device end bits by generating an end pulse.**

- **Request TIE uses the same circuits as no-op and mode set.**
- **Request TIE is used only when programming for error correction.**
- **Request TIE command must be issued to a nine-track tape unit address or the command will be executed like a no-op command.**

A request TIE command is used in the error correction programming sequence. When an error block is about to be corrected, the TIE (track in error) information must be sent to the tape control. The request TIE command transmits one byte to the tape control EPR. The byte should contain one bit indicating which track of the previous read operation was in error. Figure 514 is a flow chart of request TIE.

A request TIE command generates a line called mode or no op (Figure 301, coordinates 3E) just as in other non-motion control commands. The no-op command trigger will not be turned on because the TIE command contains bits 3 and 4 but not bit 2 and conditions the AND circuit to block its setting. Since the no-op command trigger remains off, the no-op request TIE trigger is now allowed to turn on.

Because of the AND circuit which blocked the set to "no-op command" latch (Figure 301, coordinates 3E) channel *cannot* address this command to a seven-track tape unit. Otherwise the no-op command latch will be allowed to set and inhibit setting of the no-op request TIE latch.

The no-op request trigger remains on after initial selection. During a no-op command the request trigger is reset by the no-op command trigger before its output can be used. In a request TIE operation the request trigger remains on to generate gating lines.

The purpose of a request TIE command is to supply a TIE (track in error) byte to the tape control EPR (error pattern register). To accomplish this, a request for service must be generated and one byte must be transmitted from channel. In this respect, a request TIE command is similar to a one byte write operation. During the command out portion of initial selection, unit check sample turns on the sense service trigger. This

is shown in Figure 302, coordinates 3C. For most commands the sense service trigger is held reset. In a request operation the sense service trigger generates service in until service in and service out, when the TIE byte is transmitted from channel. The sense service trigger is eventually reset by "set correct trigger" line which becomes active by service in and service out.

The service in line to the data channel requests a byte of data. The data channel responds with the byte on the bus out and a service out tag line. In Figure 301, coordinates 4E, service in and service out generate "set correct trigger." The "set correct" line:

1. resets the sense service trigger,
2. sets the correcting trigger on (Figure 404),
3. gates the byte on the bus out lines into the EPR (bottom of Figure 404).

Only one byte is needed, so the sense service trigger remains off. When the channel drops service out, the tape control delays for about one-half microsecond then generates an end pulse. End pulse in turn initiates an end status sequence to complete the operation.

One byte has been transmitted to the EPR and the correcting trigger turned on for error correction. Figure 404 shows the correcting trigger for error correction operation. The following operation must be a correction read or read backward. The correcting trigger is reset by any other command. The objective is to have the correcting trigger set while performing a read or read backward command. It reminds these commands to correct any errors which may develop. Reset conditions to the correcting trigger are:

1. EPR cannot contain a 6 and 7 bit.
2. Next command cannot be a seven-track operation.
3. Next command must be a read or read backward.

Test I/O

- **Test I/O is a means of checking to see if an I/O device, such as a tape unit, is available to perform an operation.**
- **Test I/O relieves the control unit of outstanding status if the test I/O is addressed to the same tape unit that produced the status.**
- **Test I/O requires no execution time. It is completed during initial selection.**
- **Unlike any other command, test I/O *will not* cause a busy bit (3) to be set in the status byte when clearing a previously rejected status byte.**
- **Test I/O will set the busy bit (3) in the status byte only if it is the present status and the tape unit is rewinding.**

The primary purpose of test I/O is to see if a particular I/O device is available to perform an operation. If the control unit is busy, a status byte with busy (bit 3) and status modifier (bit 1) will be returned to the channel. This occurs for other operations as well as test I/O because the status in is forced before channel sends the command. Figure 300 and the text of Initial Selection explain the control unit busy sequence. If the control unit is not busy, initial selection can proceed. When command out rises, the bus out lines are checked to see if the command is test I/O. All commands except test I/O will have at least one bit in positions 5, 6, or 7. If bits 5, 6, and 7 are blank, the command must be a test I/O. See Figure 101.

In Figure 301, coordinates 2E and 3E, the bus out lines are checked. The "no test I/O" trigger is turned on for every command except test I/O. At the end of initial selection the no test I/O trigger is reset.

Output of the no test I/O trigger is used in Figure 302, coordinates 1D. This line is active for every command except test I/O. During test I/O, "no test I/O" inactive blocks busy indications (bit 3) from being included in any outstanding status byte. The original contents of the previously rejected status byte are transferred to the channel. Busy (bit 3) can only be activated by the tape unit status – "ready and rewinding."

The "ready and rewinding or switched" line, on Figure 302, coordinates 1D, indicates the status of the selected tape unit. This line will be active if the selected tape unit indicates status A and B. Tape unit status A and B indicates that the tape unit is ready but busy performing a rewind operation. Status A and B can also be active if the tape unit is ready but switched to another control unit by a 2816 Switching Unit. If the tape unit is ready and rewinding or switched, a busy status (bit 3) will be returned to channel in the initial selection status byte.

The busy status line is inverted to block unit check sample, 1C. If the busy status line is down, unit check sample can occur. As the name implies, unit check sample samples the status A line from the tape unit to see if unit check should be turned on. If the status A line is down, the tape unit is not ready. In this case a unit check, bit 6, is returned in the initial selection status byte. When the tape unit is either not ready or nonexistent, unit check will be generated in answer to test I/O.

A test I/O, when the tape unit is ready and not busy, will generate a blank status byte. (Figure 302, coordinates 5C and 5D.) Busy (bit 3) cannot be brought up because the tape unit does not have status B. Bits 4 and 5 do not apply in this case. Bit 6 will not be brought up because tape unit status A blocks unit check. A blank status byte (except for a parity bit) indicates that the I/O path to a device is clear and the device is available.

At the end of initial selection the no test I/O trigger is reset. Since it is already off for test I/O, no action occurs. Test I/O is completed at the end of initial selection.

Sense

- **Six sense bytes are transferred to the channel via bus in lines, although only the first five bytes contain information.**
- **The sense command can be issued to a not ready or nonexistent tape unit and the command will be executed.**
- **The status of the selected TU is degated from setting unit check during initial selection.**
- **Sense byte 1 contains all the conditions to a unit check.**
- **Programmer would issue a sense command following an end status byte that contained a unit check.**
- **End status byte of a sense operation contains channel end and device end.**

In a sense operation, tape control transfers up to six sense bytes across the interface lines to channel. Bit positions in the sense bytes represent conditions in various tape control circuits. During a sense command the tape control does not communicate with any tape units. Information for the sense bytes is gated to the bus in lines from circuits in the control unit. Figure 8 lists the sense byte positions and the conditions they represent. Figure 513 is the flow chart for the sense command.

During initial selection, unit check sample attempts to set the "sense service in" trigger (Figure 303, coordinates 2A). For most commands it is held reset but when a sense command is decoded, sense service in is allowed to turn on. In addition, unit check sample is inhibited from interrogating the status of the selected tape unit (Figure 302, coordinates 1C). The sense command can be executed even if a "not ready" or nonexistent TU is addressed.

Near the end of initial selection the control unit sends a status byte to the channel. Channel responds with service out. The rise of service out steps the sense byte counter to 1 (Figure 303, coordinates 3C). Each service out response in a sense operation will step the three-stage counter.

After initial selection, "sense service in" turns on the service in trigger (3A) and the force burst mode trigger (4A). Force burst mode is used when the tape control is attached to a multiplex channel. Sense byte information is available in the control unit so there is no need to transfer data slowly in multiplex mode.

The service in trigger activates a circuit to generate a service in line to channel (5A). At the same time the sense byte counter sends a sense byte to the bus in gating circuits. Service in gates the byte onto the bus in lines (5D, E and F). Bytes transferred to channel must have odd parity. The bus in parity generator (5C) inserts P bits to maintain an odd count in each byte.

After the byte has been accepted, the channel responds with a service out or a command out. A command out reply turns on the stop trigger (Figure 303, coordinates 2A) to initiate action to end the operation by resetting the "sense service" trigger. A service out reply indicates that the channel is ready for the next byte.

Service out steps the sense byte counter. Service out delayed resets the service in trigger (3A). If the counter has not reached 7 (all three triggers on), tape control gates the next sense byte to the bus in circuits and turns on service in when service out drops. The service in/service out sequence continues until channel responds with command out or the sense byte counter reaches 7.

When channel sends command out, the stop trigger is turned on, which, in turn, resets the sense service trigger. Sense service trigger off gates an end pulse (3B) to initiate a normal end status sequence. If the sense byte counter is at 7 during a service out, the sense service trigger is reset which gates an end pulse and initiates an end status sequence.

A sense command can transmit up to six bytes of information to the channel. There are no circuits to provide bits for the sixth byte so it remains blank except for a parity bit.

The end pulse will reset the control unit and set the channel end latch to initiate an end status byte. This byte will contain channel end and device end bits.

TC-Channel Burst/Multiplex Modes

- **Tape control units can operate in either burst or multiplex mode.**
- **TC is "locked" to the channel whenever "operational in" line is active.**
- **TC and channel are always "locked" to each other during initial selection period regardless of command to be executed.**

Many System/360 I/O control units, such as the 2803 or 2804 Tape Control (TC), can operate in either burst mode or multiplex mode. In burst mode, the channel is committed to an I/O operation until the control (TC) transmits a status byte containing "channel end" status. In multiplex mode, the channel can disconnect from

the I/O operation after the initial selection sequence, regardless of the operation designated.

During multiplex operation, the tape control must interrupt the channel and identify itself prior to each byte transfer.

Most of the larger systems utilize "selector" type channels which operate in burst mode operation by holding up "select out" after an operation has begun. Many systems, however, use "multiplex" type channels that can operate in either multiplex or burst mode. A control unit can force burst mode on a multiplex channel if the "force burst mode" latch is on. Burst mode is forced by a control unit holding up the "operational in" line. A complete description of "force burst mode" is available in a later section.

The key to remaining "locked" to a channel is the "operational in" line. Once this line is active, the channel already knows which control unit it is communicating with. There is no need for the control unit to identify itself when data or status is transferred with the "operational in" line active. In Figure 31, the "operational in" latch is set whenever channel activates "select out" and the address in trigger of the control unit is active. This condition occurs when both channel and TC are available to communicate and either one is attempting selection of the other. During initial selection, it is the channel that is seeking TC; however, during multiplex entry, it is TC seeking to communicate with the channel.

In the resets to the "operational in" latch in Figure 31, there are many conditions to be considered. Each plays an important role as to "when" to allow reset. All are dependent on timing conditions to allow reset, however. Some of the more significant lines are:

1. "Not select out": The channel has received a channel end bit which implies that it is freed of the control unit. On multiplex channel it normally drops select out. Channel drops select out to disconnect.

2. "Not Status In": The TC has already submitted its status byte regardless of whether channel has accepted it. "Stack" will handle the resubmission via multiplex entry if the channel did reject the first transfer of the status byte.

3. "Not service in": TC did request entry for a data transfer. Service in is reset by either service out (continue data transfers) or command out (stop data transfers).

4. "Not force burst mode": TC is not forcing channel to operate in burst mode. If force burst mode remained active, TC would remain active provided it did not receive a signal to stop (regardless of whether channel dropped select out).

5. "Not performing a command": TC has its command register reset because it has completed the command. Most commands generate an end reset at the end of the operation.

6. "Stop trigger active": Channel has indicated the termination of the data transfers or an error has occurred which can terminate the operation (i.e., "overrun").

The particular time to reset the operational in latch is determined by the type of command and the mode (burst or multiplex) being employed at the time. The next sections will attempt to relate each condition to its mode of operation. However, there are two categories of commands which will reset operational in at the exact time regardless of the channel and mode used. They are:

1. Motion control commands: At the completion of initial selection, operational in is reset because "force burst mode" is inactive and the channel will deactivate "select out." Command register "P" resets "force burst mode" latch and when channel responds to "status in," operational in is reset.

2. No op, diagnostic, mode set and test I/O: Each of these commands do not set the command register. Therefore, "performing command" line is inactive. Thus, at the end of initial selection, operational in is reset when the channel deactivates "select out."

Burst Mode Operation

- **During burst mode, TC and channel are "locked" to each other during entire command including end status time.**

If burst mode is forced either by the channel (normal for a selector channel) or the TC, a read, write, request TIE, or sense command communicates with channel until after the first end status in sequence. The end status byte includes both channel end and device end status bits.

Motion control commands that require only the tape control (TC) and a tape unit (TU) to complete, free the channel after initial selection is complete ("operational in" and "select out" both drop). The normal end status must now be submitted via a multiplex entry since channel and TC are no longer "locked in."

Remaining commands (no op, test I/O, diagnostic, mode set) are completed by the end of initial selection. Channel is freed at this time.

Data Service — Burst Mode

Since the channel and TC are "locked in" via the "operational in" line active, prior identification is not needed when communicating for service. On Figure 304, coordinates 2B and 2D, the "service in" line is primarily dependent upon the shift control circuits when the operational in line is active.

End Status — Burst Mode

At the end of a command operating in burst mode (all commands except motion control) an end status byte must be submitted after the last data byte has been exchanged. The exception to this is the read/read backward commands that have more data characters in the record than the channel desires. Then, the end status byte is sent after the entire record is read. When operating in burst mode, there is no need for TC to identify itself prior to sending status because "operational in" is still active. The commands (read, write, read backward, sense, request TIE) set the channel end latch via an end pulse and generate an end reset.

Using the flow chart on Figure 512, the channel end latch would initiate an end sequence by setting "first command" trigger instead of activating "status in." The circuits involved are illustrated on Figure 304, coordinates 2A, for channel end and 7B for setting the first command latch. Channel end can activate the first command latch immediately after channel response of command out or service out because the operational in trigger is active. The remainder of the flow chart on Figure 512 and the "status in" objectives on Figure 304 illustrate:

1. The generation of the status byte—objective 3,
2. The acceptance of the end status byte and termination of the operation—objective 4 or,
3. The rejection of the end status byte—objective 5.

If the end status byte is rejected and this is not a valid control command, "stack" is set to begin resubmission of the end status byte via multiplex interrupt as described in the next section.

Rejection of any status byte which sets "stack" resets TC, which, in turn, allows "operational in" to reset because the "performing command" line is inactive (Figure 31). Now, the status byte must be resubmitted via multiplex interrupt since TC and channel are no longer "locked in."

Multiplex Channel Operation

- **After initial selection in multiplex mode, TC and channel are "locked" to each other only during data transfer time and end status transfer time.**
- **"Multiplex interrupt" implies TC is requesting to communicate with channel for either data service or status entry.**
- **During multiplex interrupt, TC must first request entry and then, if channel permits, identify itself by an address byte *prior* to data transfer or status transfer.**
- **A priority of multiplex entries is assigned to each control unit attached to the channel.**

A multiplex channel can operate in either burst or multiplex mode. If burst mode is forced by the tape control unit, operation is exactly the same as described under "burst mode operation." Burst mode will most probably be used for the faster tape units, especially on the slower channels. Multiplex mode will be used, most likely, with the slower tape units, especially on faster channels.

Burst Mode

The force burst mode latch (Figure 31) prevents the select out line from resetting the "operational in" latch when TC is performing a command on a multiplex channel where burst mode is being forced.

Whenever the "stop trigger" is set, the "operational in" trigger will be allowed to reset at that time (when select out drops) rather than after the end status byte is sent to the channel.

Multiplex Mode

After initial selection is complete, the channel is free of the TC, in multiplex mode. After initial selection, channel drops "select out" and TC then drops "operational in" because the force burst mode latch is off.

A write operation initiates a "service" interrupt via request in for its first data byte as soon as the initial selection "service out" (delayed) falls. If channel sends the byte, the operation is begun and channel is freed; otherwise, the operation is terminated.

A read or write operation will cause a "request in" (interrupt) sequence for each data byte transfer (explained in the next section). A sense operation causes a "request in" sequence for the first sense byte only; burst mode is forced for the remainder of the sense bytes. Figure 31 illustrates how "sense service" trigger will force burst mode. This also applies to request TIE command.

In addition to "service interrupts," TC will also generate request in for "status interrupts" for all end status bytes when operating in multiplex mode in a multiplex channel.

Service Data Interrupt—Multiplex

In the multiplex mode of operation, the channel and TC are disconnected at the end of initial selection. The control unit has the command to be executed and will inform channel that it either requests a data byte (write and request TIE) or it has a data byte available for the channel (read, read backwards, or sense). In all cases, it is the TC that is seeking to select the channel. However, the channel may be operating with several control units at this time. Each must identify itself prior to the transfer of data. Figure 7B illustrates the sequence of interface lines during a data interrupt.

On Figure 511, the request for service begins at point 3. TC will activate "request in" if channel "suppress out" is inactive. Interrupt objective 4 on Figure 304 indicates the circuits used at this time. Channel will allow the "requesting" control unit to "lock in" via its operational in line by activating "select out." The "select out" line is propagated through all control units to establish a definite priority among all control units attached to that channel. The section entitled "Priority" explains how priority is established.

Following the interrupt objectives on Figure 304 and the flow chart on Figure 511, the following events occur:

1. Activate operational in: Channel is now communicating with one control unit.
2. The tape control unit identifies itself with an address byte and "address in" tag (similar to initial selection).
3. The channel indicates "proceed" with the "command out" line.
4. TC activates the "service in" tag line for the transfer of the data byte.
5. Channel activates "service out," "continue," or "command out" (stop data transfer).
6. Operational in is deactivated and channel disconnects ("not force burst mode" and "select out" dropped).

The tape control unit must continue to initiate all data interrupts in the same sequence as just described, provided channel does not indicate "stop" (command out). The only exception to this is the sense command that forces burst mode on the first data interrupt. This is illustrated in Figure 31 by the "sense service" trigger becoming active.

Status Interrupt—Multiplex

- **Motion control commands, with the exception of rewind-unload, submit their end status byte via a multiplex entry (request in) upon completion of the operation.**
- **Rewind-unload command submits end status byte via multiplex entry when TU begins to rewind.**
- **Multiplex entry involves TCU identification and "locking" into the channel prior to sending status byte.**
- **End sequence can be initiated by either channel end trigger or selected device end.**
- **If end status is not accepted (command out) then end status is resubmitted via multiplex entry regardless of the command that was completed.**

Any time the tape control unit must submit a status byte via multiplex entry (operational in inactive), it uses the "request in" circuits to connect to the channel. The following conditions indicate a need for transmission of a status byte to channel via this route:

1. End status byte for all motion control commands regardless of selector or multiplex channel (interrupt objective 3, Figure 304).
2. End status byte for read, write, read backwards, sense, and request TIE commands in multiplex mode (interrupt objective 2, Figure 304).
3. Submitting a previously rejected initial selection status byte regardless of type of channel or mode of operation (interrupt objective 1, Figure 304).

The sequence of interface lines utilized during a status interrupt sequence is illustrated in Figure 7C. In addition, the flow chart on Figure 511, point "F," lists all the actions as they occur. The actual status byte generation and submission will occur on the flow chart on Figure 512 when Figure 511 leaves via exit "T" to join Figure 512 at entrance "V." Interrupt objectives 1 through 3 describe the circuits utilized to initiate a "request in" for any of the above conditions that utilize the status interrupt. Interrupt objectives 5 through 9 are the same circuits used for the data interrupt which are primarily used to identify the TC and "lock" into the channel. "Status in" objectives 1 through 5 are used in conjunction with Figure 512 in describing status byte generation and submission.

The major differences between the data and status interrupts are only the conditions which initiate the interrupt and the activation of status in, rather than service in. Identical circuits are utilized to request interrupt, identify TC to channel, and lock into the channel.

During motion control commands a constant reset is held on the channel end latch (Figure 304, coordinates 2A); however, if the control immediate trigger is *off* or the control unit end latch is on, the channel end latch will be set at "end pulse" time. Control unit end will be set if a unit check or unit exception is encountered by the motion control command (Figure 304, coordinates 4G). Since "stack" cannot be set, channel end will now cause a "control unit busy" (short sequence) if the channel attempts to select the *same* TU and status (UC or UE) is outstanding (Figure 300, cordinates 3D).

The following outline shows the bits that are normally expected in the end status byte of each operation:

WRITE—One Interrupt
1. Channel End and Device End.
2. Channel End, Device End, and Unit Exception when End of Tape is encountered.

READ—One Interrupt
1. Channel End and Device End.
2. Channel End, Device End, and Unit Exception when End of File is encountered.

READ BACKWARDS—One Interrupt
1. Channel End and Device End.
2. Channel End, Device End, and Unit Exception when End of File is encountered.
3. Channel End, Device End, and Unit Check when into/at Load Point.

SENSE—One Interrupt
1. Channel End and Device End.

REWIND—One Interrupt
1. Device End.

REWIND-UNLOAD—One Interrupt
1. Control Unit End, Unit Check, and Device End.

WRITE END OF FILE—One Interrupt
1. Device End.
2. Device End and Unit Exception when End of Tape is encountered.

ERASE GAP—One Interrupt
1. Device End.
2. Device End and Unit Exception when End of Tape is encountered.

BACKSPACE FILE—One Interrupt
1. Device End.
2. Device End, Unit Check when into/at Load Point, and Control Unit End.

BACKSPACE RECORD—One Interrupt
1. Device End.
2. Device End, Unit Check when into/at Load Point and Control Unit End.
3. Device End, Control Unit End, and Unit Exception when End of File is encountered.

FORWARD SPACE FILE—One Interrupt
1. Device End.

FORWARD SPACE RECORD—One Interrupt
1. Device End.
2. Device End, Control Unit End, and Unit Exception when End of File is encountered.

MODE SET (Excluding Request TIE)—No Interrupt
1. Channel End and Device End on Initial CSW Stored.

REQUEST TIE—One Interrupt
1. Channel End and Device End.

"Suppress out" is a line raised by channel when it cannot service a "request in" from any CU (other than the one it is servicing currently). "Suppress out" will prevent TC from raising "request in."

If TC is not working with channel and a service data or status interrupt condition exists, "request in" is raised to channel and also sets the address in trigger (Figure 304, coordinates 5A). When a multiplex channel receives "request in," it raises "select out" to the first (highest priority) control unit. As long as no control unit with higher priority needs service, select out is propagated to the requesting TC (2803 or 2804).

Chain Trigger

Channel might want to make certain that a selected tape unit *remains* selected so the TU device end signal priority and/or successive commands can be directed to the TU without delay. A TU is "chained" by locking its address in the TU select address register; this is done by preventing the reset of the TU select trigger, which prevents the device end scanner (TU register) from cycling.

During any "status in" sequence, channel must condition both "service out" and "suppress out" to set the chain trigger (Figure 302, coordinates 3F and 3G). The chain trigger, which blocks the reset of the TU select trigger, remains set as long as "suppress out" remains active; this line can remain active through a number of initial selection operations to prevent the reset of the chain trigger.

Control Unit Priority

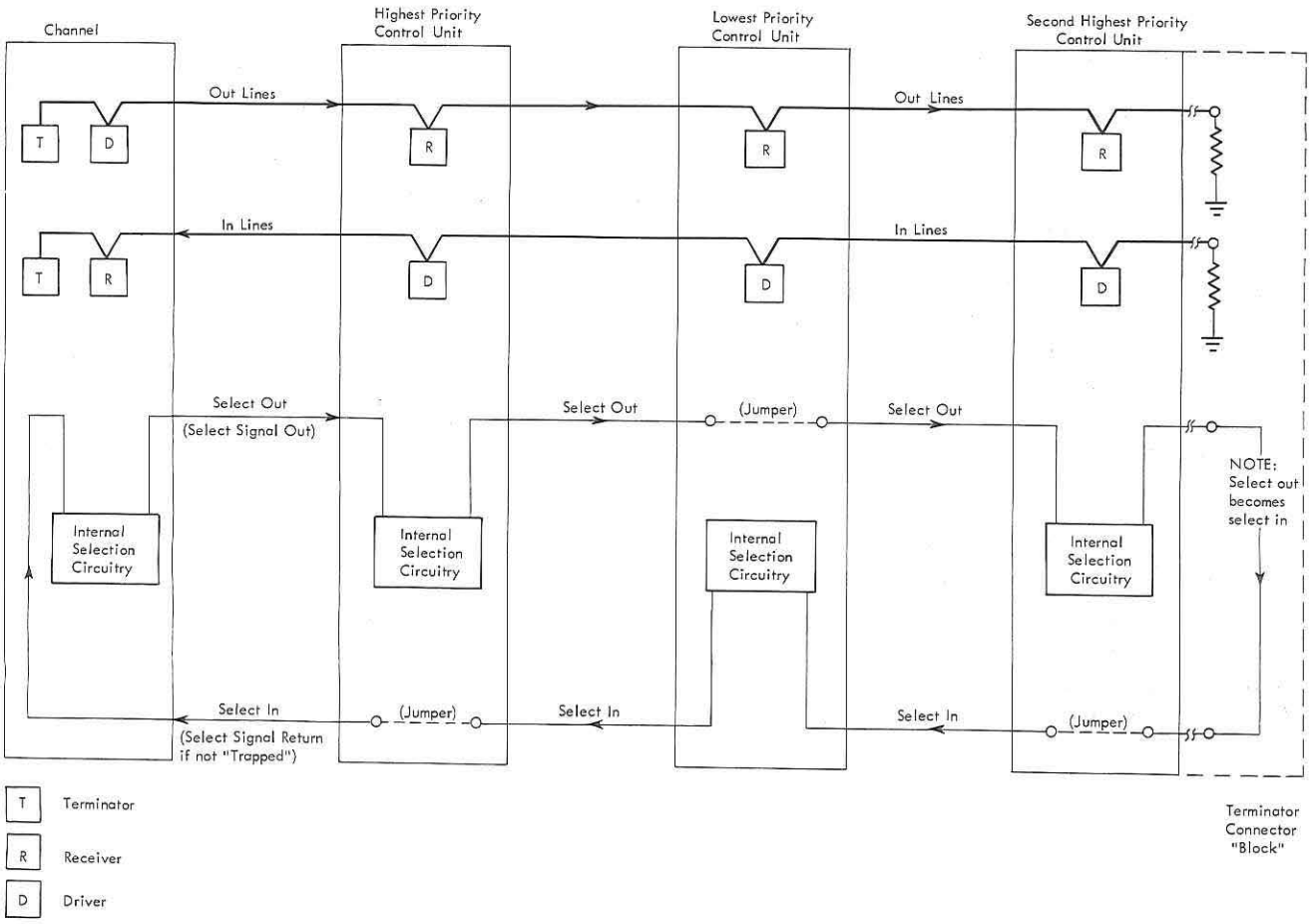
- **Control unit priority is determined by "internal select signal" wiring and physical connections to the interface signal cables.**
- **Interface priority is the order in which control units are selected if more than one unit requires channel service.**
- **The select signal leaves the channel as "select out," but may reach some control units as "select in."**
- **When the select signal reaches a control unit for which the signal is *not* intended, the signal must be propagated to the next lower priority control unit.**

Priority of a control unit is determined by the wiring of "select out" and "select in" signals. A select out signal from the channel proceeds through each control unit to the last control unit on line. The "select out" becomes "select in" and returns through each control unit to form a complete loop back to the channel. Figure 32 illustrates the "select" cable sequence.

The control unit which first receives the select out in its logic circuits has the highest priority. If the control unit does not need a "select," it must propagate (pass along) the signal to the next control unit. If it does require service, the control unit retains the select out and does not send a "select in" to the next CU (Figure 300, coordinates 4B). If none of the control units needs "select" at this time, the select signal will return to the channel at the end of the loop – out on the "select out" lines and back on the "select in" lines.

The actual priority sequence may not be the same as the physical cable sequence. Select out can pass through a control unit without entering the logic circuits as illustrated by the wiring of the middle control unit on Figure 32. In this case, the control unit must wait until the "select" signal returns along the select in lines. The control unit logic circuits can be wired to accept the select signal from the "select out" or the "select in" lines according to the desired priority sequence.

The select out and select in lines form a *series* loop from and to the channel through all attached control units (Figure 32). The select out line is jumpered to the select in line in the terminator block located at



● Figure 32. Interface Connections

the "tail gate" of the control unit that is physically the farthest unit from the channel on the interface cable hookup. Electrical priority is established by wiring within each control unit when the system is first installed in a customer installation.

The higher priority control units are "wired" in series with the select out line from the channel; the lower priority units are wired in series with the select in return line to the channel. The *highest* priority unit is the unit which first receives the select (out) signal; the *lowest* priority unit is the unit that returns the select (in) signal directly to the channel.

If power is down in a control unit, the "select out" and "select in" signals will be passed on to the next control unit or back to channel whichever is next in the sequence. When power is down, the relay (Figure 300, coordinates 2A) is down. The select signals pass through the normally closed points and out to the next unit. When power is up, the relay picks and the select circuits can accept the select if it is needed. (Priority is not needed for initial selection of a command because the channel specifies which control unit should respond by sending "address out.") Only one unit can respond to initial selection.

"Select" sets the operational in trigger (Figure 304, coordinates 5B), which raises "operational in" to channel; the tape control is now ready to communicate with the channel. After approximately 0.7 microsecond delay, the tape control raises "address in" to channel. At the same time, tape control gates an address byte on the bus in interface lines to channel (Figure 304, coordinates 7B), to identify itself.

Force Burst Mode

- **"Force burst mode" latch set inhibits the "operational in" latch from being reset, thus allowing the TCU to remain locked to the channel.**
- **Sense command forces burst mode during data transfer when operating on multiplex channel.**
- **Burst mode can be forced on a multiplex channel for read, write, and read backward commands if:**
  1. **Burst mode latch is forced set by ground jumper and,**

2. TCU is performing the command without a "stop" signal (CO) from channel, and

3. The diagnostic jack is not plugged on CE panel.

When operating with a multiplex channel, the TC can operate in either multiplex mode or burst mode. The 2403/2404/2803/2804 tape control units possess the circuitry to force burst mode on a multiplex channel. Burst mode is indicated by retaining the active state of the operational in latch in TC.

As shown in Figure 31 the operational latch cannot be reset if:

1. Force burst mode latch is active,
2. TC is performing a command, and
3. Stop trigger is not set.

If a ground jumper is not installed on the set side of the latch, the force burst mode line will be active only when the "sense service" trigger is active. This implies that all sense bytes of the sense command are transferred to channel via a burst mode. However, the first sense byte must initiate a multiplex entry via request in first.

If the ground jumper is installed when TC is connected to a multiplex channel, various effects will depend on particular conditions:

1. All motions control commands are executed via multiplex mode because "command register P" resets the force burst mode (FBM) latch.

2. If the diagnostic jack is not plugged, all commands except motion control commands are executed in burst mode.

3. If the diagnostic jack is plugged and the diagnostic trigger is reset, all commands except motion control and read/read backward commands are executed in burst mode.

The reason to reset FBM when executing the read type commands when the diagnostic jack is plugged is to allow tape diagnostics to operate properly during IRG tests. The channel and TC must function in multiplex mode to allow CPU to run. CPU can now measure interrecord gaps by measuring time displacement between receipt of data bytes. Thus, although TC is attempting to force burst mode, the read type commands in conjunction with the diagnostic jack can revert tape control back to multiplex mode.

Power supply units used in the tape control units are the same type as those used in tape units. In the 2403 and 2404 configurations, the tape unit and control unit share a common power supply. For detailed information on power supply circuits refer to:

Field Engineering Theory of Operation, *IBM Magnetic Tape Units: 2401, 2402, 2403 Models 1-6; 2404 Models 1-3*, Form Y22-2819

Field Engineering Maintenance Manual, *IBM Magnetic Tape Units: 2401, 2402, 2403 Models 1-6; 2404 Models 1-3*, Form Y22-6631

Field Engineering Maintenance Manual, *IBM Tape Controls: 2403 Models 1-6, 2404 Models 1-3, 2803/2804 Models 1 and 2*, Form Y22-6635.

Features

Tape controls described in this manual can be equipped with features to extend the usefulness of tape operations. Five features are available:

FEATURE	NUMBER
Seven-track compatibility	7125
Date conversion	3228
Two-channel switch	8170
Remote switch attachment	6148
Sixteen-drive addressing	7185

Seven-track compatibility and data conversion are available on the 2403/2803 tape controls and the 2404/2804 simultaneous R/W tape controls. The remaining three features can be installed only on the 2403/2803 tape controls.

Seven-track compatibility involves nearly every circuit in the tape control, and the descriptions of seven-track operations are integrated into the text and diagrams throughout the manual. This section of the manual contains descriptions of the four remaining features.

Data Conversion Feature

- **Data conversion (DC) is an optional feature to allow processing binary seven-track information with maximum packing efficiency in System/360 storage.**
- **Data conversion always uses odd parity mode.**
- **During write operation, 3 eight-bit bytes are converted to 4 six-bit characters to be written on tape.**
- **During read operation, 4 six-bit characters from the tape unit are converted to 3 eight-bit bytes to be sent to the channel.**
- **Data conversion cannot be used for read backward operations.**
- **Position 2 of the mode register controls data conversion. Mode register 2 on = data convert off.**
- **Data conversion and translate mode cannot be used at the same time.**

Data conversion is an optional feature for a tape control that has the seven-track feature installed. When seven-track data is processed, six bits of each character are useful data. If these six-bit characters are placed in System/360 (eight-bit) storage, two bit positions are unused. Data conversion stores data in all (eight) bit positions to increase storage packing efficiency.

During write operation, 3 eight-bit bytes from the data channel are converted to 4 six-bit characters to be written on tape. Three eight-bit bytes have the same number of information bits (24) as 4 six-bit characters.

For read operation the conversion is reversed. Four six-bit characters (24 bits) from the tape unit are converted to 3 eight-bit bytes (24 bits) to be sent to the data channel.

Tape controls equipped with this feature can operate with the data converter on or off. When the data converter (DC) is off in seven-track operation, each six-bit character occupies an eight-bit byte location. The six-bit characters occupy positions 2 to 7 on the interface bus lines and positions 7 to 12 in the data register.

Data conversion cannot be used in read backward operation. If the characters are read in reverse order, the bits cannot be placed in the proper byte sequence on the bus in lines. Data conversion is automatically turned off during read backward.

During seven-track operation, bit 2 of the mode register controls the data converter. If mode register 2 is on, the converter is turned off. When the mode register is reset, a seven-track read or write (not backward) will be processed in DC-on mode.

Data convert cannot be used with even redundancy or translate mode. If a mode set attempts to turn on even redundancy or translate and the data converter (no bit 2), it no longer is a mode set command. A command with 5 off, 6 and 7 bits on (control), and bits 2 and 3 off, becomes a no-op or diagnostic command. See command format in Figure 101. A mode set attempting to turn on the data converter in a seven-track tape control that has no converter will be command rejected.

DC Functional Units

The heart of data conversion is the byte counter. It is a three-stage cyclic counter that generates gating levels to process data. Outputs of the three triggers are decoded to produce the gating levels.

Shift data pulses step the counter in synchronism with data transfer through the tape control. Each shift pulse steps the counter twice. A complete byte counter cycle of eight trigger settings requires only

four shift data pulses. Each shift data pulse represents one six-bit character processed to or from a seven-track tape unit (four in each byte counter cycle).

Byte counter positions and corresponding output gates are shown in the chart on Figure 408.

Each time the shift data line rises, the A trigger is complemented. The B trigger is complemented when the A trigger is on and the shift pulse falls. The C trigger is complemented when the A trigger is off and the shift pulse falls. Only one trigger at a time changes state.

The byte counter is reset at the beginning of each operation by a "start reset." If an overrun condition occurs, the byte counter is reset by the "set overrun trigger."

In nine-track operation the "data convert" line from the mode register remains down to keep shift pulses from stepping the byte counter. After start reset of a nine-track operation, the byte counter remains reset.

Output gates of the byte counter are used in nine-track as well as seven-track operation. When the byte counter is at zero during a nine-track read or write, the output gates used are:

1. Read shift byte 2 or 4.
2. Read byte 4.
3. Write shift byte 2 or 4.
4. Write set byte 3.

These gates establish a standard "straight through" data path because the data converter is automatically turned off during nine-track operation. During seven-track data convert operations, the byte counter gates are used to gate data into and out of the data register. Figures 309 and 310 illustrate data flow with the converter on.

DC Theory of Operation

Write Data Convert

With DC on, tape control converts 3 *eight-bit* data bytes from channel to *4 six-bit* characters, and transmits one six-bit character to the tape unit during each write clock cycle. The three-stage byte counter gentrates input and output gates for the data register. The first input byte to the data register sets positions 1 through 8; the second input byte sets positions 1 through 4, and 9 through 12; the third input byte sets positions 5 through 12.

The first and third output characters from the data register are taken from positions 1 through 6; the second and fourth output characters from the data register are unloaded from positions 7 through 12. Each six-bit character transferred from the data register sets R/W register positions 2 through 7. R/W register positions P, 0, and 1 are not used in seven-track write operations.

Tape control does not request a data byte from channel after transferring the third character from the data register to the R/W register. The byte counter advances after each character is unloaded from the data register and resets after the *fourth* data register character sets the R/W register. Therefore, the states of the byte counter triggers indicate data register positions to be loaded and unloaded and initiate a "service in" request to channel when a data transfer from channel is required.

The first "service in" of a write operation is generated at the end of initial selection. After initial selection, the "tape demand" line from the byte counter generates "service in."

The following example illustrates byte counter and data register actions in a write operation.

The byte counter supplies three input gates ("write set byte 1," "write set byte 2," and "write set byte 3") to designate data register positions for input interface data, and two output gates ("write shift byte 1 or 3" and "write shift byte 2 or 4") to specify data register positions to be unloaded. When the interface transfers the first byte to the data register, "write set byte 1" is active, causing the byte to be stored in data register positions 1 through 8. When the tape control brings up "write shift byte 1 or 3," bits in data register positions 1 through 6 transfer to the read-write register; the tape control requests another byte from the interface.

"Write set byte 2" is active when the interface sends the second byte to the data register; the byte is loaded in data register positions 1 through 4 and 9 through 12. "Write shift byte 2 or 4" allows bits in positions 7 through 12 to transfer to the read/write register, and the tape control signals the interface for a third byte.

"Write set byte 3" loads the third interface byte in data register positions 5 through 12; at this time, all data register positions are filled. "Write shift byte 1 or 3" is conditioned for the second time to transfer bits in data register positions 1 through 6 to the read/write register. The tape control does not request another byte from the interface because data register positions 7 through 12 are loaded. "Write shift byte 2 or 4" is conditioned for the second time to transfer bits in data register positions 7 through 12 to the read/write register; the data register is now empty. For every three input interface bytes, the data register and byte counter repeat the operation.

If the channel responds with "command out" instead of "service out," the operation will be terminated. "Service in" and "command out" can turn on the "stop trigger" regardless of the byte counter setting. A "command out" response to the first service

in (initial selection) will terminate the operation before write delay by generating a "word count zero" signal. After initial selection, command out (stop) depends on byte counter setting and shift pulse to terminate the write operation. If the stop trigger is on when the byte counter A and B triggers are off, data transfer will stop and write condition will be reset. When either byte counter trigger A or B is on (conversion cycle), data transfer will stop after the next shift data pulse.

The additional shift pulse is needed to process any bits remaining in the data register. If the channel has transmitted one eight-bit byte to the data register, 2 six-bit characters must be written on the tape. The last four bits of the second character will be blank. Two eight-bit bytes require 3 six-bit characters. The remaining two bits of the third character will be blank. Three eight-bit bytes are written as 4 six-bit characters.

The "stop data transfer" trigger blocks shift pulses and resets "write condition" at WC-14-15. When write condition resets, the normal write operation ending sequence begins.

During write data conversion the data register reset pulses follow the shift data pulses. After a character is shifted (into the read/write register), the data register positions it occupied are reset. Write shift byte 1 and 3 are followed by a reset to data register 1-6. Write shift byte 2 and 4 are followed by a reset to data register 7-12.

The read/write register is reset each write clock 2 time. WC-2 occurs just before the shift data pulse.

Read Data Convert

In data conversion read operations, the tape control converts 4 six-bit characters to 3 eight-bit code bytes. The three-stage byte counter controls the conversion by generating input and output gates for the data register. The first and third input characters to the data register set positions 1 through 6; the second and fourth input characters set positions 7 through 12. Therefore, bits stored in R/W register position 7 set data register position 12 or 6, depending on the status of the byte counter. Read/write register positions 0 and 1 are not used in conversion operations, and R/W register position P output does not set a data register position. Figure 310 illustrates data flow for a read conversion operation.

In DC-on mode operation, the byte counter controls the "tape demand" line. "Tape demand" is conditioned only when the byte counter indicates that at least eight data register positions are loaded. For every four input six-bit characters to the data register, the byte counter conditions "tape demand" three times (after the second, third, and fourth six-bit characters are stored in data register positions). The byte counter causes tape control to unload the first output byte from data register positions 1 through 8; the second byte from positions 1 through 4, and 9 through 12; the third byte from positions 5 through 12.

The following example illustrates byte counter and data register actions in a read operation. For this case assume that 4 six-bit characters are read from the tape and converted to 3 eight-bit bytes for channel.

The byte counter supplies two data register input gates from the R/W register, "read shift byte 1 or 3," and "read shift byte 2 or 4." The byte counter also supplies three gates to place data register bits on the bus in lines, read byte 2, read byte 3, and read byte 4.

Characters are gated into the R/W register by a read clock 7 pulse. Read shift byte 1 or 3 gates R/W register 2-7 into data register 1-6. At this time there are not enough bits in the data register to construct a complete eight-bit byte so there is no output to the bus in lines.

The second character (in R/W register 2-7) is gated into data register 7-12 by read shift byte 2 or 4. The data register now is filled. After the shift data pulse falls, read byte 2 gates data register 1-8 to the bus in lines 0-7. Tape demand generates a "service in" to tell the channel that a data byte is on the bus in lines. Data register positions 9-12 are held for the next byte.

The third character from tape is shifted from the R/W register into data register 1-6 (same positions as the first character). The data register now contains data in positions 1-6 and 9-12. Read byte 3 gates positions 1-4 and 9-12 to bus in positions 0-7. Note that data register positions 5 and 6 are retained for the next byte. Service in is brought up and the second byte is sent to the data channel.

The fourth character is shifted from the R/W register into data register 7-12. Positions 5-12 of the data register now contain the third byte to be sent to channel. Tape demand, service in, and read byte 4 will transmit the third and last byte to the data channel.

Read operations can be terminated either by the tape control sensing the end of data or the channel refusing to accept more data. If the channel responds to "service in" with "command out," the stop trigger turns on and blocks service in. No more data will be transmitted to the channel; however, the tape control continues reading until the end of block is sensed. When the tape control ends the operation, the status of information in process will determine if an additional shift pulse is needed. If bits remain in the data register after the last character has been read, one more data shift is required to transfer these bits to the channel. The "left-over" bits will not form a

complete eight-bit byte. The unused positions of the byte will be blank on the bus in lines.

Seven-track tapes read in data convert mode may have been generated on a System/360 or on a different system. Because of differences in data format, tapes from other systems can have any number of characters in a block and any combination of bits in each character. If a tape from another system contains blocks that are not multiples of four characters, System/360 tape control units may not have a full eight-bit byte to send to the channel when the block ends. When the left-over portion of a byte is transmitted to the channel, the remainder of the byte contains zero bits that are generated by the tape control. This type of operation is called "padding." The data convert check indicator is turned on to indicate that padding has occurred and that the last byte transmitted is incomplete.

It is possible that the last six-bit character from tape has zeros in the portion "left over." In this case, since no active bits remain in the tape control, no "extra" byte is transmitted and data convert check is not signaled because padding of the last byte did not actually occur.

Seven-track tapes that are generated by a System/360 in convert mode cannot have a combination of characters which will produce a data convert check. In convert mode, the characters on tape are generated from eight-bit bytes from storage. No matter how many bytes are written in convert mode, the characters on tape cannot cause a data convert check when read. The portion of the last tape character that would be left over on a read operation will automatically be blank. Zeros added in the write operation will be ignored on the read operation.

System/360 can generate tapes that will cause data convert check if the tapes are written with data convert off. Without data convert, the last character on tape can have any combination of bits and, therefore, may cause a data convert check when read.

When a block read from tape does not contain a multiple of four characters, there may be one, two, or three characters processed in the last byte counter cycle. If one six-bit character remains for the last byte counter cycle, shift data and read byte 2 pulses will be generated at RDD-36 time. The DCC (data convert check) trigger will be turned on to indicate that the last byte sent to the data channel was "padded" with blanks. Bus in lines 6 and 7 will be the blank positions when one six-bit character is processed.

When two characters are left over, the first eight-bit byte is sent to channel in the normal manner. The last four bits of the second character are in data register 9-12. At RDD-36 time the tape control has determined that there is no more data. If there are any bits on in data register 9-12, they must be processed. If 9-12 are blank, the operation can be terminated.

If data register 9-12 contains bits, a data convert check and shift pulse will be generated to transmit the last byte to channel. A read byte 3 pulse gates data register 1-4 (blank) and 9-12 to the bus in lines. Blanks in data register 1-4 are gated to bus in lines 4-7.

In the case where 3 six-bit characters are left over, 2 eight-bit bytes are sent to channel in the normal manner. The remaining bits of the third character are in data register 5 and 6. The tape control examines positions 5 and 6 to see if either contains a one bit.

If data register 5 and 6 are blank, the operation terminates. If 5 or 6 contains a bit, then a data convert check is signaled and a shift data pulse is generated. Tape demand and read byte 4 transmit the last byte to channel. Data registers 5 and 6 are gated to bus in lines 0 and 1. Bus in lines 2-7 will be blank.

The R/W register is reset by an SS2 pulse which follows each data shift. The data register must retain bits from one byte to the next; therefore, the whole data register cannot be reset each time. During initial selection, the entire data register is reset but after initial selection the positions reset depend on byte counter status.

At the end of read byte 2, data register positions 1-6 are reset. At the end of read byte 3, data register positions 7-12 are reset. After read byte 4, all twelve data register positions are reset. Data register resets depend on a "service out" signal from the data channel. Service out indicates that the byte on the bus in lines has been accepted and the data register need no longer retain it.

C-Compare Check

C-Compare check is a data check not an equipment check. Bytes transmitted through the tape control in a read or write operation should not change parity except during translate mode. C-Compare detects parity changes during data transmission through the control unit.

Figure 212 illustrates C-Compare circuits. During data convert operations, not all bits in a byte are transmitted at the same time. For this reason parity comparison cannot be performed for every byte.

When the byte counter reaches the end of a cycle, the total number of bits transmitted should be the same number received. The status of the C-Compare circuit is checked each time the byte counter reaches 000. If the data parity check trigger is on when the byte counter reaches 000, the C-Compare trigger is turned on. Note that translate mode blocks setting of the C-Compare trigger. This is done because the total number of bits in a byte may change during

translation; for this reason, C-Compare cannot be checked when the translator is operating. The translator cannot operate at the same time as data conversion. A mode set command attempting to turn on the translator and data conversion (4 bit and no 2 bit) is no longer a mode set command. Refer to Figure 101.

Two Channel Switch or MIS Feature

- **The multiple interface switching (MIS) tape control is a standard tape control unit modified to operate with two data channels and one group of tape units.**
- **The modifications add circuits so that the MIS tape control can perform standard operations as well as dual-channel system operations.**
- **The MIS tape control can operate with either of two channel interfaces.**
- **Two new commands called "reserve" and "release" are used to place interface switching under program control.**
- **Reserve and release functions are performed during initial selection time of sense commands.**
- **Interface reset circuits are interlocked to prevent interference from one channel while the other channel is operating.**
- **Two sets of device end triggers make sure that a selected device end signal is returned to the same channel that initiated the command.**

The multiple interface switching (MIS) tape control is a 2803 Tape Control modified to operate with two data channels instead of one. All of the 2803 operations can be performed on either channel interface. In addition to standard operations, the MIS tape control can perform reserve and release commands for program control of interface switching. Figure 33 illustrates the organization of MIS circuits.

The large block in the center of Figure 33 represents the basic tape control circuits without MIS. When the MIS feature is installed, an additional set of device end triggers is connected to the scanner in the tape control circuits.

In Figure 33, the smaller blocks are logical functions of the MIS feature. The physical location of the added circuitry is inside the tape control unit. Figure 33 represents only the *logical* separation of circuits.

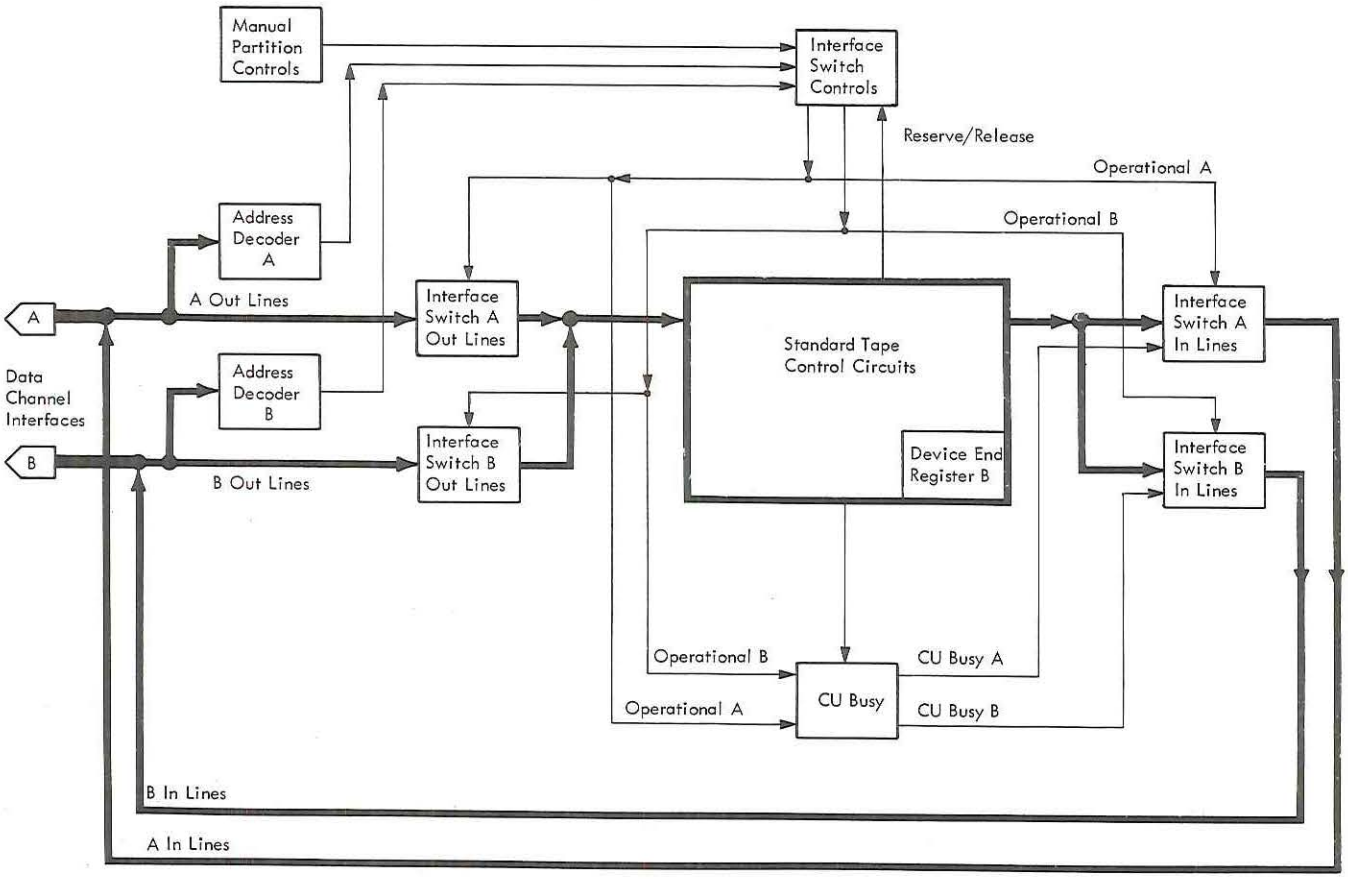


Figure 33. Organization of MIS Tape Control

Channel interface lines going in or out of the tape control must pass through "switch" circuits. The switch circuits consist of gated drivers to connect one or the other of the channels to the tape control but not both at the same time.

Interface switch control circuits supply the control lines to select the correct groups of drivers for operation with interface A or B. The switch control circuits are interlocked to prevent one channel from interfering with operations on the other. When neither interface is reserved or operating, the switch circuits are in a "neutral" state. When the switch circuits are in neutral, a select can be accepted from either interface.

Address decoder circuits monitor the bus out lines of each interface. If the address of the tape control unit appears on the bus out lines along with an address out tag, the decoders send a signal to the interface switch circuits. When no interfering conditions exist, the switch circuits connect that interface to the tape control. If the tape control is reserved or operating with the other interface, a busy signal is sent to the interface that attempted to break in.

When the tape control finishes operations and becomes available, a "control unit end" status byte is sent to the channel that previously received the busy signal.

Channels attached to the MIS interfaces can be either on the same system or on separate systems. One group of tape units on the MIS tape control can be shared between two systems or two channels on one system.

MIS switching circuits can be locked onto one interface so that the other interface does not have access to the control unit. The command that performs this operation is called "reserve." A "release" command resets the reserve condition and allows the control unit to accept commands from either interface. Both the reserve and release functions are performed as part of a sense command. Modifier bits in the sense command determine whether the command is a "reserve/sense" or a "release/sense." Reserve or release is performed during initial selection time of the sense operation.

When a tape unit completes an operation, a device end signal is sent to the channel. Because the MIS tape control is connected to two chanels, the device end circuits must be modified to be sure that the device end is returned to the channel that initiated the operation. This is accomplished by adding another group of device end triggers. Device end register A is armed by operations with the A interface. Device end register B is armed by interface B operations. Device end signals from the A group are returned to interface A; device end signals from the B group are returned to interface B. The A and B device end triggers represent the same group of tape units.

Resets

A general or selective reset from a data channel normally resets the tape control unit unconditionally. In MIS two-channel operation this could cause interference problems. The idle channel must not disrupt an operation by resetting the control unit. MIS reset circuits are interlocked so that a reset from one channel cannot disrupt operations on the other channel. A reset can be accepted only from the channel which is operating. Resets are further conditioned to prevent a channel from destroying information needed by the other channel. The A or B device end triggers cannot be reset by the opposite interface and an A or B CU end trigger cannot be reset by the other interface.

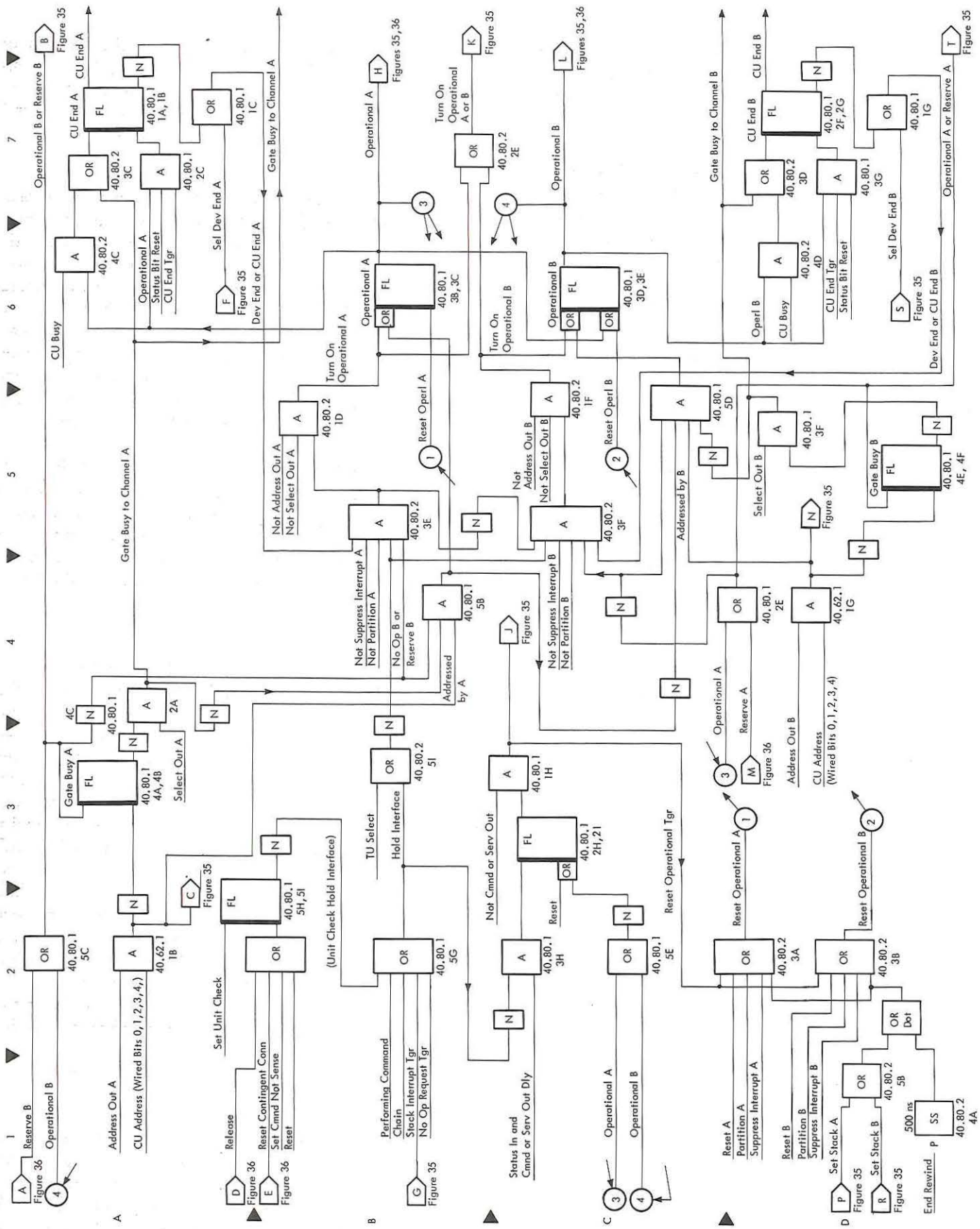
Interface Switch Control Circuit

- **The multiple interface switch (MIS) tape control must monitor addresses on two channel interfaces.**
- **When the tape control receives its own address, it attempts to begin operation with that interface.**
- **If not busy or reserved, the operational trigger for interface A or B is set during address out A or B.**
- **If busy or reserved with B, address out A will be answered with a "busy" status to the A interface, and vice versa.**
- **The interface which received "busy" will be informed with a "CU end" when the control unit is available.**
- **If the channel stacks status containing unit check or unit exception, the control unit will remain connected to that interface until the status is accepted.**

The basic purpose of interface switching circuits is to connect the tape control common circuits to whichever interface is operating at that time. Either of two channel interfaces can operate with the MIS tape control. To operate with an interface the corresponding operational trigger must be on to generate the necessary gate lines.

Selection

Address decoders in the tape control continuously monitor both interfaces. If the correct address bits arrive on the bus out lines along with an address out tag, the tape control generates a line called "addressed by channel (A or B)." Figure 34 illustrates the relationship of address decoders (3A and 3C) to the interface switch control triggers (6B and 6C).



● Figure 34. MIS Interface Switch Controls

Assume that the tape control unit is idle and is addressed by channel A. The AND circuit to turn on operational A (5B) requires three conditions:

1. Addressed by channel A.
2. Not reserve B or operational B.
3. The "gate busy to channel A" trigger must be off.

If these three conditions exist at the same time, the operational A trigger (6B) is turned on. Output of the operational A trigger generates gate lines to connect tape control circuits to the channel A interface.

A similar circuit is used to turn on operational B (6C); however, there is one important difference. The AND circuit to turn on the operational B trigger (5C) has four conditions instead of three as in the A circuit. The four conditions are:

1. Addressed by channel B
2. Not reserve A or operational A
3. The "gate busy to channel B" latch must be off.
4. The turn on circuit of operational A must be inactive.

The fourth condition is added to allow the A interface to take precedence over the B interface if both channels address the control unit at the same time. If the control unit is addressed by both interfaces at the same time, the turn on of operational B is blocked (5C) by "not turn on operational A."

Busy

While the tape control is operating with one interface, a select from the other interface will be answered with a "busy" signal. Assume that the B interface is operating when the A interface attempts to address the control unit (Figure 34). Operational B blocks the turn on circuit of operational A (5B). Operational B also conditions the turn on of the "gate busy to channel A" trigger (4A). There will be an active output from the off side of the gate busy trigger because it is held reset until addressed by channel A.

When the correct address and tag are received from channel A, the operational A trigger cannot turn on but the gate busy trigger is allowed to turn on. The gate busy trigger performs three functions:

1. Gates a busy response to the channel A interface (7B).
2. Turns on the CU end A trigger (7A) to remember that channel A tried to break in during operations with B.
3. Interlocks the turn on of operational A (5B) until addressed by channel A drops. (This prevents turning on operational A and gate busy to channel A at the same time if operational B happened to drop during channel A address out.)

The busy signal sent to channel A is a status byte with bits 1 and 3 on. Bit 3 indicates "busy" while bit 1 (status modifier) indicates that the busy condition applies to the control unit. Bits P, 1, and 3 are forced onto the bus in lines at the same time as the status in tag line is forced up. The status in trigger is not turned on during this short sequence.

CU End

Control unit (CU) end A trigger (Figure 34, coordinates 7A) remains on to remember that channel A tried to break into channel B operations. When the control unit is no longer operating with or reserved by interface B, the CU end A trigger turns on operational A (6B) to send a status byte to channel A. The status byte will contain a CU end (2 bit) to indicate that the control unit is now available for operations. A standard request in sequence is used to transmit the CU end status byte. The scanner unit address may not be the same as the previously selected unit.

At the end of an operation, the operational trigger for the active interface is reset unless a chain, stack, interrupt or unit check condition exists. The circuit to reset the operational triggers to neutral is shown in Figure 34. The reset is accomplished by a line called "reset operational trigger" (2C) which becomes active when the end status byte has been answered by command out or service out (1C).

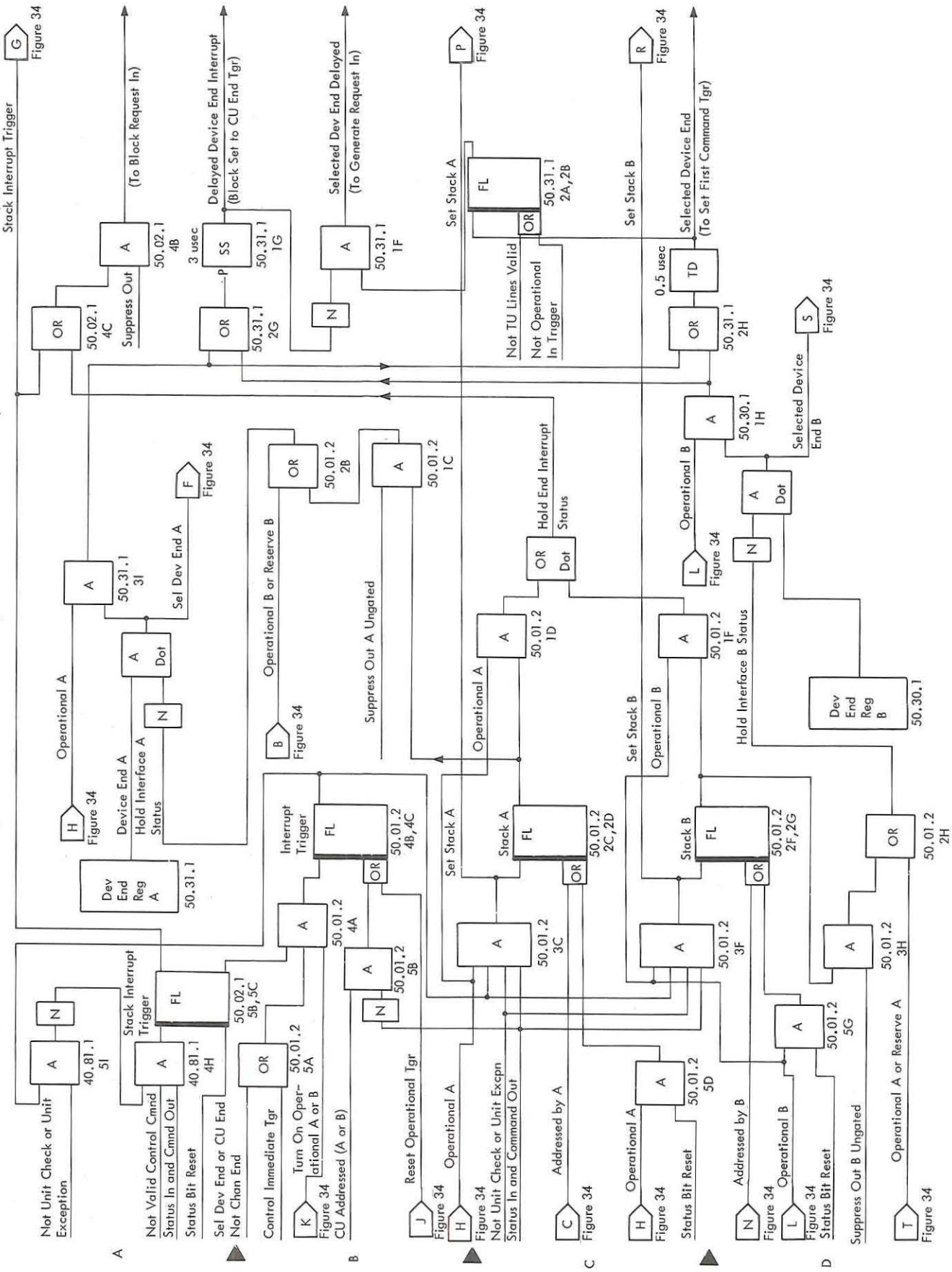
Stack

In some cases the data channel may refuse the end status byte that turns on a "stack" condition (Figure 35, coordinates 1A). If the status byte contains unit check or unit exception, the interface switch will remain connected to that interface until the channel accepts the status. If the status contains unit check, the connection will be maintained until a command other than no op or test I/O is received from the channel to which the status was presented (Figure 34, coordinates 2B). This procedure makes certain that the channel has an opportunity to investigate unit check condition before the other channel disturbs the control unit. When the interface connection is maintained because of unit check, the connection is defined as "contingent" (not part of the normal routine).

Stacking of status other than unit check or unit exception will not maintain the interface connection. The interface switch will be reset to neutral and the control unit will be available to either channel.

Stack Interrupt

The stack interrupt trigger (Figure 35, coordinates 1A) will turn on if the data channel refuses to accept a status byte at the end of a data transmission operation (not valid control command). Output of the stack interrupt trigger maintains the interface connection (Figure 34, coordinates 2B) until status bit



● Figure 35. MIS Device End

reset is generated (status is accepted or tape control is reset). If the status byte contained unit check, a contingent connection would be established (Figure 34, coordinates 1A).

Stack A and Stack B

The stack A or stack B trigger will turn on if a channel refuses to accept a "request in" status byte. This situation could arise when the tape control has a "selected device end" at the end of a control command because "request in" and "selected device end" turn on the interrupt trigger (Figure 35, coordinates 2B). The circuit which sets stack A or stack B also resets its operational trigger (Figure 34, coordinates 2D). Notice, on Figure 35, coordinates 2C, that the stack A or stack B trigger cannot be set if the status byte contains unit check or unit exception.

A status byte that contains only device end can be stored (stacked) in the tape control because the device end circuits have triggers for each tape unit. Operations on other tape units will not disturb the device end condition. Selected device end will be generated again when the interface is clear. A unit check or unit exception would be changed by new operations because they have single triggers (not one for each tape unit).

Device End

- **MIS device end circuits contain the standard device end triggers and scanner with an additional set of device end triggers for two-channel operations.**
- **If the tape control is operating with one channel, a device end for the other channel will be held until the tape control is free.**

The purpose of device end circuits is to signal the data channel when a device (tape unit) has completed a task and is ready to accept a new one. Because the MIS tape control is connected to two channels, the device end circuits must be modified to make sure the device end signal is sent to the correct channel.

Two sets of device end triggers are used, one set for each interface. Only one set of "TU rewinding" triggers is needed because the tape control can attach to only one group of tape units.

Selected Device End

In operation, the two sets of device end triggers perform the same function as the single set in a "one channel" 2803. They remember which unit has been selected for a task. With two sets of triggers the control unit can also remember which channel interface the select came from. Operations on interface A will "arm" the A device end triggers. Operations on interface B "arm" the B triggers.

If a tape unit completes a rewind operation by reaching load point, a selected device end signal will be returned to whichever channel initiated the rewind. A selected device end (Figure 34, coordinates 6A and 6B) from the A device end triggers will turn on operational A to send an interrupt status byte to channel A. A selected device end from the B triggers turns on operational B for a request in interrupt sequence to channel B (Figure 34, coordinates 7D). Output of the second set of device end triggers can stop the scanner in the same way as the first set of triggers. A line from each of the B triggers OR's with the corresponding A trigger. Either group can stop the scanner at the correct unit address (Figure 406).

Hold Status

Circuits have been added to hold a selected device end if the control unit is operating with or reserved by the opposite interface. Figure 35 shows the "hold interface status" lines (3A and 3C). If a hold interface status line is active, device ends cannot be gated out of the corresponding group of device end triggers.

Another condition which can cause the "hold interface status" condition occurs when suppress out is active (Figure 35, coordinates 1B and 1D) from the channel and the stack trigger is on. Selected device ends will not be gated out until either suppress out or stack drops (2B and 2D).

An MIS tape control contains separate stack triggers for the A and B interfaces (2B and 2C). Either stack trigger will remain on until the corresponding channel addresses the control unit or accepts a status byte from the control unit. Activity on the opposite interface does not directly affect the A or B stack triggers.

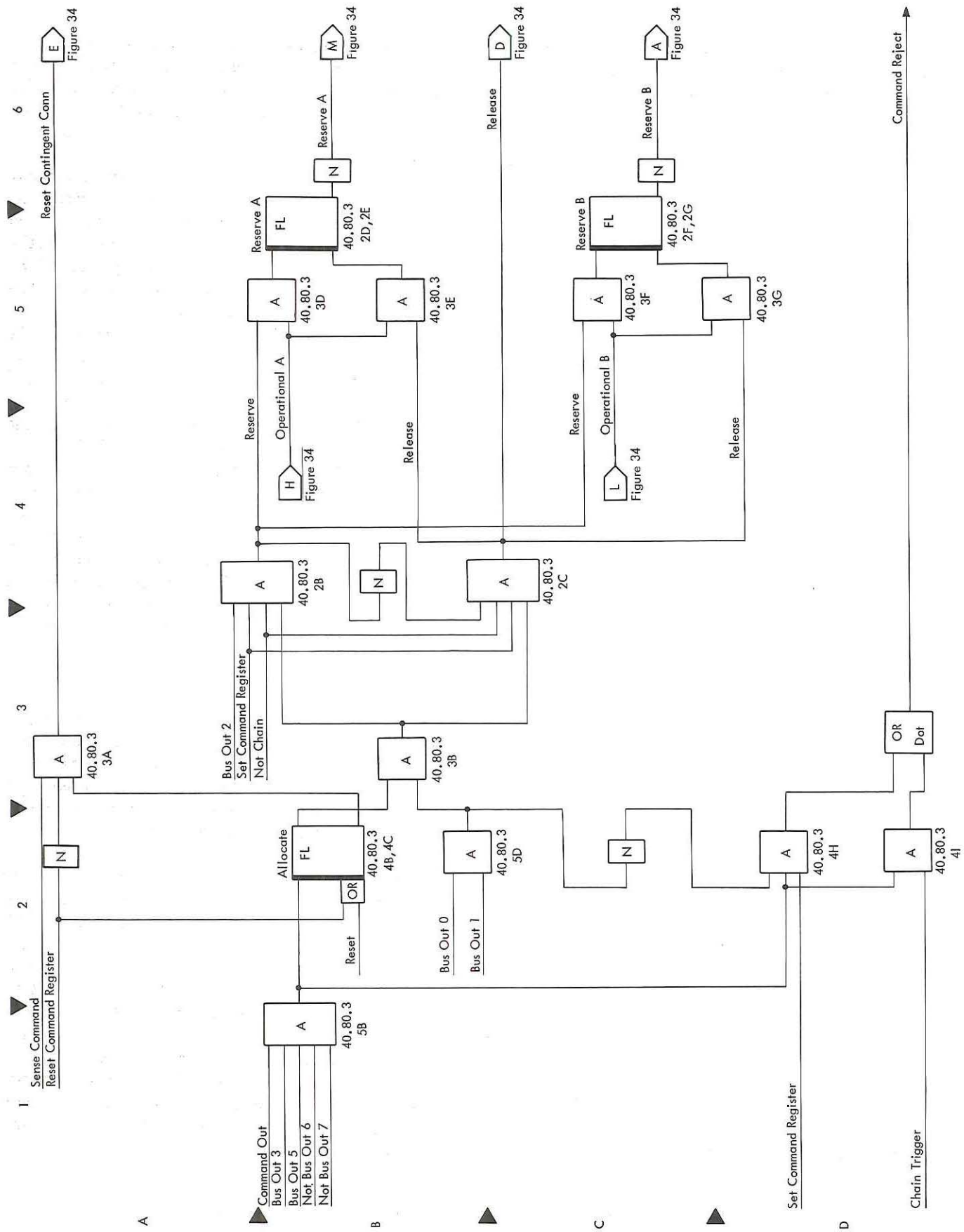
A selected device end, blocked by hold interface status, must wait until the hold condition drops. When the tape control is free of the hold condition, the selected device end is gated out to cause an interrupt sequence (Figure 35, coordinates 5A and 5C).

Resets

- **The MIS has two sets of general and selective reset triggers.**
- **A reset from an interface is blocked if the other interface is operating.**

The MIS tape control contains dual reset circuits which are interlocked to prevent interference between the channels. When the tape control is operating on one channel, a reset from the other channel is blocked.

General and selective resets function the same as for single-channel operation except for the interface



● Figure 36. MIS Reserve/Release

switching circuits and device end triggers. A reset from an interface will reset its own operational trigger but not the operational trigger for the other interface. A reset from channel A will turn off device end triggers in the A group but not the B group, etc.

In addition to the resets for A and B device ends and switching, a channel reset operation activates reset lines for the remainder of the tape control unit. A reset from channel A will reset operational A, device end A, CU end A, and control unit circuits. A reset from channel B will reset the corresponding B circuits as well as the common control unit circuits.

Reserve/Release Operation

- **A reserve command locks the MIS tape control onto one interface until a release command or reset from that interface.**
- **A release command resets the reserve triggers to allow operation on either interface.**
- **If command chaining is used, reserve or release can be used only in the first command of a chain.**
- **A reserve or release command while chained or when error correction "found track" is active will result in command reject.**
- **Reserve and release are performed during initial selection (command out) of a sense command.**

During some operations the task may require the tape control unit to remain attached to one interface exclusively. The reserve/release commands provide this ability under program control.

Reserve/Release

A reserve command places the tape control unit under control of one channel exclusively until that channel issues a release command. A reserve command from A or B turns on the reserve trigger for A or B. A release command resets the reserve trigger.

Both reserve and release functions are accomplished during initial selection time of a sense command. Modifier bits, in positions 0, 1, 2, and 3 of the sense command byte, identify the reserve and release operations. The sense command proceeds as usual after initial selection. Modifier bit 2 determines whether the command is a reserve or a release. A 2 bit indicates reserve. No 2 bit indicates release. The rest of the modifier bits remain on.

Reserve Command

A reserve command locks the tape control onto whichever channel interface the reserve came from. Figure 36 shows decoding and gating of reserve.

The bit configuration of a reserve command is a sense command except that modifier bits have been added. Sense has a 5 bit and no 6 or 7 bits. The added modifier bits are positions 0, 1, 2, and 3 on. A sense/reserve command appears on the bus out lines as 011 110 100.

During command out of a sense/reserve command, the "allocate" trigger is set (Figure 36, coordinates 3B). The same line that turns on the allocate trigger also samples the command reject conditions (2C). If chain is active, command reject is signaled (2D). If the reserve command does not have modifier bits 0 and 1 active, the command will be rejected.

A legitimate sense/reserve command turns on the allocate trigger (3B), which gates the input circuit for reserve A and reserve B triggers (4B). If the 2 bit is on, the turn-on circuit to the reserve triggers is active (5A and 5C). The reserve trigger for the operating interface is turned on (6A or 6C).

Output of the reserve trigger blocks the interface switch circuits for the opposite interface. The operational trigger for the other interface cannot turn on and the gate busy trigger is activated to send busy signals when addressed (Figure 34, coordinates 3A).

The reserve trigger will keep the control unit from operating with the opposite interface until a reset or a release command is received from the operating interface.

Release Command

A sense/release command resets the reserve trigger to allow the tape control to operate with either interface. The bit configuration of a sense/release command is the same as for a sense/reserve except for the P on and 2 off. A sense/release command appears on the bus out lines as 111 010 100 (sense command with modifiers).

As in the reserve operation (Figure 36) the release command turns on the allocate trigger (3B) and checks for command reject conditions (2C). A legitimate release command gates the input circuit to the reserve triggers (4B). Release contains no 2 bit so the turn on circuit to the reserve triggers cannot be active (4A). Because the turn on circuit is down, the reserve trigger reset circuit is active. The reserve trigger for the operating interface is reset (6A or 6C). Both the sense/reserve and sense/release operations are performed during initial selection of a sense command. After initial selection, the sense command is performed as usual.

Remote Switch Attachment

Tape controls with the two-channel switch feature can have the remote switch attachment. Channel

switching circuits can be manually controlled by switches in the remote attachment. When a remote switch is turned off, the corresponding channel interface is prevented from communicating with the data channel. This function is called "partitioning."

When an interface is "partitioned" by the remote switch, it appears to the channel the same as when power is down. Select out is bypassed to the next control unit and interface drivers are degated. If an operation is in progress when the switch is turned off, the operation will be completed before partitioning takes place.

The remote switches to control partitioning are attached to the tape control by a 6-wire cable. A 6-position connector for this cable is mounted on a bracket just above the tape control connector panel on the side of the B card gate.

The control switches are mounted in different locations according to the configuration of the system. (For example, the switches can be mounted on the 2167 Configuration Unit control console in a System/360 Model 67).

Review Questions

1. The multiple interface switching (MIS) feature, when installed on a 2403/2803 control unit, will:
   a. Allow one of two channels to reserve (attach) both the control unit and its tape units to that channel.
   b. Allow one of two channels to reserve (attach) only the control unit to that chanel, not the tape units.
   c. Allow one of two channels to reserve (attach) *either* the control unit *or* any number of attached tape units to the channel.

2. True/False

   _____ A tape control unit can be "locked" to one of two channels by either a programming function (reserve command) or by manual switches (partitioning).

3. Match the following command bytes to their associated functions.

   P 0 1 2 3 4 5 6 7
   _____ 1 0 0 0 0 0 1 0 0
   _____ 1 1 1 0 1 0 1 0 0
   _____ 0 1 1 1 1 0 1 0 0

   a. A reserve command that locks the control unit to one interface.
   b. A sense command whose only function is to submit up to six bytes of information to channel.
   c. A release command that resets the reserve status of the control unit from one channel.

4. True/False

   _____ When the tape control unit is neither reserved or "operational A/B" (triggers are both reset), the TCU is considered to be in a neutral state.

5. Once the TCU is reserved by Channel B, the opposite channel (A) can:
   a. never use TCU until released by the Channel B.
   b. use TCU if the Channel B is not executing a command..
   c. use TCU if Channel B is not executing a command and there isn't any stacked status.

6. Assume that neither channel reserved the tape control unit. Which of the following conditions *would not* generate a "busy" status to Channel B when B attempted to use TCU.
   a. TCU is executing a command for Channel A.
   b. TCU is retaining a previously rejected end status byte for Channel A containing a channel end and a device end bit.
   c. A unit check condition was encountered during the last operation performed for Channel A and Channel A has not yet interrogated the unit check condition.
   d. TCU is retaining a previously rejected end status byte from a motion control command for Channel A — only device end bit.
   e. None of the above.

7. Which action *does not* occur under a "busy" condition caused by either channel attempting to use TCU when already in use by the opposite channel?
   a. Force status byte containing P-1-3 bits on "bus in" lines.
   b. Set control unit end latch to present an interrupt to same channel when TCU is no longer busy.
   c. Force "status in" tag line active.
   d. None of the above.

8. A "contingent" connection is when the interface connection between TCU and channel is retained because of a:
   a. Unit exception condition
   b. Unit check condition
   c. Busy condition
   d. All of the above

9. Which of the following commands, from the same channel, can reset a "contingent" connection?
   a. No op
   b. Test I/O
   c. Any command except no op and test I/O
   d. All of the above

10. True/False

    _____ A general reset from channel A will reset both the "A" device end triggers and the "B" device end triggers.

Answers to Review Questions

1. a
2. True
3. b, c, a
4. True
5. a
6. d
7. d
8. b
9. c
10. False

Sixteen Drive Addressing

The sixteen drive addressing feature (SXT) allows the tape control to address sixteen tape units instead of only eight. Power circuits are not changed so a tape control still powers only eight tape units. The additional eight tape units must receive power from another source.

The SXT feature is used when the 2816 Switching Unit is attached to the control unit. The 2816 allows more than one control unit to communicate with a "pool" of tape units. In this case, the SXT feature provides up to sixteen possible addresses instead of only eight.

Tape Unit Addressing

The SXT feature adds a bit position to the tape unit select triggers (DE scanner). In a standard tape control, positions 5, 6, and 7 of the address byte are used as the tape unit address. When SXT is installed, position 4 is added to the TU select register to generate sixteen possible addresses.

Tape unit select triggers and device end scanner circuits are shown in Figure 406. The timing chart at the bottom illustrates the stepping sequence used during scanning. Eight device end triggers are added so that selected device end signals can be generated for any of the sixteen tape units.

CE Panel

A CE panel is built into the front of the A gate in each tape control unit. The CE panel shown in Figure 37 is used on IBM 2403, 2404, 2803, and 2804 tape control units.

Using this panel, a CE can check most of the circuitry in the control and tape units without disturbing the channel. The CE panel operates only when the off-line switch is on, logically isolating the control unit lines from the channel interface. A select out signal from channel bypasses the control unit and proceeds to the next control unit on line.

Some tape control functions cannot be simulated from the CE panel. Operations such as the control unit busy sequence and rejection of a status byte require channel lines. Data transmission operations such as reading and writing can be simulated without data or controls from the channel.

CE panel control circuits are shown on Systems 10.01.1 to 10.82.3.

• Indicators

The CE panel is organized into three sections: indicator lights, control switches, and plugboard. ALD page locations for the CE panel are in Figure 37.

Logically related indicators are designated by a horizontal line above all the positions in the group. A lighted indicator designates the 1 bit or ON condition of the corresponding circuit.

There are seven horizontal rows of indicators. The following descriptions are arranged, starting with the top row and proceeding left to right and down to the next row; then left to right again.

Out Indicators

The first group of indicators in the top row is the interface "out" tag group: SUP, OPR, ADDR, COM, SERV, and SEL. During on-line operations, these lights represent the active or inactive states of the "out" tag lines from the data channel. During off-line operations from the CE panel, they represent "tag" conditions generated by circuits in the control unit to simulate channel operation.

Initial Selection

Initial selection (INIT SEL) is active during the time that address information and commands are sent from the channel to the control unit. Every operation performed by the control unit must begin with an initial selection sequence. Initial selection is activated by on-line operations from the channel or off-line operations from the CE panel.

Chain

The chain indicator is active when a tape unit address is being held in the control unit by a chain condition.

Stack

Stack condition is turned on when the channel rejects a status byte by responding to "status in" with command out.

Read Delay

Read delay (RD) is active while tape is accelerating at the beginning of a read, read backward, or space operation.

Read Disconnect Delay

The read disconnect delay (RDD) latch is active for read, read backward, space, and write (checking) operations. The RDD latch is used primarily to check timing between bytes and to detect check characters, tape marks, lost characters, and interblock gaps.

Read Condition

Read condition (RD CND) is active when the tape control is accepting bytes from the tape unit. Read condition is active during read and space operations as well as during write (checking) operations.

Correct

The correcting latch is turned on during a nine-track error correction read operation.

Found Track

Found track ON indicates that a nine-track read or read backward has encountered a data error and has located the tape track that contains the error.

In Indicators

The first group of indicators in the second row is the interface "in" tag group: OPR, ADDR, STA, SERV, and SEL. During on-line operations, these indicators represent the active or inactive states of the "in" tag lines from the control unit to the channel. During off-line operations, they represent "tag" conditions generated by circuits in the control unit to simulate channel operation.

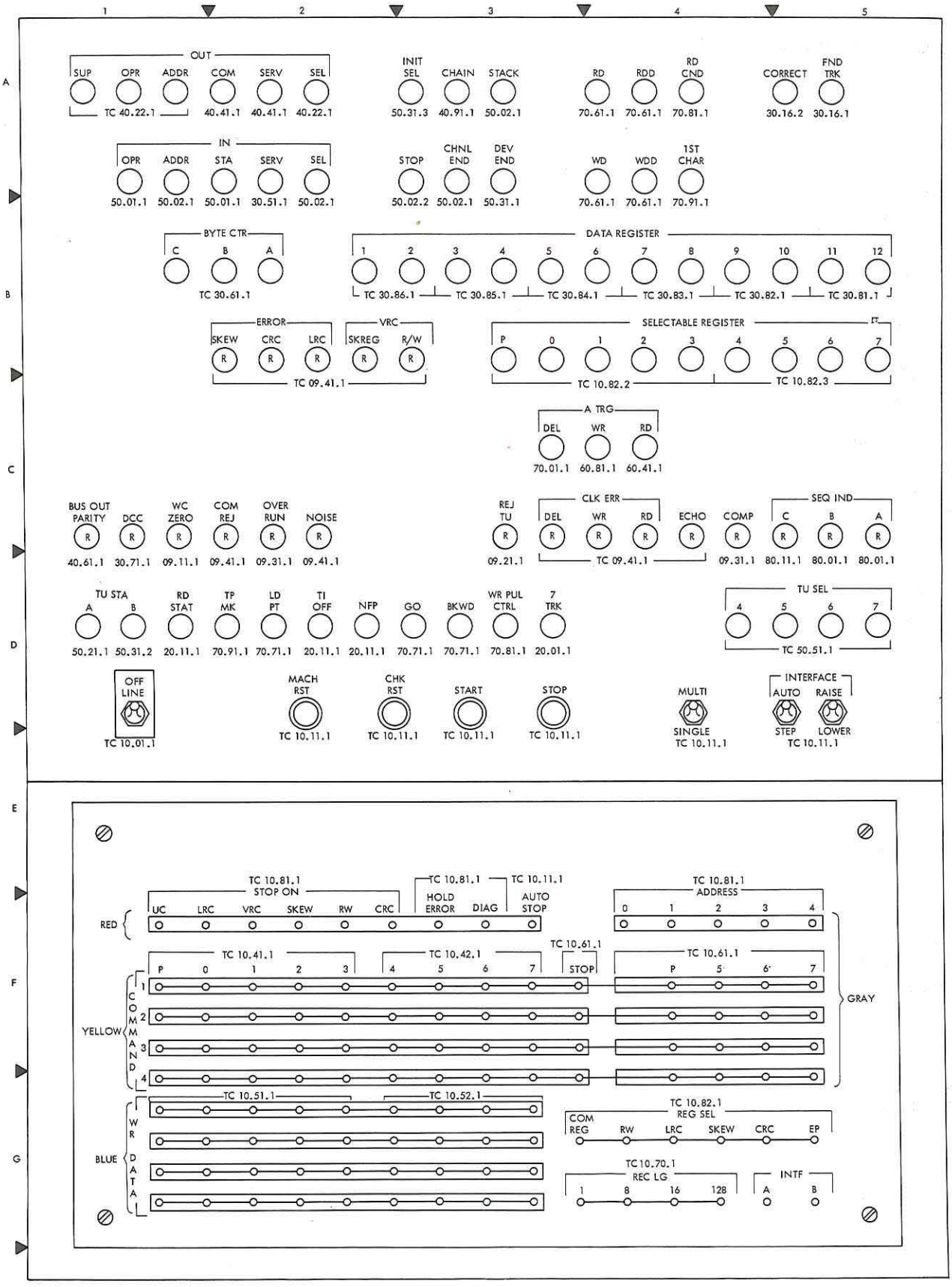


Figure 37. CE Panel Locations 2403/2803

Stop

Stop is active for four types of conditions:

1. Channel refused to accept or transmit more data bytes during a read, read backward, or write operation.
2. Interface disconnect: when select out is inactive while operational in and address out are active.
3. Error stop as a result of a "stop on error" plug in the CE panel.
4. End of a write tape mark operation.

Channel End

Channel end (CHNL END) is a status latch to signify that the data channel is no longer needed to complete the operation in progress. Channel end occurs at the completion of data transmission operations and also when an end pulse occurs as a result of TU reject, TAU end word count zero, or stack no op.

Since the channel is not needed to complete control commands, a channel end is generated for the initial selection status byte. This channel end is generated by valid control command circuits and does not turn on the channel end indicator.

Device End

Device end (DEV END) is a status condition that indicates a device (tape unit) has completed an operation.

Write Delay

Write delay (WD) is active while tape is accelerating at the beginning of a write, write tape mark, or erase operation.

Write Disconnect Delay

Write disconnect delay (WDD) is active during the ending sequence of write operations.

First Character

Start reset turns on first character (1ST CHAR) at the beginning of each operation and after a selective, general, or machine reset. The first character indicator remains on until RC reset time of the next read clock cycle (first byte).

Byte Counter

The first group in the third row of indicators is the byte counter. Three indicators (C, B, and A) represent the three triggers in the counter circuit. The byte counter is used only for seven-track data conversion operations. Data passing to and from tape is divided into groups of bytes during the conversion process. The byte counter generates gating lines to control these data groups as they pass through the control unit.

Data Register

Twelve indicators show the contents of all twelve positions of the data register. Positions 1-4 are used only for seven-track data convert operations.

Skew Error

The first indicator in the fourth row is skew error. Skew error indicates that excessive skew has been detected in the data bytes during a read, read backward, or write operation.

CRCR Error

During a nine-track operation, a CRCR error indicates that a data error occurred but the CRC circuits did not locate the track in error (TIE). Cyclic redundancy checking is not used during a seven-track operation.

LRCR Error

An LRCR error indicates that one or more tracks in the data block do not contain an even number of bits. Any bit that remains in the LRC register at the end of the operation can cause an LRCR error, unless it is the track that is being corrected. NRZI error correction circuits automatically block LRCR errors from the track in error.

Skew Register VRC (SK REG)

During write checking, the high-clip skew register VRC circuit indicates errors in the bytes just written on the tape. Skew register VRC does not indicate errors during read backward operations.

Read/Write VRC

A VRC circuit monitors the parity of bytes in the read/write register. During a read or read backward operation, the R/W VRC circuit can indicate errors. During a write operation, this circuit does not indicate errors; it generates correct parity (P bits) for the bytes being written.

Selectable Register

The selectable register is a group of indicators that can display any one of a group of registers when a jack plug is inserted in the desired position on the plugboard. Registers that can be selected are:

1. Command register.
2. Read/write register.
3. Longitudinal redundancy check register (LRCR).
4. High-clip skew register.
5. Cyclic redundancy check register (CRCR).
6. Error pattern register (EPR).

A Triggers

There are only three indicators in the fifth row:

1. Delay counter "A" trigger.
2. Read clock "A" trigger.
3. Write clock "A" trigger.

When the tape control unit is stopped, or reset, these three indicators are off. The "A" triggers are on only during actual operation of the corresponding circuits.

Bus Out Parity

The first indicator in the sixth row is bus out parity. This indicator turns on if the tape control unit finds a parity error (even parity) in a byte on the bus out lines from the channel. The error can be in a command byte, a data byte, or a TIE (request track in error) byte.

Data Convert Check (DCC)

Data convert check is used only during seven-track data conversion operations. During conversion operations, data bytes are handled in groups of four on the tape. If a data block read from tape does not have a multiple of four bytes, a DCC can result.

Word Count Zero (WC ZERO)

When a write command is received from the channel, the tape control requests the first byte of data before any tape motion is started. If the channel does not send the first data byte, tape does not move and the word count zero indicator is turned on.

Command Reject

A command reject condition indicates that the tape control has received a command that it cannot perform. This can be caused by a command to write on a tape that is file-protected or by a mode set command to turn on the data converter in a control unit that does not have a data converter.

Overrun

An overrun error is caused by timing conditions on the channel interface. If the channel does not respond within a specified time limit to a data request (service in), an overrun error occurs. Overrun can occur during read or write operations.

Noise

The noise detection circuits are used to detect signals from the tape when there should be no noise. Bits read from the tape during start or stop delay times can cause noise errors.

Reject Tape Unit (REJ TU)

A "reject tape unit" indicates that the selected tape unit did not respond correctly. For write, write tape mark, or erase operations, the tape unit should respond with write status. For read-type operations (including spacing), the tape unit should respond with read status. If the correct response does not occur, a tape unit reject signal is turned on. Reject tape unit can also occur if the tape unit does not respond with a model (N) line (select and ready).

Clock Error (DEL, WR, and RD)

Checking circuits monitor the stepping sequence of the delay counter, write clock, and read clock. If the triggers fail to step in the correct sequence, the corresponding error indicator is turned on.

C-Compare (C-COMP)

C-Compare circuits monitor data bytes as they pass through the tape control unit. If the parity of a byte changes, an error has occurred somewhere in the control unit data path.

Sequence Indicators

A tape control unit must perform a specific sequence of steps for each command. Sequence indicators are turned on or off as the command sequence progresses. When the command is completed, all three sequence indicators should be off. Any sequence indicator that is on at the end of a command will cause an equipment check.

If a command is not completed (hangs up), the sequence indicators that are on can be analyzed to determine how far the operation progressed before the "hangup."

Tape Unit Status A and B

A selected tape unit indicates its status to the tape control by submitting two lines: "model (N)" and "not ready." These two lines become "TU status A" and "TU status B" in the tape control. The following chart summarizes TU status A and B conditions:

INDICATORS		CONDITIONS	
A	B	TU STATUS	STATUS BYTE RESPONSE
0	0	Nonexistent tape unit	Unit check
0	1	Not ready	Unit check
1	0	Ready and not rewinding	Clear status
1	1	Ready and rewinding or switched	Busy status

Read Status (RD STAT)

With the read status light active, the selected tape unit is in read status.

Tape Mark (TP MK)

The tape mark indicator is turned on if the control unit detects a tape mark during a read, read backward, or space operation.

Load Point (LD PT)

If an "at load point" signal is received from the selected tape unit, the load point indicator is turned on. It remains on until the control unit receives another command.

Tape Indicator Off (TI OFF)

The tape indicate line from the tape unit notifies the control unit that the end of tape has been reached. As long as tape indicate remains off (indicator lit), the tape can be used.

Not File Protect (NFP)

Valuable information on a reel of tape can be protected by removing the "write enable ring" from the tape reel. Each tape unit has a sensing pin to determine whether the ring is in or out. As a warning to operators and programmers, the not file protect indicator turns on when the "write enable ring" is in place (not protected).

Go

The go indicator turns on when the tape control instructs the tape unit to move tape either forward or backward.

Backward (BKWD)

During backward tape motion commands (read backward or backspace), the backward latch instructs the tape unit to move tape backwards. The backward latch is reset at the beginning of the block as soon as the check characters have been read.

Write Pulse Control (WR PUL CTRL)

The write pulse control indicator is attached to the write trigger release circuit. This circuit is active while information is being written on tape. When this circuit is inactive, write triggers in the tape unit are held reset so that nothing can be written on the tape.

Seven-Track (7 TRK)

The seven-track indicator is active when the selected tape unit is equipped with a seven-track read/write head. It remains off during nine-track operations.

Tape Unit Select (TU SEL)

Tape unit select indicators (4-7) represent the status of the tape unit select latches in the control unit. They correspond to bits 5, 6, and 7 of the address byte received from the channel (or the CE panel plugs) during initial selection. Position 4 is active only if the control unit has the SXT feature installed.

The tape unit select latches are also the device end scanner latches. When the device end scanner stops during a command, the lights can be read on the indicator panel. When the scanner is running, the latches change too fast to be readable.

Switches and Pushbuttons

Off Line: The off line toggle switch in the off line (on) position activates the CE panel controls and degates the channel interface circuits to isolate the control unit. Machine reset and check reset are both active for on line and off line mode. Registers, latches, and indicators can be reset during on line operation; however, this disrupts any operation in progress.

Check Reset: The check reset pushbutton is gated by not busy except test I/O, which means check reset is disabled if any one of three conditions exist: selected device end, channel end, or stack interrupt. A part of the check reset circuit is gated by the hold error jack plug. If hold error is jack plugged on, the reset is blocked for seven data error triggers. They are:

1. LRCR error.
2. CRCR error.
3. High-clip vertical redundancy error.
4. Forward stop delay noise.
5. Read/write vertical redundancy error.
6. Echo error.
7. Skew error.

During normal operation the data error triggers are reset each time a new command is received. When hold error is plugged on, the errors are retained so the CE can observe or record them.

Start and Stop: The start and stop pushbuttons control starting and stopping the program in the plugboard command jacks.

Multi/Single: The multi/single toggle switch controls stepping the four position instruction ring for the plugboard commands. In the MULTI position the ring steps through all four commands, returns to position 1, and repeats the sequence until it is stopped by a reset or stop. Single cycle holds the ring reset to position 1. Command 1 is repeated continuously unless reset or stopped.

Interface Sequencing: Two switches (marked auto/step and raise/lower) control simulation of interface tag out lines. When the auto/step switch is in the auto position, tag out lines are simulated at electronic speeds similar to channel operation. The step position places tag out simulation under control of the raise/lower switch.

During step operation the start pushbutton starts an initial selection sequence. If the raise/lower switch is in the "lower" position, initial selection sequence does not progress past address in. If the switch is in raise position, the operation continues through command out. In the raise position the reset to command out is blocked. Operation halts at this point because the tape op trigger and the status in trigger cannot be turned on until command out is reset.

When the switch is returned to the lower position, command out is reset, allowing the operation to proceed to service out.

Service out cannot be set until the switch is moved to the raise position again. Service out is reset when the switch is moved again to the lower position.

If the interface step (raise and lower) is used for a read or write operation, the overrun trigger is turned on. Overrun blocks shift pulses so data transfer is blocked.

Plugboard

The plugboard portion of the CE panel consists of 16 groups of jacks to program commands, data, and controls into the tape control manual circuits. Inserting a plug in the jack completes a circuit to activate the function plugged. Plugs used have no wires attached; all circuit contacts are completed by the shaft of the plug inside the jack.

Commands (Yellow): Four groups of jacks (marked command 1, 2, 3, 4) are used to set up instructions for the tape control. Any four commands can be programmed to execute one at a time or repeatedly at high speed. Commands plugged into the CE panel must be in odd parity to prevent errors on the bus out lines.

An instruction ring controls the sequencing of commands. The ring is actually two binary triggers decoded into 4 steps:

00 = 1	10 = 3
01 = 2	11 = 4

Stepping of the ring triggers is controlled by the "any command trigger." Each time the Any Command trigger is reset by status in, the instruction ring is stepped.

Stop (Yellow): A stop jack is to the right of each command group. If a stop jack is plugged, the tape control stops after execution of the associated command.

Addresses (Gray): Each command group has a corresponding address group on the right side of the panel. The instruction ring activates the address group and the corresponding command group at the same time. Each address group supplies the tape unit address for its command. Up to four different tape units can be used with four different commands without changing jack plugs.

Above the unit address group is a tape control address group. The address of the tape control must be plugged into this group for CE panel operations. Tape control and unit address positions are sampled for the address byte during interface sequencing. The control unit address (0, 1, 2, 3, 4) and tape unit addresses (P, 5, 6, 7) must have odd parity when combined in an address byte on the bus out lines.

Write Data (Blue): Below the command jacks are four groups of write data jacks. Any four bytes can be plugged into these positions as long as correct parity is maintained. A character counter, similar to the instruction ring, controls the outputs of the data jacks. Data bytes are gated (in sequence) to the bus out lines. Service in gates the data bytes on the bus out lines and steps the character counter. The four data bytes repeat as many times as necessary to complete the specified record length. The same write data bytes are used by all four command positions.

Record Length: Record length is specified by plugging a jack marked 1, 8, 16, or 128. Records written from the CE panel consist of the selected number of bytes plus check characters. If more than one record length is plugged, the record size written is the smallest number plugged.

Records written from the CE panel using the data and character count in Figure 38 produce the results indicated. Be sure that the correct bit is plugged in all four characters of the write data.

During CE panel write operation, the delay counter counts the number of bytes in each record. Off line AND's with a write clock 2 pulse to drive the delay counter once for each byte. When the delay counter reaches the specified record length, a stop command is brought up through the jack plug. If no record length is plugged, continuous data are written until the stop trigger is turned on by the stop switch or tape indicate.

Register Select: The register select group allows selection of a register to be displayed in the selectable register indicators. When the plug is inserted, the corresponding register contents appear in the indicators. If more than one position is plugged, the indicators contain the OR of all the selected registers.

Stop On Error (Red): Just above the command jacks is a group of jacks called stop on. Plugs in these positions cause the tape control to stop when an error of the type plugged occurs. These registers retain the data they contained at the time of the error:

1. Read/Write Register.
2. Skew Register.
3. Longitudinal Redundancy Check Register.
4. Cyclic Redundancy Check Register.
5. Error Pattern Register.

Error types which can be selected to cause a stop are:

1. Unit Check.
2. Longitudinal Redundancy Check.
3. Vertical Redundancy Check (High Clip Register).
4. Skew Error.
5. Read/Write Vertical Redundancy Check.
6. Cyclic Redundancy Check.

Any combination of error types may be selected. The tape control will stop for whichever one occurs first.

CAUTION: Be careful to remove these plugs after CE panel operation because they remain active during on line operation (jacks outlined in red).

Stop on error can be used as a method of pinning down intermittent failures during customer operation or CE system time. Also, the hold error jack blocks the reset to seven error triggers:

1. Longitudinal redundancy check
2. Cyclic redundancy check

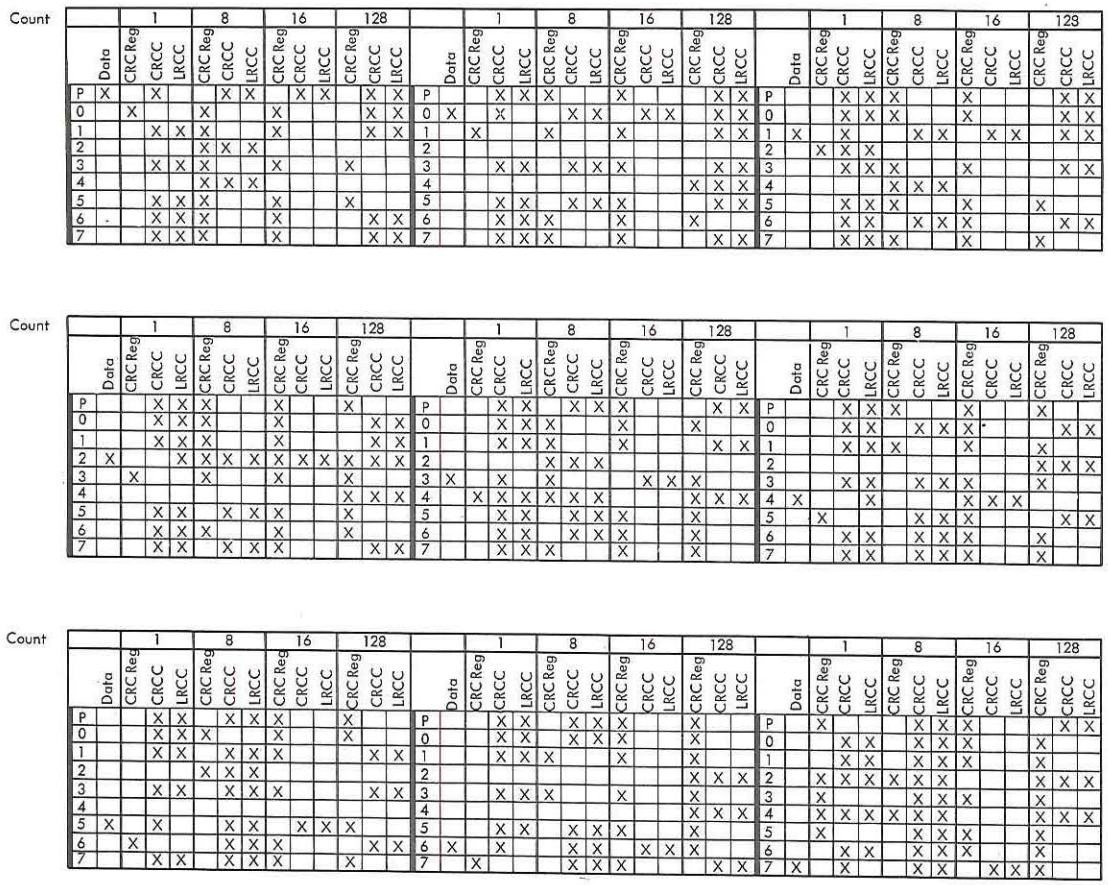


Figure 38. CRC Register Bit Patterns

3. High-clip vertical redundancy check
4. Forward stop delay noise
5. Echo error
6. Read/Write vertical redundancy check
7. Skew error

Hold error is effective during on line operation; it is outlined in red.

Auto Stop (Red): The auto stop jack (also red), causes the tape control to stop when a unit exception or tape indicate is recognized. Unit exception is turned on by either of two conditions: reading a tape mark unexpectedly (when file search is down); or a tape indicate signal during a write operation. Auto stop is active but generally not used during customer operation.

Diagnostic Mode (Red): Diagnostic mode is enabled by the diag jack. This plug must not be in during customer operation. With the diag plug in place, a diagnostic command can be given from the CE panel (off line) or the channel (on line). The plug allows a diagnostic command to set the diagnostic trigger, which blocks all parity bits on the write bus lines.

Diagnostic mode is very useful when writing error circuit test records from the CE panel or the channel. A bad parity character can be written in a record, and then read back to test error detection circuits. One or more blanks can be written in a record to check lost character detection.

The diagnostic mode trigger also enables the latch back of the force burst mode trigger. When the tape control is attached to a multiplex channel, it can operate in either burst mode or byte mode. If the diagnostic jack is in place but the diagnostic trigger is reset, a read command forces byte mode. Byte mode allows simultaneous compute and I/O operation so that tape record gaps can be timed by the CPU. When the diagnostic jack is not plugged, the burst mode trigger is allowed to operate normally.

Interface A-B: Two jacks in the bottom right corner of the CE panel are marked interface A-B. These jacks are used for operation on an IBM 2404 or 2804 simultaneous (read and write) tape control unit. A plug in the A or B jack logically connects the CE panel circuits to the corresponding interface during off line operation.

Diagnostic Mode

Diagnostic mode is used with the CE panel to check tape operation. With the diag plug in place in the CE

panel, a diagnostic mode command turns on the diagnostic trigger. During write operations, the diagnostic trigger blocks all bits in the P track of the write bus.

Records can be written with good or bad parity characters to test error detection circuits. Blank characters can be written in a record to check lost character detection.

The diagnostic mode trigger also affects the force burst mode trigger. Tape motion diagnostics such as IBG tests must compute time while tape is being read. When the tape control is attached to a slow speed multiplex channel, byte mode must be forced, to allow the CPU to run while the channel is busy reading tape. The three conditions needed to disable the burst mode trigger and force byte mode are: read command, the diag jack in place, and the diagnostic trigger off. The burst mode trigger operates normally for commands other than read. The diagnostic mode trigger must remain reset during IBG test reading on a multiplex channel.

Marginal Checking

The −12M output from the power supply is used for marginal checking. A jack is provided on the power supply so that a portable ±3v marginal check power supply (IBM P/N 210860) may be connected in series with the −12M output. The 3v power supply will either add to or subtract from the normal voltage to produce a usable range of −9v to −15v. When the portable supply is not plugged into the jack, the −12M line is connected to the −12 line by a shunt contact in the jack.

The −12M supply has been selected for marginal checking because it is the bias reference voltage for most of the transistor bases in the tape control. By disturbing the normal operating levels, many intermittent troubles can be located and repaired during scheduled maintenance instead of on customer time.

The tape control unit must operate error free in all modes of operation through a range of ±3v on the −12M line. If errors occur between −9v and −15v, the cause must be found and corrected as soon as possible.

Appendix

Tape Control Data Rates

7/9 TRACK, 800 BPI	RATE, KC
Model 1 TU	30.0
Model 2 TU	60.0
Model 3 TU	90.0

7-TRACK, 556 BPI	RATE, KC
Model 1 TU	20.8
Model 2 TU	41.7
Model 3 TU	62.5

7-TRACK, 200 BPI	RATE, KC
Model 1 TU	7.5
Model 2 TU	15.0
Model 3 TU	22.5

Physical Characteristics*

UNIT	WEIGHT, LB
2403-1, 2, 3	2000
2803-1, 2, 3	1400
2404-1	2000
2804-1	1600

*For all units: Height is 60 inches; width is 60 inches; depth is 29 inches.

Power Requirements

60-cycle, 208v and 230v ± 10%, or
50-cycle, 195, 220, and 235v ± 10%

UNIT	POWER USAGE, KVA
2403-1, 2,3	2.1
2803-1, 2, 3	1.0
2404-1	2.4
2804-1	1.5

Air Conditioning Requirements

	OPERATING	NON-OPERATING	STORAGE
Temperature	60 to 90°F	50 to 110°F	33 to 150°F
Humidity	20 to 80%	8 to 80%	0 to 80%

UNIT	HEAT DISSIPATION, BTU/HR	AIR FLOW, CFM
2403-1, 2,3	5500	1000
2803-1, 2, 3	2500	500
2404-1	6300	1200
2804-1	4000	700

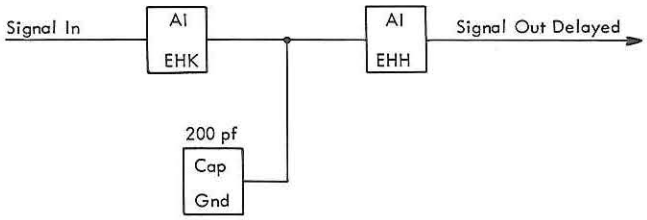
Capacitor Delays

A delay used in the tape control (mostly for interface lines) consists of two inverters with a capacitor to ground on the line between the inverters (Figure 39).

When the input to the first inverter is down, the capacitor is charged to the plus level of the inverter output. The capacitor cannot discharge immediately when the input rises. A period of time passes before the capacitor discharges enough to change the state of the second inverter. The size of the capacitor determines the length of delay.

Action is reversed when the input changes to a down level. The capacitor must charge past the switch point to change the state of the second inverter.

Because of differences in the charge and discharge paths, the delay time of the leading and trailing edge of the pulse may be unequal, so this delay is used only where the delay timing is not critical.



Capacitor Size	Delay Timings		
	Low	Med	High
200 pf	0.3 usec	0.6 usec	1.0 usec
390 pf	0.5 usec	0.9 usec	1.5 usec
200 pf + 390 pf	0.6 usec	1.25 usec	2.0 usec

Figure 39. Capacitor Delays

Cabling

Figure 40 illustrates tape control and tape unit cabling, including power and signal cables. Units shown in Figure 40 do not represent a typical installation. Combinations of units were chosen to best illustrate a variety of cabling layouts.

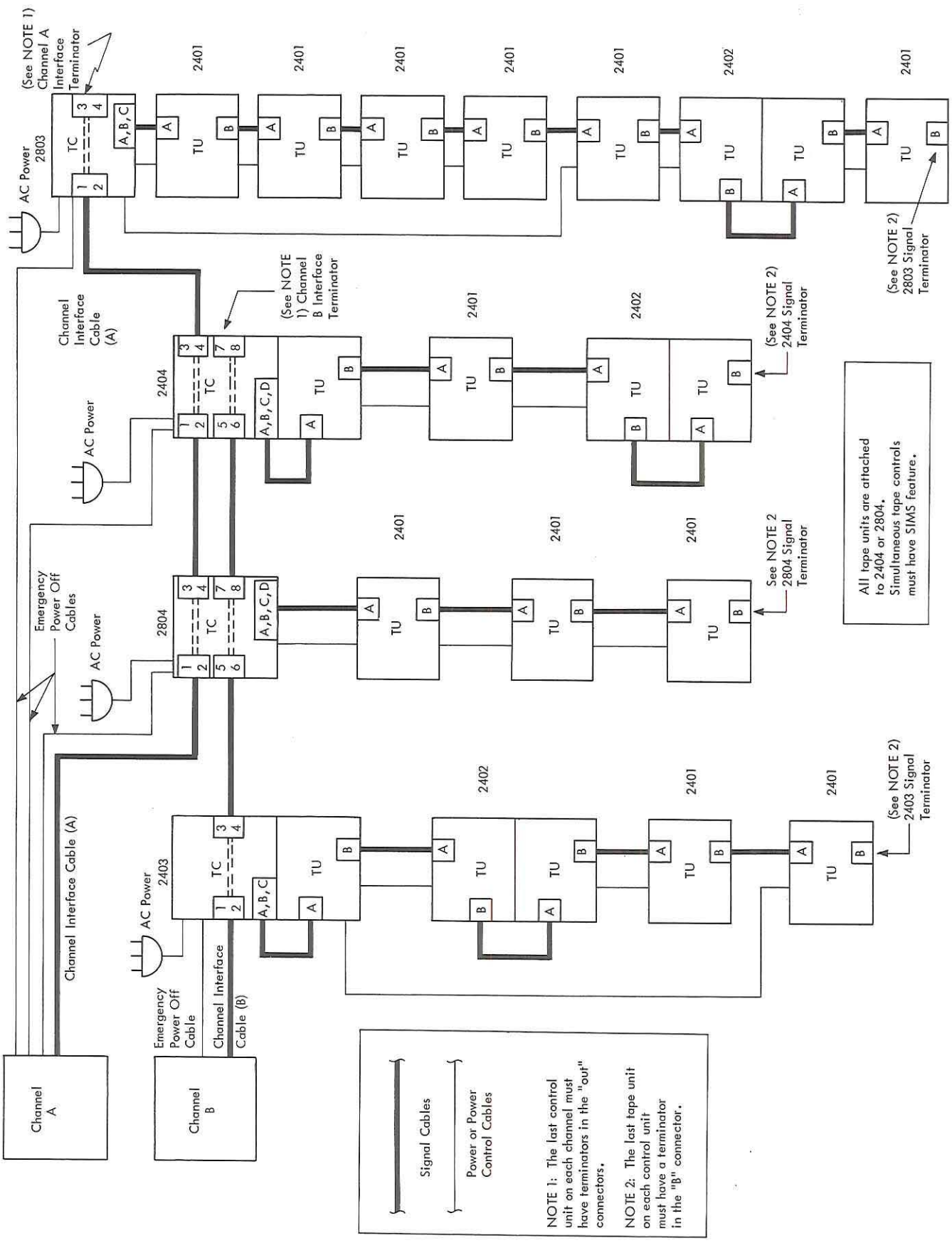


Figure 40. Tape Control and Tape Unit Cabling

CUT ALONG LINE

COMMENT SHEET

2803/2804 MODEL 1 TAPE CONTROL AND TAPE CONTROLS FOR 2403/2404 MODELS 1, 2, AND 3

FIELD ENGINEERING THEORY OF OPERATION, FORM Y22-2853-1

FROM

NAME _____ OFFICE/DEPT NO. _____

CITY/STATE _____ DATE _____

To make this manual more useful to you, we want your comments: what additional information should be included in the manual; what description or figure could be clarified; what subject requires more explanation; what presentation is particularly helpful to you; and so forth.

How do you rate this manual: Excellent ____ Good ____ Fair ____ Poor ____

Suggestions from IBM Employees giving specific solutions intended for award considerations should be submitted through the IBM Suggestion Plan.

NO POSTAGE NECESSARY IF MAILED IN U.S.A.

FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), STAPLE AND MAIL

fold fold

FIRST CLASS
PERMIT NO. 419
POUGHKEEPSIE, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

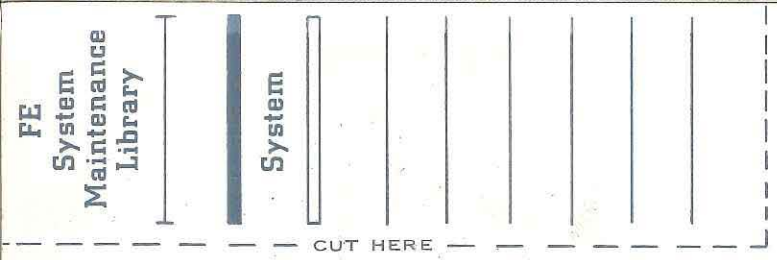
IBM CORPORATION

P.O. BOX 390
POUGHKEEPSIE, N.Y. 12602

ATTENTION: FE MANUALS, DEPT. B95

fold fold

SEQ INDICATOR	WRT. OP	RD OP	RD BKWD OP
A ON	WRT. DELAY	RD. DELAY	RD. DELAY
B ON	WRT. 1ST CHAR	RD. 1ST CHAR	RD. 1ST CHAR (BKWD)
C ON	STOP WRITTING (WDD ON)	END OF RD. DATA (RDD-ON)	READ LAST CHAR. (RDD-ON)
A OFF	END OF RD CHECKING WRT. DATA	RD. 1ST CHK. CHAR.	RDD AND SEQ C ON
B OFF	RD LAST CHK. CHARACTER	RD. LAST CHK. CHAR.	RD. CONDITION OFF
C OFF	INTO I.B.G.	INTO I.B.G.	INTO I.B.G.



Y22-2853-1

Printed in U.S.A. Y22-2853-1

IBM®

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601