

GA22-6974-08  
File No. S360/S370-19

**Systems**

**IBM System/360 and  
System/370 I/O Interface  
Channel to Control Unit  
Original Equipment  
Manufacturers' Information**

The IBM logo, consisting of the letters 'IBM' in a bold, sans-serif font, with each letter formed by a series of horizontal bars of varying lengths, creating a striped effect.

## **Ninth Edition (March 1986)**

This is a reprint of GA22-6974-7 incorporating changes released in the following Technical Newsletters:

GN22-0684 (dated 12 June 1985)

GN22-0689 (dated 22 November 1985)

This manual is provided for use in conjunction with other relevant IBM publications, and IBM makes no warranty, express or implied, relative to its completeness or accuracy. The information in this manual is current as of its publication date but is subject to change without notice. Before using this publication, consult the latest *IBM System/360 Bibliography*, GC20-0360, and the *IBM System/370, 30xx, and 4300 Processors Bibliography*, GC20-0001, for editions that are applicable and current.

This manual is not intended to be a manufacturing or engineering specification; it is assumed that the reader understands the interrelationships among any affected systems, machines, programs, and media. However, this manual does provide information that may be of interest to designers and manufacturers of equipment to be attached to the IBM System/360 or IBM System/370. IBM's responsibilities resulting from such an attachment are defined in the *IBM Multiple Supplier System Bulletin*, G120-6648. Any references to this manual should include the revision level and the publication date.

IBM may have patents or pending patent applications covering subject matter described herein, including appearance patents or applications. Furnishing this manual does not constitute or imply a grant of any license under any patents, patent applications, trademarks, copyrights, or other rights of IBM or of any third party, or any right to refer to IBM in any advertising or other promotional or marketing activities. IBM assumes no responsibility for any infringement of patents or other rights that may result from the use of this manual or from the manufacture, use, lease, or sale of apparatus described herein.

Licenses under IBM's utility patents are available on reasonable and nondiscriminatory terms and conditions; IBM does not grant licenses under its appearance design patents. Inquiries relative to licensing should be directed, in writing, to: Director of Contracts and Licensing, International Business Machines Corporation, Armonk, New York 10504.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by IBM Corporation, Product Publications, Department B98, PO Box 390, Poughkeepsie, NY, USA 12602.

© Copyright International Business Machines Corporation 1971, 1978, 1981, 1983, 1984.

The I/O interface is one type of communication link between a channel<sup>1</sup> and the various I/O control units in IBM System/360 and System/370. It employs information formats and control-signal sequencing that provide a uniform means for attaching and controlling various types of control units.

Information -- in the form of commands, data, status information, sense information, control signals, and I/O-device addresses -- is transmitted over the time- and function-shared lines of this interface. All exchanges of information are between a channel and attached control units.

The design of this interface provides the following important features:

- Consistency in input/output programming over a wide range of control units.
- A ready physical connection to System/360, System/370, ECPS:VSE, and 370-XA channels of control units designed by any manufacturer to operate with this interface.
- The means to physically accommodate future control units designed to meet the parameters of this interface.
- The capability for providing an interlocked interface operation that is, in most cases, time-independent; this feature increases the range of control units that may be attached.
- The capability for providing a noninterlocked interface operation

that permits attachment of control units having higher data-rate requirements than those achievable through interlocked operations.

- The servicing of up to eight control unit attachment points for each set of lines.

This manual provides a functional description of the interface lines; a description of the electrical, mechanical, and cabling considerations; and an operational description of the I/O interface. It does not define the interface between control units and I/O devices. Additionally, the flow diagrams (Figures C-1 through C-8) and the sequence charts (Figures C-9 through C-13) in Appendix C are provided as examples only and should not be used as a means of obtaining precise definitions of interface sequences. Refer to the main body of this manual for the I/O-interface-sequence definitions.

Unless otherwise stated, the information in this manual pertains to control units attached to only one I/O interface.

For a specific IBM channel or control unit that implements this interface, parts of the I/O interface definition are model-dependent or may not apply; for further information, refer to the appropriate System Library manual for that unit.

No statement in this manual should be construed as limiting extensions to the I/O interface. The I/O interface, or any aspect of the I/O interface, may be altered from time to time by IBM or may be withdrawn by IBM in part or in whole.

<sup>1</sup> The term "channel" is used in this manual when referring to the end of the interface with which a control unit communicates. When the term is used, it should be understood to apply to the System/360 architecture, System/370 architecture, ECPS:VSE architecture, and System/370 Extended Architecture (370-XA). If, however, the description applies only to systems operating in the 370-XA mode, the term "channel subsystem" is used instead.



## CONTENTS

<b>CHAPTER 1. FUNCTIONAL DESCRIPTION</b>		1-1		2-14
Line Definition		1-3	Control-Unit End	2-14
I/O-Interface Lines		1-3	Temporary Control-Unit Busy	2-15
Buses -- General		1-5	Busy	2-15
Bus Out		1-6	Channel End	2-16
Bus In		1-6	Device End	2-16
Selection Controls and Tag Lines		1-7	Unit Check	2-17
Operational Out		1-7	Unit Exception	2-18
Request In		1-7	Status Combinations	2-18
Address Out		1-7	Sense Information	2-19
Select Out/Hold Out and Select			Sense Byte	2-20
In		1-8	Sense Conditions	2-20
Operational In		1-9	Command Reject	2-20
Address In		1-9	Intervention Required	2-20
Command Out		1-9	Bus-Out Check	2-20
Status In		1-10	Equipment Check	2-20
Service Out		1-10	Data Check	2-20
Service In		1-10	Overrun	2-21
Suppress Out		1-11	General System Considerations	2-21
Metering Controls		1-11	Interface-Timeout Considerations	2-21
Clock Out		1-11	Propagation of Select Out	2-21
Metering In		1-11	System Configuration	2-22
Metering Out		1-11	Offline/Online	2-22
Reserved Lines		1-11	Power Effects	2-23
Signal-Interlock Summary		1-11		
<b>CHAPTER 2. OPERATIONAL DESCRIPTION</b>		2-1		3-1
Interface Sequences		2-1	CHAPTER 3. FEATURES	3-1
Selection		2-1	Bus-Extension Feature	3-1
Initial-Selection Sequence		2-1	Mark-Out Lines	3-1
Short-Busy Sequence		2-1	Mark-In Lines	3-1
Control-Unit-Initiated Sequence		2-2	Early Data-Bus-Width Indication	3-2
Data Transfer		2-2	I/O-Error-Alert Feature	3-2
Data-Transfer Sequence		2-2	Command-Retry Feature	3-2
I/O-Interface Connection		2-2	Command-Retry Sequence	3-2
Ending Sequence		2-3	High-Speed-Transfer Feature	3-2
Control-Unit Types		2-4	Data In	3-3
Type-1 Control Unit		2-4	Data Out	3-3
Type-2 Control Unit		2-4	Data-Streaming Feature	3-3
Type-3 Control Unit		2-4	Data Transfer While Data	
Addressing		2-4	Streaming	3-4
Address Assignment		2-4	Stop/Command Out While Data	
Address Decoding		2-5	Streaming	3-5
Commands		2-6	Suppress Data While Data	
Basic Operations		2-6	Streaming	3-6
Read		2-6	Interface Disconnect While Data	
Read Backward		2-7	Streaming	3-6
Write		2-7	Selective Reset While Data	
Control		2-7	Streaming	3-6
Sense		2-7	Response-Time Requirements While	
Test		2-8	Data Streaming	3-6
Sequence Controls		2-8	Dynamic-Reconnection Feature	3-7
Proceed		2-8		
Stop		2-8	APPENDIX A. ELECTRICAL	
Stack Status		2-9	SPECIFICATIONS	A-1
Suppress Data		2-9	Physical Requirements	A-1
Accept Data		2-9	Multiple Drivers and Receivers	A-1
Data Ready		2-9	General Electrical Requirements	A-1
Suppress Status		2-10	Voltage Levels	A-1
Accept Status		2-10	Cable	A-1
Command Chaining		2-10	Terminating Networks	A-1
Interface Disconnect		2-11	Ground Shift and Noise	A-1
Selective Reset		2-12	Interface-Circuit Requirements	A-2
System Reset		2-12	Receivers	A-2
Status Information		2-12	Drivers	A-2
Status Byte		2-13	Fault Conditions	A-3
Unit-Status Conditions		2-13	Circuits	A-3
Attention		2-13	Electrical Specifications for	
			Select-Out Circuitry	A-3
			General	A-3

Receiver . . . . .	A-4
Driver . . . . .	A-4
Terminator . . . . .	A-4
Measuring I/O-Interface Timings . . . . .	A-4
Uptime . . . . .	A-4
Downtime . . . . .	A-4
Valid Data . . . . .	A-5
Overlap Not Greater Than . . . . .	A-5
100-Nanosecond Delay . . . . .	A-5
Interface-Connector Pin Assignments . . . . .	A-5

APPENDIX B. SIGNAL CABLING AND CONNECTORS . . . . .	B-1
Cabling . . . . .	B-1
Cable Halves . . . . .	B-1
Mounting . . . . .	B-1

Offline Utilization . . . . .	B-1
Connectors . . . . .	B-3
Connector Blocks . . . . .	B-3
I/O Cable Capacity . . . . .	B-3
Serpentine Contacts . . . . .	B-3
Description . . . . .	B-3
Wire Termination . . . . .	B-4
Electrical Specifications . . . . .	B-5
Terminators . . . . .	B-6

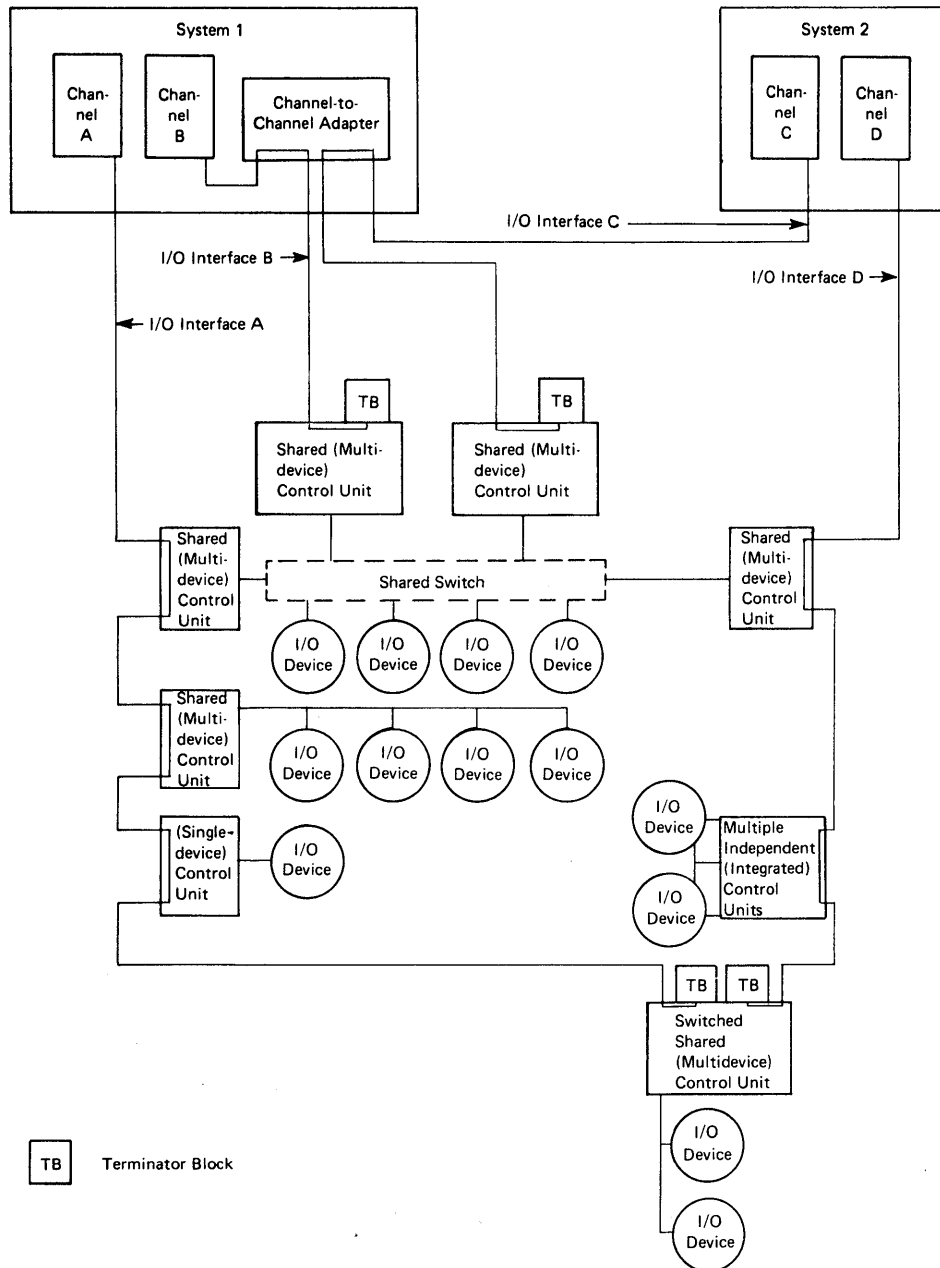
APPENDIX C. SUPPLEMENTARY APPLICATION INFORMATION . . . . .	C-1
---	-----

APPENDIX D. SUMMARY INFORMATION . . . . .	D-1
---	-----

INDEX . . . . .	X-1
-----------------	-----

Input/output (I/O) devices are used to provide external storage, to communicate between data-processing systems, and to communicate between a data-processing system and the external world.

I/O devices are attached to central processing units (CPUs) by means of control units and channels in various configurations (Figure 1-1).



TB Terminator Block

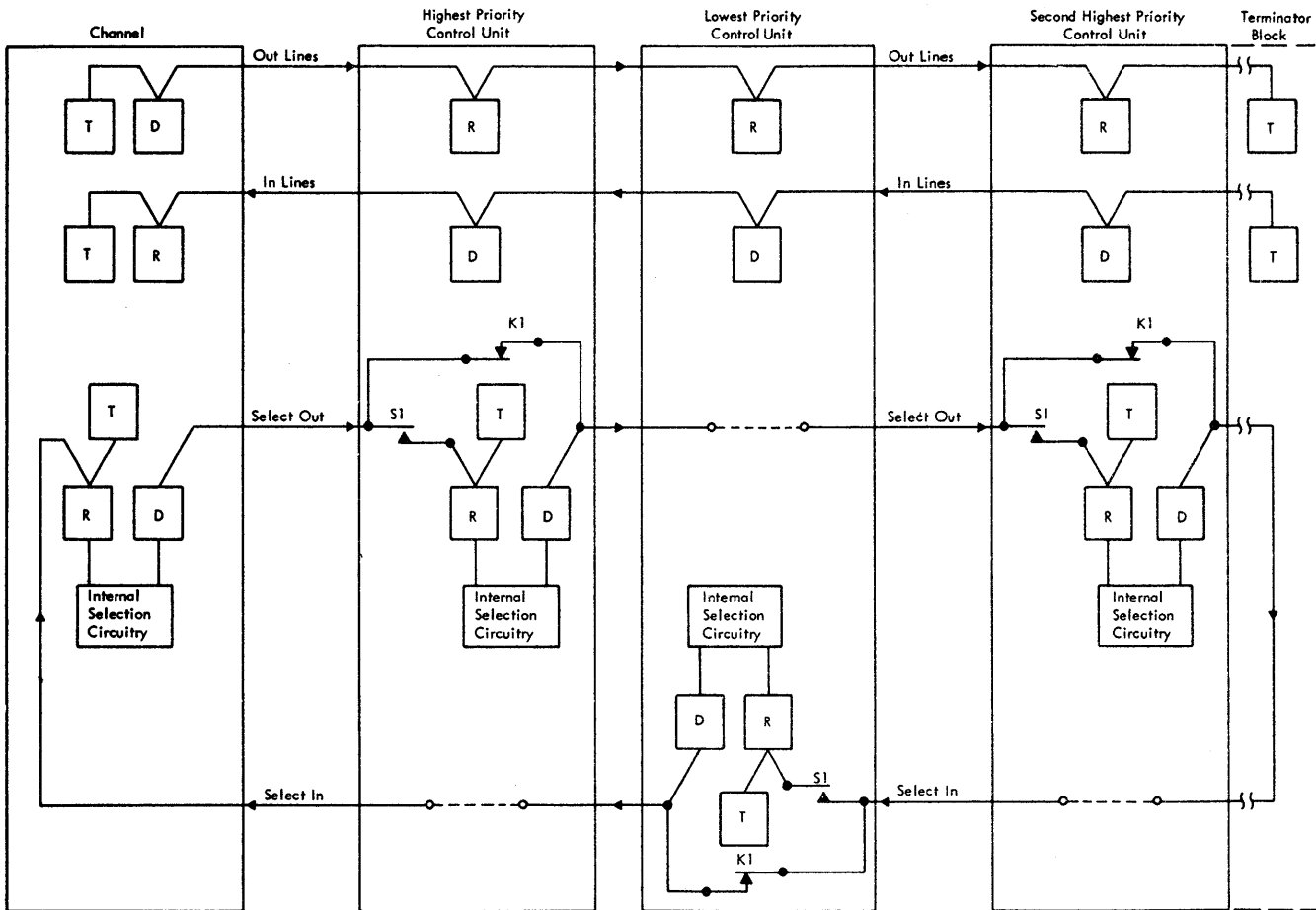
Figure 1-1. I/O Interface -- Example of Multiple-Channel Configurations

The channel directs the flow of information between control units and main storage.

The control unit provides the logical capability necessary to operate and control an I/O device, and it adapts the characteristics of each I/O device to the standard form of control provided by the channel. A control unit may be housed separately or it may be physically and logically integrated with the I/O device.

An I/O device attached to the control unit may be designed to perform only certain limited operations, such as recording data and moving the recording medium. Regardless of the design, the I/O device needs detailed signal sequences peculiar to that type of I/O device to accomplish these functions. The control unit decodes commands received from the channel, interprets them for the I/O device, provides the signal sequence for executing the operations, and reports the status of operations to the channel.

One method of connection between a channel and a control unit is called the I/O interface. Each I/O interface, which consists of parallel signal lines that connect a number of control units to a channel (Figure 1-1), provides a common signaling protocol and information format to all control units that connect to the channel. All signals, of the I/O interface, with the exception of selection signals, are available to all control units attached to that I/O interface at the same time (Figure 1-2). However, only one control unit can be connected logically to each I/O interface at any one time. Selection of a control unit by the channel is controlled by a signal passing serially through all control units attached to one I/O interface. This serial passing of the selection signal permits each control unit, in turn, to respond to the other signals provided by the channel. Once the control unit is selected, it remains logically connected to the I/O interface until it completes the transfer of the information it has or needs, or until the channel signals the control unit to disconnect.



D Driver  
R Receiver  
T Terminator

Figure 1-2. Interconnections on the I/O Interface



The I/O-interface addressing facilities can accommodate up to 256 directly addressable I/O devices. However, within the given address limitations, and because of timing and electrical considerations, the number of control unit attachment points that can be accommodated is generally limited to eight. One attachment point may be associated with a (single-device) control unit, a shared (multidevice) control unit, or multiple independent (integrated) control units.

The multiplexing facilities of the interface permit operating a number of the 256 I/O devices concurrently on a single I/O interface; portions of various messages can be transmitted over the I/O interface in an interleaved fashion to or from different I/O devices, or a complete message can be transmitted in a single I/O-interface operation. The operation actually performed is determined by the particular command, channel, and control unit.

The rise and fall of all signals transmitted over the I/O interface are generally controlled by interlocked responses. The interlocked sequences remove the dependence of the I/O interface on circuit speed and make the I/O

interface applicable to a wide variety of circuits and data rates. Furthermore, interlocking permits connecting control units of different circuit speeds to a single channel. Those sequences that are not interlocked allow for designs capable of achieving higher data rates than those achievable through interlocked sequences. However, the electrical specifications of circuits not using interlocked sequences are more restrictive and do not allow as wide a range of possible circuit speeds.

#### LINE DEFINITION

The I/O interface connects a channel with control units. External cables physically connect all control units in a chain, with only the first control unit directly cabled to the channel. See Figures 1-1 and 1-2.

#### I/O-INTERFACE LINES

The I/O-interface lines and their uses are described in Figure 1-3.

Line Name	Abbreviation	Uses
Bus 0 out position P Bus 0 out position 0 Bus 0 out position 1 Bus 0 out position 2 Bus 0 out position 3 Bus 0 out position 4 Bus 0 out position 5 Bus 0 out position 6 Bus 0 out position 7	Bus 0 out P Bus 0 out 0 Bus 0 out 1 Bus 0 out 2 Bus 0 out 3 Bus 0 out 4 Bus 0 out 5 Bus 0 out 6 Bus 0 out 7	The 'bus out' lines are used to transmit information (data, I/O-device address, commands, control information) from the channel to the control unit. (Bus 1 is only available with the bus-extension feature.)
Bus 1 out position P Bus 1 out position 0 Bus 1 out position 1 Bus 1 out position 2 Bus 1 out position 3 Bus 1 out position 4 Bus 1 out position 5 Bus 1 out position 6 Bus 1 out position 7	Bus 1 out P Bus 1 out 0 Bus 1 out 1 Bus 1 out 2 Bus 1 out 3 Bus 1 out 4 Bus 1 out 5 Bus 1 out 6 Bus 1 out 7	
Bus 0 in position P Bus 0 in position 0 Bus 0 in position 1 Bus 0 in position 2 Bus 0 in position 3 Bus 0 in position 4 Bus 0 in position 5 Bus 0 in position 6 Bus 0 in position 7	Bus 0 in P Bus 0 in 0 Bus 0 in 1 Bus 0 in 2 Bus 0 in 3 Bus 0 in 4 Bus 0 in 5 Bus 0 in 6 Bus 0 in 7	The 'bus in' lines are used to transmit information (data, selected I/O-device address, status information, sense information) from the control unit to the channel. (Bus 1 is only available with the bus-extension feature.)
Bus 1 in position P Bus 1 in position 0 Bus 1 in position 1 Bus 1 in position 2 Bus 1 in position 3 Bus 1 in position 4 Bus 1 in position 5 Bus 1 in position 6 Bus 1 in position 7	Bus 1 in P Bus 1 in 0 Bus 1 in 1 Bus 1 in 2 Bus 1 in 3 Bus 1 in 4 Bus 1 in 5 Bus 1 in 6 Bus 1 in 7	
Mark 0 in Mark 0 out Mark 1 in Mark 1 out Mark in parity Mark out parity	Mk 0 in Mk 0 out Mk 1 in Mk 1 out Mk in P Mk out P	The mark lines are used to indicate the buses being used. (Except for 'mark 0 in', these lines are only available with the bus-extension feature. 'Mark 0 in' is also available with the command-retry feature.)
Address out Address in Command out Status in Service out Service in Data out Data in	Adr out Adr in Cmd out Sta in Srv out Srv in Dat out Dat in	The tag lines are used for interlocking and controlling information on the buses and for special sequences. ('Data out' and 'data in' are only available with the high-speed-transfer or data-streaming features.)

Figure 1-3 (Part 1 of 2). I/O-Interface Lines

Line Name	Abbreviation	Uses
Operational out Operational in Hold out Select out Select in Suppress out Request in Disconnect in	Opl out Opl in Hld out Sel out Sel in Sup out Req in Dis in	The selection-control lines are used for the scanning or selection of attached I/O devices. ('Disconnect in' is only available with the I/O-error-alert feature.)
Metering out Metering in Clock out	Mtr out Mtr in Clk out	These three lines are no longer used.

Figure 1-3 (Part 2 of 2). I/O-Interface Lines

The remainder of this chapter discusses those lines that are available with the basic I/O interface. For information concerning lines that are only provided when features are implemented, see Chapter 3, "Features."

**Note:** Except when the data-streaming feature is used, the validity of information on the buses and the timing of the signals on the tag lines are specified for measurement at the channel cable connectors.

#### BUSES -- GENERAL

Each bus is a set of nine lines, consisting of eight information lines

and one parity line. Information on the buses is arranged so that bit position 7 of a bus always carries the rightmost bit within an eight-bit byte. The leftmost bit is in position 0, and the intervening bits are in order from position 1 to position 6.

When a byte transmitted over the interface consists of less than eight information bits, the bits are placed contiguously in the highest-numbered bit positions of the bus. Any unused lines of the bus include the low-numbered positions (position 0 and adjacent positions). Unused lines present logical zeros to the receiving end. The parity bit of any byte appears in the parity position (P). The byte always has odd parity (Figure 1-4).

BUS LINE	BCD (POSITION VALUE)	PACKED NUMERIC (POSITION VALUE)	UNPACKED NUMERIC (POSITION VALUE)	EBCDIC (BIT POSITIONS)	BINARY (POSITION VALUE)
P	P	P	P	P	P
0	(unused)	8	0	0	128
1	(unused)	4	0	1	64
2	B	2	0	2	32
3	A	1	0	3	16
4	8	8	8	4	8
5	4	4	4	5	4
6	2	2	2	6	2
7	1	1	1	7	1

X = Higher order digit  
X + 1 = Lower order digit  
(unused) = Logical Zero

Figure 1-4. Organization of Information on the Bus

### Bus Out

'Bus out' is used to transmit addresses, commands, and data to control units. The type of information transmitted over 'bus out' is indicated by the outbound tag lines:

1. Except for interface disconnect, when 'address out' is up during the initial-selection sequence, 'bus out' specifies the address of the I/O device with which the channel wants to communicate. (See "Interface Disconnect" in Chapter 2, "Operational Description.")
2. When 'command out' is up in response to 'address in' during the initial-selection sequence, 'bus out' specifies a command.
3. When 'service out' is up in response to 'service in' (or 'data out' is up in response to 'data in') during the execution of a write or control command, 'bus out' specifies a data byte that is being transferred between the channel and the control unit. For example, during an operation that is specified by a write command, 'bus out' contains data to be recorded by the I/O device. During an operation that is specified by a control command, 'bus out' may contain either data that describes in more detail how the operation is to be executed or a second-level address within the control unit or I/O device.

**Note:** When data streaming occurs, the following paragraph does not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

The period during which information on 'bus out' is valid is controlled by the tag lines. During transmission of the I/O-device address during the initial-selection sequence, information on 'bus

out' must be valid from the rise of 'address out' until the rise of 'operational in' or 'select in'. In the short-busy sequence, the I/O-device address on the bus-out line must be valid from the rise of 'address out' until the fall of 'status in'. When the channel is transmitting any other type of information, the information on 'bus out' is valid from the rise of the signal on the associated outbound tag line until the fall of the signal on the responding inbound tag line.

Some skew on 'bus out' is accommodated by the channel. The channel provides a delay that accommodates skew caused by its own circuitry and, in addition, provides a delay of at least 100 nanoseconds. This delay compensates for skew caused by the cable and, for most control units, also is sufficient to accommodate the skew caused by the I/O-interface receivers. (Except as noted under "Address Out" in this chapter, the channel delays raising of the signal on the outbound tag lines so that the information on 'bus out' precedes the signal on the outbound tag line by at least 100 nanoseconds. This delay is measured at the channel cable connectors and under the worst-case skew conditions.) The control unit, when it can cause more skew, provides additional delay to compensate for the skew.

### Bus In

'Bus in' is used to transmit addresses, status, and data to the channel. A control unit can place and maintain information on 'bus in' only when its 'operational in' is up, except in the short-busy sequence. (See "Short-Busy Sequence" in Chapter 2, "Operational Description.")

The type of information transmitted over 'bus in' is indicated by the inbound tag lines:

1. When 'address in' is up, 'bus in' specifies the address of the currently selected I/O device.
2. When 'status in' is up, 'bus in' contains a byte of information that describes the status of the I/O device or control unit.
3. When 'service in' (or 'data in') is up during execution of a read, read-backward, or sense command, the nature of the information contained on 'bus in' depends on the type of operation. During a read operation, it may contain a byte of data from the record medium. During a basic sense operation, 'bus in' contains data describing unusual conditions detected at the I/O device.

**Note:** When data streaming occurs, the following paragraph does not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

The period during which information on 'bus in' is valid is controlled by the tag lines. Information on the bus is valid within 100 nanoseconds after the rise of the associated inbound tag and remains valid until the rise of the responding outbound tag or, in a short-busy sequence, until the fall of 'select out'. The 100-nanosecond delay between the rise of the inbound tag and the time the signal becomes valid on 'bus in' places the responsibility on the channel for deskewing 'bus in'. The channel provides a delay in the inbound tag lines to accommodate skew caused by the channel circuitry (including its receivers) and, in addition, provides a delay of at least 100 nanoseconds. This delay compensates for skew caused by the cable, and, for most control units, for the skew caused by their drivers. This delay provides sufficient time to deskew the information so that the inbound tag can be raised by the control unit at the same time information is placed on the bus. When a control unit and cable can cause more skew than can be accommodated by a 100-nanosecond delay, the control unit provides the additional delay to eliminate this greater skew.

## SELECTION CONTROLS AND TAG LINES

### Operational Out

'Operational out' is a line from the channel to all attached control units and is used for interlocking purposes. Except for 'suppress out', all lines from the channel are significant only when 'operational out' is up. Whenever 'operational out' is down, all inbound

lines from the control unit drop, and any operation currently in process over the interface is reset. Under these conditions, all control-unit-generated interface signals are down within 1.5 microseconds after the fall of 'operational out' at the control unit. (See "Selective Reset" and "System Reset" in Chapter 2, "Operational Description.")

### Request In

'Request in' is a line from all attached control units to the channel. This line, when raised, indicates that the control unit requires service and is requesting a selection sequence.

'Request in' is dropped when:

1. 'Operational in' rises, unless additional control-unit-initiated sequences are required, or
2. The control unit is no longer ready to present the status information or data, or
3. The selection requirement is satisfied by another path.

'Request in' never falls later than 250 nanoseconds after the fall of 'operational in' if the sequence satisfies the service requirements of the control unit.

'Request in' does not remain up when 'suppress out' is up if the request is for presentation of suppressible status. (See "Suppress Status" in Chapter 2, "Operational Description.") When the control unit is requesting a selection sequence in order to present suppressible status, 'request in' falls at the control unit within 1.5 microseconds after the rise of 'suppress out' at the control unit.

'Request in' can be signaled by more than one control unit at a time.

### Address Out

'Address out' is a tag line from the channel to all attached control units. 'Address out' is used to signal all the control units to decode the I/O-device address on 'bus out'. If the control unit recognizes the address, it responds by raising 'operational in' when 'select out' (or 'hold out') rises with 'address out' still up (except in the short-busy sequence). (See "Select Out/Hold Out and Select In.") 'Address out' rises at least 250 nanoseconds after the I/O-device address is placed on 'bus out' or at least 250 nanoseconds after the rise

of 'operational out', whichever occurs later. 'Address out' is down for at least 250 nanoseconds before its rise for I/O-device selection. If 'address out' falls before 'select out' rises, the I/O-device selection is canceled.

'Address out' can rise for device selection only when 'select out' (or 'hold out'), 'select in', 'status in', and 'operational in' are down at the channel. Ultimate use of the I/O-device address on 'bus out' at the control unit is timed by the next rise of 'select out' (or 'hold out') at the addressed control unit. Once 'address out' and 'select out' (or 'hold out') are up, 'address out' stays up until either 'select in' or 'operational in' rises or, in the short-busy sequence, until 'status in' falls. Except when interface disconnect is being signaled, 'address out' cannot be up concurrently with any other outbound tag line.

'Address out' is also used in the interface-disconnect sequence control. (See "Interface Disconnect" in Chapter 2, "Operational Description.")

#### Select Out/Hold Out and Select In

Control-unit selection is controlled by 'select out', 'select in', and 'hold out'. 'Select out' and 'select in' form a loop from the channel through each control unit to the cable terminator block ('select out') and again through each control unit back to the channel ('select in'). Control-unit-selection circuitry may be attached to either 'select out' or 'select in'. In this manual, the selection circuitry of all control units is assumed to be attached to 'select out'. All discussions that apply to the selection logic of control units attached to 'select out' equally apply in cases where the control unit is attached to 'select in'. The selection priority is (1) all control units with selection circuitry attached to 'select out', in order of attachment from the channel to the cable terminator block, followed by (2) all control units with selection circuitry attached to 'select in', in order of attachment from the cable terminator block to the channel. (See Figure 1-2.) If selection is not required, the selection signal is in turn propagated by each control unit to the next control unit on the line.

Each control unit ensures that the process of electrically bypassing 'select out' before power changes does not interfere with the propagation of 'select out'. Thus, 'select out' discontinuities, which may occur when another control unit on the I/O interface is powered up or down, do not affect the propagation of 'select out'.

This protection should be accomplished by the use of a special latch circuit. The latch is turned on by the AND of 'select out' and 'hold out' and is reset by the fall of 'hold out'. The circuit is in series with the remaining selection circuitry in the control unit and provides a constant 'select out' within the control unit -- and therefore to the following control unit -- regardless of variations in the input 'select out' signal. (See Figure 2-1.)

Throughout the following description, 'select out' assumes the latch operates properly, that is, the rise of 'select out' at the control unit assumes that 'hold out' is up and that the fall of 'select out' is a result of the fall of 'hold out'.

A control unit can raise its 'operational in' only at the rise of its incoming 'select out'. Once a control unit propagates 'select out', it cannot raise its 'operational in' or respond with a short-busy sequence until the next rise of the incoming 'select out'.

When an operation is being initiated by the channel, 'select out' is raised not less than 400 nanoseconds after the rise of 'address out', which indicates the address of the device being selected.

The channel keeps 'select out' up until 'select in' rises, or at least until 'address in' and 'operational in' have both risen, or in the short-busy sequence, until 'status in' rises.

When 'select in' rises, 'select out' drops and does not rise again until after 'select in' falls.

A control unit becomes selected only when it raises its 'operational in'. After the drop of 'select out', the control unit keeps 'operational in' up until the current signal sequence is completed. If a control unit raises 'operational in', it suppresses the propagation of 'select out' to the next control unit. If the control unit does not require selection, it propagates 'select out' to the next control unit within 1.8 microseconds. (See "Interface Timeout Considerations" in Chapter 2, "Operational Description.")

When 'status in' rises in response to 'select out' in the short-busy sequence, 'select out' drops and does not rise until 'address out' has dropped.

'Hold out' is a line from the channel to all attached control units and is used in conjunction with 'select out' to synchronize control-unit selection.

'Hold out' is also used to minimize the propagation of the fall of 'select out' by purging the 'select out' signal from the 'select out' signal path.

Therefore, once 'hold out' drops, it does not rise for at least 4 microseconds in general system configurations. The minimum downtime of this signal may be optionally adjusted at installation time to a minimum of 2 microseconds to handle high-speed channel configurations. In all cases, the channel is capable of providing the 4-microsecond timing for general system configurations. (See "System Configuration" in Chapter 2, "Operational Description.")

To prevent overlapping of interface sequences, one of the following procedures is performed:

1. 'Select out' is not raised until all inbound signals for the preceding sequence are in a down state.
2. In tags are not considered valid until 1.5 microseconds after the fall of 'operational in' for the preceding sequence.

'Select in' is a line that extends the select out signal from the jumper in the cable terminator block to the channel (see Figure 1-2). It provides a return path to the channel for the 'select out' signal.

### Operational In

'Operational in' is a line from all attached control units to the channel and is used to signal the channel that an I/O device has been selected. 'Operational in' stays up for the duration of the selection. The selected I/O device is identified by the address byte transmitted over 'bus in' when 'address in' is raised.

The rise of 'operational in' indicates that an I/O device is selected and the control unit is actively communicating with the channel. 'Operational in' rises only when the incoming 'select out' to the control unit is up and the outgoing 'select out' is down; that is, the control unit raises 'operational in' only in response to the rise of 'select out' and blocks the 'select out' signal from being propagated to the next control unit. 'Operational in' drops only after 'select out' drops.

When 'operational in' is raised for a particular signal sequence, it stays up until 'select out' is down and all required information has been transmitted between the channel and the control unit. When 'select out' is down, 'oper-

ational in' drops either (1) after the rise of the outbound tag associated with the transfer of the last byte of information or (2) immediately if there is no further required information to be transmitted between the channel and the control unit. With the exception of 'request in' or 'metering in', all inbound signals are down within 100 nanoseconds after the fall of 'operational in' at the control unit.<sup>1</sup>

### Address In

'Address in' is a tag line from all attached control units to the channel and is used to signal the channel when the address of the currently selected I/O device has been placed on 'bus in'. During an initial-selection sequence or a control-unit-initiated sequence, the channel responds to 'address in' by raising 'command out'. 'Address in' stays up until the rise of 'command out'. 'Address in' must fall so that 'command out' may fall. 'Address in' is not up concurrently with any other inbound tag line.

### Command Out

'Command out' is a tag line from the channel to all attached control units and is used to signal the selected I/O device in response to a signal on 'address in', 'status in', 'data in', or 'service in'. The rise of 'command out' indicates that any information on 'bus in' is no longer required to be valid. 'Command out' stays up until the fall of the associated 'address in', 'status in', 'data in', or 'service in'. However, when the data-streaming feature is used, 'command out' is not interlocked with 'service in' or 'data in'. (See "Data-Streaming Feature" in Chapter 3, "Features.") 'Command out' is not up concurrently with any other outbound tag line, except during an interface-disconnect sequence, when 'address out' may be up. (See "Interface Disconnect" in Chapter 2, "Operational Description.")

During an initial-selection sequence, 'command out' rising in response to the rise of 'address in' indicates to the selected I/O device that the channel has placed a command byte on 'bus out'. (The command byte has a fixed format. See "Commands" in Chapter 2, "Operational Description.") 'Command out' in response to 'data in' or 'service in'

<sup>1</sup> For control units and devices designed before January 1, 1977, all inbound signals except 'request in' and 'metering in' are down within 1.5 microseconds after the fall of 'operational in' at the control unit.

always means stop. (See "Stop" in Chapter 2, "Operational Description.") During a control-unit initiated sequence, 'command out' in response to 'address in' means proceed. (See "Proceed" in Chapter 2.) 'Command out' in response to 'status in' means stack. (See "Stack Status" in Chapter 2.)

When 'command out' is raised to indicate proceed, stop, or stack, 'bus out' has a byte of all zeros but does not necessarily have correct parity. 'Bus out' is not checked for parity or decoded as a command by the control unit under these circumstances.

### Status In

'Status in' is a tag line from all attached control units to the channel and is used to signal the channel when the selected control unit has placed status information on 'bus in'. The status byte has a fixed format and contains bits describing the current status at the control unit. (See "Status Byte" in Chapter 2, "Operational Description.") The channel responds by raising either 'service out' or 'command out' or, in the case of the short-busy sequence, by dropping 'select out'.

'Status in' is not up concurrently with any other inbound tag line, and does not rise if any out tag is up, except 'address out' in the short-busy sequence. 'Status in' stays up until the rise of an out tag or, in the short-busy sequence, until 'select out' falls. 'Status in' must fall so that the responding out tag may fall. In the short-busy sequence, status information on 'bus in' is valid until 'select out' (or 'hold out') falls.

### Service Out

'Service out' is a tag line from the channel to all attached control units and is raised to signal the selected I/O device when 'service in' or 'status in' has been recognized. A signal on 'service out' indicates to the selected I/O device that the channel has accepted the information on 'bus in', or has provided on 'bus out' the data requested by 'service in'.

**Note:** When data streaming occurs, the following two paragraphs do not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

When 'service out' rises in response either to 'service in' during a read, read-backward, or sense operation or to

'status in', the information placed on 'bus in' by the control unit has been accepted by the channel. In these cases, the rise of 'service out' indicates that the information on 'bus in' is no longer required to be valid and is not associated with any information on 'bus out'. When 'service out' rises in response to 'service in' during a write or control operation, the rise of 'service out' indicates that the channel has placed the requested information on 'bus out'.

'Service out' stays up until the fall of the associated 'service in' or 'status in'. 'Service out' is not up concurrently with any other out tag except during an interface-disconnect sequence, when 'address out' may be up. (With the high-speed transfer feature, 'data out' and 'service out' may be up concurrently.)

### Service In

'Service in' is a tag line from all attached control units to the channel and is used to signal the channel when the selected I/O device is ready to send or receive a byte of information. The nature of the information associated with 'service in' depends on the operation and the I/O device. The channel responds to the rise of 'service in' by raising either 'service out' or 'command out'.

**Note:** When data streaming occurs, the following two paragraphs do not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

During read, read-backward, and sense operations, 'service in' rises when information is available on 'bus in'. During execution of operations specified by either a write or control command, 'service in' rises when information is required on 'bus out'. 'Service in' is not up concurrently with any other inbound tag line. 'Service in' stays up until the rise of either 'service out', 'command out', or 'address out'. (With the high-speed transfer feature, 'data in' and 'service in' may be up concurrently.)

When, in the case of cyclic I/O devices, the channel does not respond in time to the preceding 'service in', an overrun condition may be recognized by the control unit or I/O device. In this situation, data transfer is terminated, and the unit-check status indicator and the overrun sense indicator are set to ones. When this condition is recognized, 'service in' does not drop if an out tag has not risen and does not rise if 'service out' has not dropped.



## Suppress Out

'Suppress out' is a line from the channel to all attached control units; it may rise or fall at any time. This line is used both alone and in conjunction with the out-tag lines to provide the following special functions: suppress data, suppress status, command chaining, and selective reset. Each of these functions is described in Chapter 2, "Operational Description."

## METERING CONTROLS

### Clock Out

The 'clock out' line is not used; the channel ensures that the 'clock out' line remains down at all times.<sup>2</sup>

### Metering In

The 'metering in' line is not used; each control unit ensures that the 'metering in' line is not raised.<sup>3</sup>

### Metering Out

The 'metering out' line is not used; the channel ensures that the 'metering out' line remains down at all times.<sup>4</sup>

## RESERVED LINES

Some of the signal lines in the I/O interface are reserved. (See

"Interface-Connector Pin Assignments" in Appendix A.)

All control units carry through all 48 signal transmission lines in cables 1 and 2, with the exception of 'select out' (or 'select in' if low priority), from the IN cable connector to the OUT cable connector.<sup>5</sup> All control units implementing the bus-extension feature carry through all 72 signal transmission lines in cables 1, 2, and 3, except for 'select out' (or 'select in' if low priority). (See "Interface-Connector Pin Assignments" in Appendix A.)

## SIGNAL-INTERLOCK SUMMARY

The following rules for interlocking of signals are followed in the design of channels and control units using this I/O interface. These rules describe the protocols to be followed by channels and control units in the absence of error conditions. That is, in situations where timing restrictions are specified in this manual and either the channel or control unit has failed to respond by dropping or raising the appropriate signal line within the required time interval, then it is assumed that the interlocking rules do not apply.

Note: When either high-speed transfer or data streaming occur, rules 1, 2, and 3 do not apply.

1. Except when interface disconnect is signaled, no more than one out tag is up at any given time. During interface disconnect, 'address out' may be up with another out tag.
2. No more than one in tag is up at any given time.
3. An in tag rises only when all out tags are down, except in the short-busy sequence.

<sup>2</sup> For all control units designed before January 1, 1977, the 'clock out' line from the channel to these control units is used to provide the CPU interlock control for changing the enable/disable states of the units (the signal must be down to permit changing states). In addition, the control-unit transition between the enabled and disabled state requires the same conditions as those that prevail for the offline/online transition. The down state of 'clock out' is at least 1 microsecond in duration. (See "Offline/Online" in Chapter 2, "Operational Description.")

<sup>3</sup> For some control units designed before January 1, 1986, the 'metering in' line from the control unit to the channel is used to indicate control unit and device activity. 'Metering in' can be signaled by more than one control unit at a time. The use of 'metering in' is model-dependent for both control units and channels.

<sup>4</sup> For some channels designed before January 1, 1986, the 'metering out' line from the channel to the control unit is used to condition usage meters which may be present on control units and I/O devices. The use of 'metering out' is model-dependent for both channels and control units.

<sup>5</sup> Control units and devices designed before October 2, 1970, carry through the 40 assigned signal transmission lines in cables 1 and 2, except for 'select out' (or 'select in' if low priority), from the IN cable connector to the OUT cable connector.

Note: When data streaming occurs, rules 4 and 5 do not apply.

4. An in tag falls only after the rise of a responding out tag, except for 'status in' in the short-busy sequence.
5. 'Service out' and 'command out' rise only in response to the rise of an in tag.
6. 'Address out' for an initial-selection sequence rises when 'select in', 'operational in', 'status in', and 'select out' are down at the channel.
7. Once 'address out' and 'select out' have risen for an initial-selection sequence, 'address out' stays up until after the rise of 'select in' or 'operational in', or (for the short-busy sequence) the fall of 'status in'.
8. Once 'address out' has risen for the interface-disconnect sequence, it does not drop until 'operational in' drops.
9. None of the out lines, except 'suppress out', have meaning when 'operational out' is down.
10. 'Select out' rises only if 'operational in' and 'select in' are down.

11. 'Operational in' does not fall until one of the following events occurs:
  - a. 'Select out' falls, and an out-tag response is sent for the last in tag of any given signal sequence.
  - b. 'Operational out' falls.
  - c. An interface-disconnect sequence occurs.
12. 'Operational in' does not rise unless 'operational out' is up and, if currently up, drops when 'operational out' drops, or when the interface-disconnect sequence occurs.

Note: Designers should carefully consider the effects of interface signal-transition times. Although transition time should not generally be a problem, some cases may exist, because of wide variations in circuit tolerance or in the physical integration of channel and control units, in which the transition time must be considered. Because signaling on the I/O interface usually causes the receiving unit to signal a response, the general rule is that a unit should not signal a response until it has fully recognized internally the receipt of any previously transmitted signal.

This chapter discusses the operation of the basic I/O interface. For information concerning functions that are only provided when features are implemented, see Chapter 3, "Features."

## INTERFACE SEQUENCES

During the execution of an operation involving the channel and control unit, the selection, data transfer, and ending sequences can occur. These sequences are defined in this section. (For examples of these sequences, see the flow diagrams, Figures C-1 through C-8, and the sequence charts, Figures C-9 through C-13, in Appendix C. These diagrams should be used as examples only and should not be considered as containing precise definitions of interface sequences.)

### SELECTION

#### Initial-Selection Sequence

To initiate an I/O operation, the channel places the address of the desired I/O device on 'bus out' and raises 'address out'. Each control unit connected to the channel attempts to decode the address on 'bus out'; however, only one control unit should be assigned to a given I/O address. To be acceptable, the address must have correct parity.

The channel then raises 'select out', and the incoming 'select out' signal appears at the control unit for the addressed I/O device. If the control unit and the addressed I/O device are available to execute the operation, the control unit blocks the propagation of the 'select out' signal and raises the 'operational in' line. When 'operational in' rises, the channel responds by dropping 'address out'. The control unit places the address of the I/O device on 'bus in' and, after 'address out' falls, raises 'address in'. 'Hold out' with 'select out' may drop any time after 'address in' rises. After the channel checks the I/O-device address on 'bus in', it responds by placing the command on 'bus out' and raises 'command out'. The selected control unit processes the command and drops 'address in', which allows 'command out' to fall. After 'command out' falls, the control

unit places status information on 'bus in' and raises 'status in'; this is referred to as initial status. At this time, the status is analyzed to determine if the command has been accepted. The command is considered to have been accepted if (1) command retry is requested, or (2) the command sent to the addressed I/O-device is not test I/O, and the status is zero, or (3) the command sent to the addressed I/O device is not test I/O, and the status contains channel end but does not contain attention, control-unit end, busy, unit check, or unit exception. Once the command is accepted, the I/O operation is considered to have been initiated.

If the channel accepts the initial status, it responds by raising 'service out', allowing the control unit to drop 'status in'. If the channel does not accept the initial status, it responds by raising 'command out', allowing the control unit to drop 'status in'. When 'status in' falls, in either case the initial-selection sequence is completed. (See "Stack Status" in this chapter.)

If, during the initial-selection sequence, the I/O device or the path to the I/O device is not available to execute the operation because it is currently being used to execute a previously initiated operation, or if the control unit is not available because it (1) is executing a previously initiated operation, or (2) has pending status for another I/O device, the control unit may signal that a busy condition exists. The control unit can indicate the busy condition in either of two ways, depending on the design of the control unit: it may continue execution of the initial-selection sequence until completion, or it may modify the sequence and cause execution of the short-busy sequence. If the control unit executes the initial-selection sequence, the busy condition is presented as initial status. If instead it executes the short-busy sequence, the busy condition is also presented. (See "Short-Busy Sequence," described next, and "Busy" in this chapter.)

**Note:** A channel response of 'command out' to 'status in' cannot prevent the execution of an immediate command.

#### Short-Busy Sequence

This sequence can be initiated during an initial-selection sequence when the

control unit, because of an existing condition, is unable to handle the initial-selection sequence. When this occurs, the control unit converts the ongoing initial-selection sequence to a short-busy sequence. Specifically, when the channel has raised 'select out' after having placed the I/O-device address on 'bus out' and raised 'address out', each control unit attempts to decode the address on 'bus out'. When 'select out' rises at the addressed control unit, the control unit blocks the propagation of 'select out', places the busy status byte on 'bus in', and raises 'status in'. 'Operational in' is not raised.

The channel signals that the status byte is no longer needed on 'bus in' by dropping 'select out'. The control unit responds by dropping 'status in' and disconnecting from the interface. The channel keeps 'address out' up until 'status in' drops, thus completing the short-busy sequence.

During execution of the short-busy sequence, the control unit presents status of either (1) busy and status modifier, (2) busy, status modifier, and control-unit end, or (3) busy. Presentation of any other status condition by the control unit or device may cause an error condition to be recognized.

Note: The short-busy sequence is not used in response to an initial-selection sequence addressed to a device for which chaining has just been indicated.

#### Control-Unit-Initiated Sequence

When a control unit requires service, it signals the channel by raising 'request in'. The next time 'select out' rises at the control unit requiring service and no selection is being attempted by the channel ('address out' down), the control unit blocks the propagation of 'select out', places the address of the I/O device on 'bus in', and raises both 'address in' and 'operational in'. When the channel recognizes the address, 'command out' is sent to the control unit, indicating proceed. When 'command out' rises, the control unit responds by dropping 'address in'. The fall of 'address in' enables the channel to respond by dropping 'command out', thus completing the control-unit-initiated sequence.

If the service request is for data, the sequence proceeds as described in "Data-Transfer Sequence."

If the service request is for status information, the sequence proceeds as described in "Ending Sequence" in this chapter.

## DATA TRANSFER

### Data-Transfer Sequence

Data transfer may be requested by a control unit after an initial-selection sequence is executed. The direction of data transfer over the I/O interface is determined by the command that was passed to the control unit during that sequence.

Note: When data streaming occurs, the following two paragraphs do not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

To transmit data to the channel, the control unit places a data byte on 'bus in' and raises 'service in'; the tag and the validity of 'bus in' are maintained until the appropriate outbound tag is raised in response. When 'service out' rises, the control unit responds by dropping 'service in'. After 'service in' falls, the channel responds by dropping 'service out', thus completing the data-transfer sequence.

To request data from the channel, 'service in' is raised, and the channel places the data on 'bus out' and signals the control unit by raising 'service out'. When 'service out' rises, the control unit responds by dropping 'service in'. The channel maintains the validity of 'bus out' until 'service in' falls. After 'service in' falls, the channel responds by dropping 'service out', thus completing the data-transfer sequence.

After selection, as a result of either an initial-selection sequence or a control-unit-initiated sequence, the control unit may remain connected to the channel for the duration of the transfer of information. The transferred information can be a single byte of data, a byte of status, a new command, a string of data bytes, or, in the case of an immediate operation, the specification of a complete operation from initiation to reception of ending status.

### I/O-Interface Connection

A connection begins at the time 'select out' rises at the control unit for the purpose of executing any sequence or sequences. The duration of the connection is under control of both the channel and the control unit. To provide a channel with a method of controlling the duration of the

connection, a control unit does not disconnect from the I/O interface before 'select out' ('hold out') falls. However, the control unit may preserve the connection after the channel permits the control unit to disconnect -- 'select out' (or 'hold out') down -- by holding up 'operational in'. In this manner, the control unit can extend the duration of the connection. The connection is considered to be ended when 'operational in' is dropped.

Depending on the duration of the connection, one of two modes is established: byte-multiplex or burst. (These modes are established to assist in the scheduling of concurrent execution of multiple I/O operations.)

**Byte-Multiplex Mode:** A byte-multiplex-mode connection is defined as any connection where the time contributed by the control unit during the connection (because of the control-unit circuitry or the sequencing method used) is equal to or less than 32 microseconds. The connection may occur for the purpose of initiating or continuing execution of an I/O operation or presenting status. The byte-multiplex mode is the normal mode for low-speed I/O devices; however, all I/O devices are designed to operate in burst mode when required by the channel.

**Burst Mode:** A burst-mode connection is defined as any connection where the time contributed by the control unit to the connection (because of the control-unit circuitry or the sequencing method used) is greater than 32 microseconds. This mode is the normal mode of operation for high-speed I/O devices. These devices force burst mode (by holding up 'operational in') when attached to a channel capable of byte-multiplex operation.

Medium-speed or buffered I/O devices, which may normally operate in either mode, are equipped with a manual or programmable switch to select the mode of operation. The switch setting is overridden when burst mode is forced by the channel. Whenever an interface disconnect, selective reset, or system reset is executed, the force-burst-mode condition of a control unit is overridden. (See "Interface Disconnect," "Selective Reset," and "System Reset" in this chapter.)

Some channels can tolerate an absence of data transfer during a burst-mode operation, such as when a long gap on tape is read, for not more than approximately half a minute. An equipment malfunction may be recognized when the absence of data transfer exceeds this time.

## ENDING SEQUENCE

An ending sequence may be initiated by either the I/O device or the channel. If the procedure is initiated by the I/O device, then ending status is presented, signaling the end of an operation. The end of an operation may be signaled by using only one ending sequence, assuming that both channel-end and device-end status conditions occur together, or the execution of two ending sequences may be required, assuming that device-end status had not been generated at the time channel-end status was presented. If the sequence is initiated by the channel, the I/O device may still require time to reach the point where the proper ending status information is available, in which case one or more ending sequences may be necessary to complete the ending procedure.

One of three situations may exist that result in the initiation of the ending sequence (assuming selection is already obtained):

1. The channel recognizes the end of an operation before the I/O device reaches its ending point. In this situation, whenever the control unit requires service again, the control unit raises 'service in'. The channel responds by raising 'command out', indicating stop. The control unit drops 'service in' and proceeds to its normal ending point without requesting further service. When the I/O device reaches the point at which it would normally send 'channel end', the control unit places the ending status on 'bus in' and raises 'status in'. The channel responds by raising 'service out', unless it is necessary to stack the status, in which case the channel responds with 'command out'.
2. The channel and the I/O device recognize the end of an operation simultaneously.
3. The I/O device recognizes the end of an operation before the channel reaches the end of an I/O operation.

For situations 2 and 3, if the control unit is currently connected and status information is available, the control unit places the ending status on 'bus in' and raises 'status in'. If the control unit is not currently connected when the end of the I/O operation is recognized by the I/O device, a control-unit-initiated sequence is performed, the control unit becomes connected to the I/O interface, and then the status is presented by executing an ending sequence.

If device end is not presented with channel end, device end is presented when it is available by executing a control-unit-initiated sequence.

## CONTROL-UNIT TYPES

Control units are classified according to their ability to concurrently operate and control the activity of attached I/O devices without causing loss of control or data.

### TYPE-1 CONTROL UNIT

The control unit can control the activity of only a single I/O device at a time. If an I/O operation or chain of I/O operations is in execution, the control unit is unable to handle the initiation of other activity associated with any other attached I/O device. The duration of execution begins with acceptance of the first command and continues until the command-chaining condition has been reset in the control unit.

When an error condition has been recognized and the associated sense information has been generated, the control unit can preserve the sense information for the attached I/O device, provided that no other activity is initiated with any other attached I/O device. The duration of time that other activity must be limited begins when the unit-check status has been accepted and continues until the I/O device that signaled unit check has again accepted a command other than test I/O or no-operation.

If, in either of the above two cases, an attempt is made to execute an initial-selection sequence with another I/O device attached to this control unit, control of the ongoing I/O operation or sense information, as applicable, may be lost.

If command chaining is indicated when channel end but not device end is presented by a type-1 control unit, the I/O device that presents channel end is the next I/O device from that control unit to present device-end status, provided the control unit is not addressed in the meantime on the same I/O interface.

### TYPE-2 CONTROL UNIT

The control unit is capable of controlling the activity of more than a single

I/O device at a time without losing pending sense information or the control of ongoing I/O operations, if any. If situations occur where the control unit should limit the amount of concurrent activity, the appropriate busy indication is signaled whenever an initial-selection sequence is attempted. (See "Busy" in Chapter 2, "Operational Description.")

A control unit that has only one assigned device address is a type-2 control unit.

### TYPE-3 CONTROL UNIT

When no error condition described by sense information has been recognized, the control unit can control the activity of more than a single I/O device at a time without losing control of those ongoing operations. Whenever the control unit needs to limit the amount of concurrent activity, the appropriate busy indication is signaled whenever an initial-selection sequence is attempted. (See "Busy" in Chapter 2, "Operational Description.") If an error condition has been recognized and the associated sense information has been generated, the control unit can preserve the sense information for the attached I/O device, provided that no other activity is initiated with any other I/O device. The duration of time that initiation of activity needs to be delayed begins when the unit-check status has been accepted and continues until the I/O device that signaled unit check has again accepted a command other than test I/O or no-operation.

## ADDRESSING

An eight-bit address byte (plus parity) is used over the I/O interface for direct addressing of attached I/O devices. A unique eight-bit I/O-device address is assigned to each I/O device at the time a control unit is installed.

### ADDRESS ASSIGNMENT

At the time of installation, control-unit and I/O-device addresses are assigned as follows:

1. I/O devices that do not share a control unit with other devices may be assigned any device address in the range 0-255, provided the device address is not recognized by any other control unit attached to that I/O interface. Logically,

such I/O devices are not distinguishable from their control unit, and both are identified by the same device address.

2. I/O devices sharing a control unit (for example, magnetic-tape units and disk-storage units) are assigned device addresses within sets of contiguous numbers. The size of such a set of contiguous numbers is equal to the maximum number of I/O devices that can share the control unit, or 16, whichever is smaller. Furthermore, the set of device addresses starts with a device address in which the number of rightmost zeros is at least equal to the number of bit positions required to specify the size of the set. The leftmost bit positions of a device address within a set identify the control unit; the rightmost bit positions designate the I/O device on the control unit.
3. Control units designed to accommodate more than 16 I/O devices may be designed to accommodate either sequential or nonsequential sets of device addresses. Each set consists of 16 device addresses, or the number required to make the total number of assigned device addresses equal to the maximum number of I/O devices attachable to the control unit, whichever is smaller. The device-addressing facilities are added in increments of a set so that the number of device addresses assigned to a control unit does not exceed the number of I/O devices attached by more than 15. For example, if a communications controller has a designed capacity of 56 direct-access paths and if only 40 I/O devices are to be installed at the time of installation, 48 device addresses can be assigned. However, if the full capacity of 56 I/O devices is to be installed, exactly 56 device addresses are assigned.

I/O devices accessible through more than one path in the same system may have a different control-unit address for each path of communications. For sets of I/O devices connected to two or more control units, the portion of the device address identifying the I/O device on the control unit is fixed and does not depend on the path of communications.

Except as indicated in the preceding rules, the assignment of control-unit and I/O-device addresses is arbitrary. The assignment is made at the time of installation, and the device addresses normally remain fixed thereafter.

## ADDRESS DECODING

Control units recognize a device address that meets the following conditions:

1. The device address has correct parity.
2. The device address is assigned to the control unit.

The control unit does not respond to any device address outside its assigned set or sets. For example, if a control unit is designed to control I/O devices that have only bits 0000-1001 in the rightmost positions of the device address, the control unit does not recognize addresses that have 1010-1111 in these bit positions. If no control unit responds to a device address, 'select out' is propagated through all control units and sent back to the channel as 'select in'. This may occur because:

1. A device address is not installed, or
2. A device address has been partitioned out of the system by the program, operator, or customer engineer.

The control unit must respond to those device addresses in the set for which the corresponding I/O devices are either:

1. Ready, or
2. Not ready but which can be made ready by means of an ordinary manual intervention. The not-ready state of an I/O device is indicated by unit-check status and sense data that specifies intervention is required. (See "Intervention Required" in this chapter.)

The control unit may respond to all device addresses in the assigned set, regardless of whether the device associated with the device address is installed. If a control unit responds to a device address for which no I/O device is installed, the unit-check status is set, and the appropriate sense information is made available.

The portion of the device address decoder that identifies the control unit (item 2 under "Address Assignment" in this chapter) can be set at the time of installation for any bit combination.

Control units that are designed to attach only a single I/O device must decode all eight bits of the device address byte. The device address decoder can be set to any bit combination at

the time of installation. (See item 1 under "Address Assignment" in this chapter.)

## COMMANDS

When 'command out' is up during the initial-selection sequence, the byte of information on 'bus out' describes a command. The command specifies to the I/O device the operation to be performed.

The rightmost bit positions indicate the type of operation; the leftmost bit positions comprise a modifier which expands the meaning of the basic operation that is to be performed. The modifier codes and the operations performed when they are decoded are model-dependent.

The command byte on the I/O interface is defined as follows, where "M" is a modifier bit and "P" the parity bit:

	Bit Position								
	P	0	1	2	3	4	5	6	7
Test	P	M	M	M	M	0	0	0	0
Test I/O	1	0	0	0	0	0	0	0	0
Reserved	P	through				0	0	0	0
		0	0	0	1				
		1	1	1	1				
Sense	P	M	M	M	M	0	1	0	0
Basic sense	0	0	0	0	0	0	0	1	0
Sense ID	1	1	1	1	0	0	1	0	0
Reserved	P	M	M	M	M	1	0	0	0
Read backward	P	M	M	M	M	1	1	0	0
Write	P	M	M	M	M	M	M	0	1
Read	P	M	M	M	M	M	M	1	0
Basic read	0	0	0	0	0	0	0	0	1
Control	P	M	M	M	M	M	M	1	1
No-operation	1	0	0	0	0	0	0	1	1

The basic sense, test I/O, no-operation, and sense ID commands are executed by all I/O devices.

## BASIC OPERATIONS

The I/O operation to be executed over the I/O interface is determined by the eight-bit command issued to the I/O device during the initial-selection sequence.

The basic operations are specified by these commands: read, read backward, write, control, sense, and test.

A command with invalid parity is not recognized and therefore not executed.

**Immediate Operation:** Some commands cause the I/O device to signal channel end as initial status during an initial-selection sequence. An I/O operation (except test I/O) causing channel end (but not busy) to be signaled as initial status is called an immediate operation.

An immediate operation is performed when the execution of that command meets the following requirements:

1. Execution requires no more information than that in the command byte; that is, no data bytes are transferred.
2. Channel-end status, without busy, is presented as initial status. Device-end status may also accompany channel end.

Any command (with the exception of test I/O) may be executed as an immediate operation.

## Notes:

1. The inadvertent use of special diagnostic commands, which by intention permit errors to occur on the I/O interface or which introduce the possibility that subsequent commands may be executed erroneously, must be prevented by some form of interlock.
2. An error condition may be recognized by the channel and the I/O operation terminated when 256 or more chained commands are executed with a single I/O device and none of the commands result in the transfer of any data.

## Read

The read command initiates execution of data transfer from the control unit to the channel. The bytes of data within a block are provided in the same sequence as those written by the write command.

A read command with all modifier bits set to zeros is a basic read command. This command is also used as an initial-program-loading (IPL) read command by those devices that provide the IPL function. To perform the initial-program-loading read, the command must be:

1. The first command sent to the I/O device following a system reset.
2. Sent no sooner than 1 millisecond following system reset.



## Read Backward

The read-backward command initiates an I/O operation in the same manner as the read command, except that bytes of data within a block are sent to the channel in an order which is the reverse of that used in writing. The control unit may be designed to cause mechanical motion in the I/O device in a direction opposite of that for a read command, or it may be designed to operate the device as it would for a read command.

Unless otherwise noted, any description in this manual that applies to read also applies to read backward.

## Write

The write command performs the same sequence of signals over the I/O interface as for a read operation. For a write command, however, the data is sent from the channel to the control unit instead of from the control unit to the channel.

## Control

The control command is similar to the write command, except that the command modifier bits received by the control unit are decoded to determine which of several possible functions are to be performed. One function performed may be second-level addressing, which may require several bytes of data to complete the control operation. When the particular control function can be completed without the transfer of data, channel-end status may be presented during the initial-selection sequence.

The byte rate for the bytes transferred during an operation specified by a control command normally need not be faster than the normal read or write rate for the same I/O device.

A control command with all-zero modifier bits performs no operation at the I/O device, except that it satisfies any previously indicated chaining operations and allows certain I/O devices to wait for conditions of checking (or any synchronizing indications) before releasing the channel. This variation

of the control command is a no-operation control.

## Sense

The sense command is similar to a read command, except that the data is obtained from sense indicators rather than from a record source.

The basic sense command (modifier bits set to zeros) initiates a sense operation on all I/O devices and causes the retrieval of up to 32 bytes of data. The basic sense command does not initiate any operation other than the sensing of sense indicators. The basic sense command sent to an available control unit is accepted even though the addressed I/O device is in a not-ready state. (See "Sense Information" in this chapter.) If the control unit detects an error during the sense operation, unit check is sent with the channel-end status condition.

The purpose of the basic sense command is to provide data detailed enough to ascertain the actual state of the device and unusual conditions associated with the execution of the I/O operation during which the error was detected.

The I/O devices that can provide special diagnostic sense information or that can be instructed to perform other special functions by means of the sense command may call for modifier bits to be defined for control of the function. The special sense operation may be initiated by unique combinations of modifier bits, or a group of codes may specify the same special sense function. Any remaining sense command codes may be considered invalid or may cause the same action as the basic sense command, depending on the particular I/O device.

The sense-ID command (modifier bits set to E hex) does not initiate any operations other than sensing the type/model number.<sup>1</sup> If the control unit or device is available and not busy, then the sense-ID command is executed, with up to seven bytes of data transferred. Basic sense data may or may not be reset as a result of executing the sense-ID command.

The sense bytes sent in response to sense ID are:

<sup>1</sup> Units designed prior to October 1, 1980 may properly execute the sense-ID command, may execute the command as a basic sense command, or may reject the command with unit-check status.

<u>Bytes</u>	<u>Contents</u>
0	FF hex
1,2	Control-unit type number
3	Control-unit model number
4,5	I/O-device type number
6	I/O-device model number

All unused sense bytes must be set to zeros.

Bytes 1 and 2 contain the four-decimal-digit control-unit type number that corresponds directly with the control-unit type number on the tag attached to the control unit.

Byte 3 contains the control-unit model number, if applicable. If not applicable, byte 3 is a byte of all zeros.

Bytes 4 and 5 contain the four-decimal-digit device type number that corresponds directly with the device type number on the tag attached to the I/O device.

Byte 6 contains the device model number, if applicable. If not applicable, byte 6 is a byte of all zeros.

Whenever a control unit is not separately addressable from the attached I/O device or devices, the response to the sense-ID command is a concatenation of the control-unit type number and the I/O-device type number.

If a control unit can be addressed separately from the attached I/O device or I/O devices, then the response to the sense-ID command depends on the unit addressed. If the control unit is addressed, the response to the sense-ID command is as follows:

<u>Bytes</u>	<u>Contents</u>
0	FF hex
1,2	Control-unit type number
3	Control-unit model number

The response consists of the control-unit type and model number, with normal ending status presented after byte 3.

If the I/O device is addressed, the response to the sense-ID command is as follows:

<u>Bytes</u>	<u>Contents</u>
0	FF hex
1,2	I/O-device type number
3	I/O-device model number

The response consists of the I/O-device type and model number, with normal ending status presented after byte 3.

## Test

The test-I/O command (modifier bits set to zeros) retrieves from the addressed I/O-device path any status that results from a status condition being recognized, that is stacked, or that is pending. The test-I/O command does not initiate an operation. If no stacked or pending status is encountered along the I/O path being tested, a zero status byte for the selected I/O device is presented to the channel.

With respect to test I/O, the busy condition has a special meaning. (See "Busy" in this chapter.)

Note: Any status presented during execution of a test-I/O command may be stacked.

## SEQUENCE CONTROLS

Sequence controls are used by the channel to control execution of the sequences that are performed between the channel and control unit. Each sequence control uses a special signaling convention over the I/O interface, and each one has a particular meaning to the control unit or I/O device.

### PROCEED

Whenever 'command out' rises in response to the rise of 'address in' during a control-unit-initiated sequence, it means proceed.

Proceed indicates to the I/O device that it should continue the normal servicing request on the interface.

### STOP

Note: When data streaming occurs, this "Stop" section does not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

The channel uses the stop sequence control when data is being transferred and the channel recognizes that the currently executing I/O operation should be ended. "Stop" is indicated by raising 'command out' in response to the rise of 'service in' or 'data in'.

On receipt of the stop signal, the I/O device proceeds to its normal ending point without sending any further 'service in' or 'data in' signals to the channel. The I/O device remains busy

until the ending status is available, is presented to the channel, and is accepted by the channel.

During a data-transfer sequence, 'command out' is transmitted in response to the first 'service in' or 'data in' that is provided after the channel determines that the current operation is to be ended. If 'select out' is down or goes down after this sequence, 'operational in' drops on force-burst-mode-type operations on I/O devices that cannot meet the timeout requirements indicated in "Interface Timeout Considerations." Also, burst-mode I/O devices that have relatively long times between stop and ending status and have no time-dependent chaining requirements drop 'operational in' at this time.

## STACK STATUS

The channel uses the stack sequence control when conditions preclude acceptance of status from the control unit. Stack status is indicated by the rise of 'command out' in response to the rise of 'status in'. The stack-status signal causes retention of status information at the control unit or I/O device until that status is accepted during a subsequent sequence. When 'stack status' occurs, the control unit disconnects from the interface after 'select out' is down. 'Command out' remains up until 'operational in' falls. Any attempt to perform a control-unit-initiated sequence in order to present the status again is under control of 'suppress out'. (See "Suppress Status" in this chapter.) Any status (except zero status presented in response to a command other than test I/O) presented by a control unit in any interface sequence (except the short-busy sequence) may be stacked.

If the channel signals stack status to a control unit as status is being presented, command chaining, if any, is not indicated when that status is subsequently accepted by the channel.

## SUPPRESS DATA

For control units whose rate of data transfer can be adjusted without causing an overrun condition, the channel may use the suppress-data sequence control. The suppression of data occurs according to the following rules. (Operations with completely buffered I/O devices or start-stop devices and the transfer of data for the basic sense command may fall into this category.)

1. Except when data streaming occurs, 'suppress out' is ignored for the first data byte following any selection sequence unless the data transfer is contiguous with initial selection; that is, unless there is no disconnection and reconnection between initial selection and the data-transfer sequence. (See "Data-Streaming Feature" in Chapter 3, "Features.")
2. To ensure recognition by the control unit, 'suppress out' must be up at least 250 nanoseconds before the rise of 'service in' or 'data in', or at least 250 nanoseconds before 'service out' or 'data out' falls. (See "High-Speed-Transfer Feature" in Chapter 3, "Features.")
3. When 'suppress out' is up at the control unit and the operation is in burst mode (either because 'select out' is up or because the control unit is forcing burst mode), the control unit does not raise 'service in' for subsequent suppressible data.

## ACCEPT DATA

Note: When data streaming occurs, this "Accept Data" section does not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

The channel uses the accept-data sequence control during data transfer from the control unit to the channel. When 'service out' or 'data out' rises in response to 'service in' or 'data in', respectively, during a read, read-backward, or sense operation, the channel signals acceptance of the information placed on 'bus in' by the control unit.

## DATA READY

Note: When data streaming occurs, this "Data Ready" section does not apply. (See "Data-Streaming Feature" in Chapter 3, "Features.")

The channel uses the data-ready sequence control during data transfer from the channel to the control unit. When 'service out' or 'data out' rises in response to 'service in' or 'data in', respectively, during a write or control operation, the channel signals that the requested information has been placed on 'bus out' and is ready for acceptance by the control unit.

## SUPPRESS STATUS

Whenever the channel is unable to immediately accept status, the suppress-status sequence control may be used. When 'suppress out' is raised, the control unit does not attempt a control-unit-initiated sequence to present suppressible status.

It is acceptable for a control unit to treat all status as being suppressible only after that status has been stacked. Alternatively, it is acceptable for a control unit to treat status as being suppressible if (1) that status contains channel end and interface disconnect has been received previously during the I/O operation, or (2) that status contains device end that ended an I/O operation when command chaining was not indicated at the time channel end was presented. Also, asynchronous status may be suppressible at the option of the particular control unit without being stacked.

'Suppress out' is up at least 250 nanoseconds before 'select out' rises at the control unit to ensure suppression of status. 'Suppress out' suppresses the initiation of the control-unit-initiated sequence when the sequence is intended to present suppressible-type status. If 'suppress out' rises after a control-unit-initiated sequence has been started, the status sequence proceeds normally.

The relationship between 'request in' and 'suppress out' is described in "Request In" in Chapter 1, "Functional Description."

## ACCEPT STATUS

During presentation of status, the channel may use the accept-status sequence control. When 'service out' rises in response to 'status in', the channel signals that the status placed on 'bus in' by the control unit has been accepted. 'Service out' falls in response to the fall of 'status in'.

## COMMAND CHAINING

During the execution of successive I/O operations, the channel uses the command-chaining sequence control. If command chaining is to occur, it is indicated each time an I/O device presents ending status and, more specif-

ically, if 'suppress out' is up when 'service out' is raised in response to 'status in'. When channel end and device end are presented together and command chaining is to occur, the command chaining is indicated when the status is accepted by the channel. When channel end and device end are not presented together and command chaining is to occur, it is indicated when channel-end status is accepted and again when device-end status is accepted. The command-chaining indication when channel-end status is accepted is different than when device-end status is accepted. These differences are noted below.

Command chaining means that another initial-selection sequence (reselection) is to occur for the I/O device in operation immediately following the presentation of device end, provided that no unusual conditions were encountered during execution of the current operation. The exact time at which the next command is presented depends on the channel and on the parameters of the system.

The command-chaining condition is recognized for each I/O device and allows command chaining to occur when command chaining is indicated by the channel. The command-chaining condition is set whenever the I/O device accepts a command<sup>2</sup> and is reset (1) whenever the I/O device receives system reset, selective reset, interface disconnect, or stack status, or (2) whenever ending status has been accepted without command chaining being indicated. Additionally, the command-chaining condition is reset as described elsewhere in this section.

Command chaining occurs whenever the command-chaining condition is set in the I/O device and command chaining is being signaled by the channel. If the command-chaining condition is reset and an initial-selection sequence is attempted while command chaining is being indicated, the I/O device causes command chaining to be terminated by not accepting the command. Unit check, along with other status, if any, is presented in response to the command. Further information detailing this unusual condition is available to the basic sense command.<sup>3</sup>

When the channel accepts status from a control unit that contains unit check (except when command retry is signaled), unit exception, busy, control-unit end, or attention, the channel terminates command chaining in the channel, and the control unit resets the command-chaining

<sup>2</sup> I/O devices designed prior to October 1, 1981 may delay setting the command-chaining condition until command chaining is indicated.

<sup>3</sup> I/O devices designed prior to October 1, 1981 may not reset the command-chaining condition and therefore do not present unit-check status to the command.

condition for the device whose device address was used to present that status.

Reselection of any I/O device attached to a type-1 control unit resets the command-chaining condition in the control unit. A type-2 or type-3 control unit maintains the command-chaining condition for each device that has an operation in progress. Type-2 and type-3 control units do not reset the command-chaining condition for devices other than the one being selected.

When command chaining is indicated at the time device end is presented, this indication remains valid until reselection is made or until 'suppress out' falls (minimum down-level time to ensure recognition is 250 nanoseconds). To ensure that command chaining occurs, 'suppress out' remains up during the reselection at least until 'operational in' rises. If 'suppress out' drops before the reselection is made, the command-chaining condition is reset in the control unit.

Depending on the particular I/O device, operation, and configuration, the command-chaining indication requires certain functional control that depends on the individual control unit.

If command chaining is indicated when channel end but not device end is presented from a type-1 control unit, the I/O device that presents channel end is the next I/O device from that control unit to present device-end status, unless the control unit is addressed in the meantime on the same I/O interface.

If command chaining is indicated when device end is presented, the control unit ensures that the path to the I/O device remains available until the next initial-selection sequence is initiated that immediately follows the acceptance of device end or until command chaining is no longer indicated. Furthermore, unless command chaining is being canceled by the channel, the immediately following sequence is a reselection of the I/O device presenting the device end.

If command chaining is indicated to an I/O device shared by more than one control unit or channel, the I/O device remains available until the initial-selection sequence is initiated that immediately follows the acceptance of device end or until command chaining is no longer indicated.

To ensure recognition of command chaining by the control unit, 'suppress out' is up at least 250 nanoseconds before 'service out' rises in response to 'status in' and does not fall before 'status in'. If command chaining is not to be indicated, 'suppress out' is down at

least 250 nanoseconds before the rise of 'service out' and does not rise before the fall of 'status in'.

## INTERFACE DISCONNECT

During the execution of an I/O operation, the interface-disconnect sequence control may be used by the channel to signal the control unit to end execution of an ongoing I/O operation.

If 'hold out' is down and 'address out' rises or if 'address out' is up and 'hold out' falls, the presently connected control unit drops 'operational in', thus disconnecting from the interface. Mechanical motion in process continues to a normal stopping point. Status information is subsequently generated and presented to the channel if execution of an operation has been ended by interface disconnect. 'Address out', in this case, may be up concurrently with another outbound tag line. To ensure that interface disconnect is recognized, it must occur at least 250 nanoseconds before the rise of the outbound tag that completes any signal sequence. If no signal sequence is currently being executed and 'select out' (or 'hold out') is active, then 'address out' rises at least 250 nanoseconds before the fall of 'select out' (or 'hold out') to ensure recognition of interface disconnect.

'Operational in' must drop within 6 microseconds after the disconnect indication is received. When 'operational in' drops, the channel may drop 'address out' to complete the interface-disconnect sequence. 'Address out' is down for at least 250 nanoseconds before a new initial-selection sequence is attempted.

The control unit responds to the interface disconnect signal by removing all signals (with the possible exception of 'request in' and 'metering in') from the I/O interface and resets the command-chaining condition, if any. (See "Command Chaining" earlier in this chapter.) On an input operation, data on 'bus in' need not be valid after the rise of 'address out'. On an output operation, data on 'bus out' must be valid until the fall of either 'service in', 'data in', or 'operational in'. (See "Data-Streaming Feature" in Chapter 3, "Features.") When the control unit reaches the normal ending point, it attempts to obtain selection on the interface to present any generated status to the channel.

To ensure recognition of interface disconnect during an initial selection sequence, interface disconnect should

not be signaled in the interval during or subsequent to the rise of 'command out' and prior to or during the rise of 'service out' in response to 'status in'. If interface disconnect is signaled during this interval, it is unpredictable whether the command is executed and whether ending status should be sent by the control unit.

Any abnormal I/O device operation should be indicated by unit check in the status, and the sense information should provide additional details concerning the operation. (See "Unit Check" in this chapter.) The control unit does not generate any status solely as a result of an 'interface disconnect'.

The I/O-device path remains busy after it receives an 'interface disconnect', while performing an operation, until device-end status is accepted by the channel. If 'interface disconnect' is received when the I/O device is not busy, no status is generated, and the I/O device is not made busy.

Note: Except when the data-streaming feature is used, if 'address out' is up concurrently with another out tag, the information on 'bus out' remains valid until the associated in tag drops or until 'operational in' drops. (See "Data-Streaming Feature" in Chapter 3, "Features.")

## SELECTIVE RESET

The selective-reset sequence control is generated by the channel and may occur any time 'operational in' is up.

Selective reset is indicated whenever 'suppress out' is up and 'operational out' drops. This condition causes 'operational in' to fall and causes the particular I/O device in operation and its status to be reset. The operation in process proceeds to a normal stopping point, if applicable, with no further data transfer. (See "Data-Streaming Feature" in Chapter 3, "Features.") The I/O device operating over the interface is the only device that is reset, even on multidevice control units. The particular I/O-device path is in a busy state throughout this procedure.

If selective reset is to be recognized by the control unit, 'suppress out' rises at least 250 nanoseconds before 'operational out' drops and remains up until at least 250 nanoseconds after 'operational out' rises. 'Operational out' stays down until 'operational in' falls or for at least 6 microseconds, whichever is greater, for the selective reset to be effective.

The ready or not-ready state of the control unit is generally not changed by a selective reset. When, however, the enable/disable or online/offline switch was changed before the reset but had not become effective because of the required inhibiting conditions, the ready or not-ready state may change if the reset clears those inhibiting conditions.

'Device end' may be returned after a selective reset has been signaled. The interpretation of the selective reset is model-dependent.

## SYSTEM RESET

The system-reset sequence control is used to reset all control units and I/O devices that are online. (See "Online/Offline" in this chapter.) System reset is indicated whenever 'operational out' and 'suppress out' are down concurrently and the I/O device is in the online mode. This condition causes 'operational in' to fall and causes all control units and their attached I/O devices, along with their status, to be reset. The control units are in a busy state for the duration of their reset procedure. 'System reset' can prepare an I/O device for an initial-program-loading sequence.

The ready or not-ready state of the control unit is generally not changed by a system reset. When, however, the enable/disable or online/offline switch was changed before the reset but is not yet effective because of required inhibiting conditions, the ready or not-ready state may change if the reset clears those inhibiting conditions.

To ensure a proper reset, 'operational out' and 'suppress out' are down concurrently for at least 6 microseconds.

The definition of when the system-reset signal occurs for a system is model-dependent, as is the interpretation of the system-reset signal by the specific I/O device.

## STATUS INFORMATION

When 'status in' is up, the information that appears on 'bus in' is the status byte. The conditions reported in the status byte are the status conditions.

The status pertains to the I/O device or control unit whose device address appeared on 'bus in' (with 'address in') during the control-unit-initiated sequence or the initial-selection sequence. In the case of the short-busy sequence, when no 'address in' occurs,

it is assumed that the status pertains to the addressed I/O device or control unit.

## STATUS BYTE

The status byte has the following format:

<u>Bit Position</u>	<u>Designation</u>
P	Parity
0	Attention
1	Status modifier
2	Control-unit end
3	Busy
4	Channel end
5	Device end
6	Unit check
7	Unit exception

The status byte is transmitted to the channel:

1. During the initial-selection sequence.
2. To present the channel-end status at the termination of data transfer.
3. To present the device-end status and any associated conditions to the channel. The I/O device remains busy during an operation until the channel accepts the device-end status.
4. To present control-unit-end or device-end status, which signals that the control unit or device that was previously busy and interrogated while busy is now free.
5. To present any previously stacked status when allowed to do so.
6. To present status describing asynchronous conditions that are recognized by either a control unit or an I/O device and that are unrelated to previously executed I/O operations. One of these conditions is described by attention, which is normally generated by console or communication devices. Another condition, which is described by device end, unit exception, and attention, is generated when the corresponding device goes from the not-ready to the ready state. These status conditions are handled in the same way as any other status information presented during a control-unit-initiated sequence and are subject to the same rules regarding presentation to the channel and stacking. (See "Device End" and "Control-

Unit-Initiated Sequence" in this chapter.)

Once accepted by the channel, any given status byte is reset and is not presented again.

## UNIT-STATUS CONDITIONS

The following status conditions are detected by the I/O device or control unit and are indicated to the channel over the I/O interface. The causes of these conditions for each type of device and the timing in presenting these conditions are model-dependent. When such conditions have been recognized, status is generated at the I/O device or control unit and is maintained until it is accepted by the channel or is reset. Status that has been generated but not presented to the channel is called "pending status." Status that has been presented to the channel but has not been accepted and is currently being held at the control unit or I/O device is called "stacked status."

When the device is accessible from more than one channel, status resulting from channel-initiated operations is signaled to the channel that initiated the associated I/O operation. The handling of conditions not associated with I/O operations, such as attention, unit exception, and device end because of a not-ready-to-ready transition, depends on the type of device and condition and is model-dependent. (See "Device End" in this chapter.)

Note: Control units and I/O devices should provide interlocks so that status is not lost, hidden, or included with other status when the result would cause the program to misinterpret the original meaning and intent of the status.

## ATTENTION

The attention condition is generated when some asynchronous condition occurs in the I/O device. The condition may be accompanied by other status. Attention is not associated with the initiation, execution, or termination of any I/O operation.

If a condition at the I/O device or control unit has caused the presentation of attention status while an I/O operation is in execution, command chaining, if any, is no longer indicated, and the operation is ended. If command chaining is indicated when channel-end status is presented, command chaining is no longer indicated when device-end status is presented and accompanied by attention.

When the attention condition is indicated, the handling and presentation of the condition to the channel depends on the type of I/O device. When an I/O device is shared between more than one channel path, presentation of attention status to channel paths is model-dependent. Depending on the I/O-device application, attention may or may not be presented until command chaining is no longer indicated.

Attention is accompanied by device end and unit exception when a not-ready-to-ready-state transition is signaled. (See "Device End" in this chapter.)

## STATUS MODIFIER

Status modifier is used by control units in five cases:

1. Control units that cannot provide current status in response to test I/O present the status-modifier bit alone during the initial-selection procedure.
2. Busy control units present the status-modifier bit with the busy bit during the initial-selection sequence or short-busy sequence to differentiate between a busy control unit and a busy device.
3. Control units designed to recognize special ending conditions (such as search equal on a disk) present the status-modifier bit with device end when the special condition occurs.
4. Control units that use the command-retry feature present status modifier with unit check and channel end -- with or prior to device end -- when requesting that the channel initiate the retry procedure.
5. Control units designed to recognize special conditions that must be brought to the attention of the program present status modifier along with the other status indications in order to modify the meaning of the status.

In the first case, provision is made for certain types of control units that cannot provide current status on a demand basis as required by test I/O. Presentation of the status-modifier bit in response to test I/O indicates that the control unit cannot execute the command, and any existing status remains unchanged and unavailable to a test I/O. This type of control unit provides status only on a control-unit-initiated selection sequence.

In the second case, provision is made for indicating that a busy condition pertains to a control unit and not necessarily to the addressed I/O device. (See "Busy" in this chapter for a description of control-unit-busy status conditions.)

In the third case, provision is made for control units designed to recognize special ending or synchronizing conditions. If the special condition occurs, the status-modifier bit with the device-end bit is presented during the status presentation. When status-modifier and device-end bits are present in the status, it indicates that the normal sequence of commands must be modified.

In the fourth case, provision is made for control units to request a retry of the execution of the current command. The situations encountered during which retry is requested are model-dependent.

In the fifth case, provision is made for control units designed to recognize special conditions that are unrelated to the execution of an I/O operation. These conditions must be brought to the attention of the program. Status modifier is used to modify the meaning of the other accompanying status indications. The meanings of the status combinations are model-dependent.

## CONTROL-UNIT END

Only control units that can indicate a control-unit-busy condition can indicate a control-unit-end condition. The control-unit-end condition occurs for one or both of the following:

1. The control unit was interrogated while it was in the busy state. Interrogated in the busy state means that, during an initial-selection sequence, conditions at the control unit precluded execution of an operation with an I/O device on the control unit and that the control unit responded with busy and status modifier (control-unit busy) in the unit-status byte. If a control unit responded with control-unit-busy status, it must return a control-unit-end status indication when the busy period no longer exists. Only one control-unit-end indication is returned, regardless of the number of times the control unit has responded control-unit busy during the busy period. Also, the single control-unit-end indication is provided independent of the different device addresses used when indicating control-unit busy. (See "Status Modifier" in this chapter.)



2. The control unit detected an unusual condition while busy, but after channel end was accepted by the channel. Indication of the unusual condition accompanies control-unit end.

If the control unit remains busy executing an operation after signaling channel end but is not interrogated and does not detect an unusual condition, control-unit end is not generated.

The device address associated with control-unit end is determined as follows:

1. The device address of the selected device is used if control-unit end is to be presented with channel end and/or device end.
2. If control-unit end is generated without channel end or device end and the status is presented during a control-unit-initiated sequence, the device address to be used when presenting this status is any legitimate address associated with the control unit. A legitimate address is any address that is not associated with an I/O device currently executing an I/O operation and that the control unit recognizes regardless of whether or not the I/O device is actually attached.
3. If control-unit end is to be presented during an initial-selection sequence, the device address is the same as the device address issued with 'address out'. The sequence may be executed as either the initial-selection sequence or the short-busy sequence.

The control-unit-end condition can be signaled with channel end, or any time after channel end, and may be accompanied by other status bits. When control-unit end is signaled with a control-unit-initiated sequence in the absence of any preceding data transfer or other status conditions, the status may be associated with any address assigned to the control unit. For control units attaching more than a single I/O device, a pending control-unit end for one I/O device does not necessarily preclude initiation of new operations with other attached devices. Whether the control unit allows initiation of other operations is at the option of the control unit. Control-unit end causes command chaining to be suppressed.

### Temporary Control-Unit Busy

When the busy state of the control unit is temporary, control-unit end can be included with busy and status modifier in response to interrogation, even though the control unit is not free. The busy condition may be considered temporary if it lasts less than approximately 2 milliseconds.

Note: When a control unit has signaled the temporary control-unit-busy condition, the channel may attempt an initial-selection sequence immediately following acceptance of the status.

### BUSY

The busy indication occurs only during an initial-selection sequence or short-busy sequence and means that conditions existing at the I/O device or control unit preclude execution of the intended I/O operation because one of these four situations exists:

1. An I/O operation initiated during a previous initial-selection sequence is being executed.
2. Stacked or pending status conditions exist (except as noted later in this section).
3. The control unit is shared by channels or I/O devices, or an I/O device is shared by control units, and the shared facility is not available.
4. A self-initiated function (for example, microdiagnostics or data movement internal to the I/O device) is being performed.

An I/O operation is being executed from the time initial status is accepted until device end is accepted. Status conditions for the addressed I/O device, if any, accompany the busy indication.

If the busy condition applies to a control-unit function, busy is accompanied by status modifier.

Busy is indicated to a test-I/O command under the same conditions that busy is indicated to other commands; however, busy is not indicated to a test-I/O command if there is an available path to the I/O device and there is stacked or pending status for that I/O device.

The busy condition causes command chaining to be suppressed.

Note: For I/O devices designed after October 1, 1980, if zero status has previously been stacked at a device and

if a command other than test I/O is received by the same device, the zero status is discarded by the control unit or device, busy is not indicated, and execution of the command proceeds normally.

#### CHANNEL END

Channel end is caused by the completion of the portion of an I/O operation involving execution of a command; this execution may also include the transfer of data between the I/O device and the channel.

Each I/O operation causes only one channel-end indication to be generated. The channel-end condition is not generated unless the command is accepted, that is, the initial status byte for the operation contained either all zeros, or channel end without busy as in the execution of an immediate operation. If the initial-status byte contained all zeros, the next status byte presented by the device includes channel end. If channel end is not included in this status byte, an error condition may be recognized by the channel and the I/O operation, if any, terminated.

During an I/O operation, the exact time that channel end is generated depends on the operation and the type of I/O device. For operations such as writing, some I/O devices generate the channel-end condition when the data has been written. On other I/O devices that later verify the writing, channel end may be delayed until verification is performed, depending on the I/O device. On I/O devices equipped with buffers, the channel-end condition may occur on completion of data transfer between the channel and the buffer. During execution of control commands that involve data transfer, channel end is usually generated after the data is transferred to the control unit, although for some I/O operations, channel end may be delayed until the operation is completed. Operations that do not cause data to be transferred, as in the case of immediate operations, can provide the channel-end condition during the initial-selection sequence.

#### DEVICE END

Device end is presented to the channel (1) when the completion of an I/O operation is signaled by the I/O device, (2) when the I/O device, having previously responded busy, signals that a change from the busy to the not-busy state has occurred, (3) when the I/O

device signals that a change from the not-ready to the ready state has occurred, and (4) when the control unit or I/O device signals that an asynchronous condition has been recognized.

Each I/O operation causes only one device-end condition. In this situation, the device-end condition is not generated unless the command is accepted.

The device-end condition associated with an I/O operation is generated either simultaneously with the channel-end condition or later. For data-transfer operations on some I/O devices, the operation is complete at the time channel end is generated, and both device end and channel end occur together. The time at which device end is presented depends upon the I/O device type and the kind of command executed. For most I/O devices, device end is presented when the I/O operation is completed. In some cases, for reasons of performance, device end is presented before the I/O operation has actually been completed at the I/O device. However, in all cases, when device end is presented, the I/O device is available for an initial-selection sequence if command chaining was indicated when the device end was accepted by the channel. During execution of control commands, the device-end condition may be generated at the time channel end is generated, or later.

If an I/O operation has been completed and status of only channel end has been accepted by the channel, the next status byte presented by the I/O device contains either device end or device end and status modifier; otherwise, command chaining, if any, is not indicated. A device-end signal received by the channel in the absence of any unusual conditions causes the channel to initiate the next command-chained I/O operation, if any. Device-end status, when received by the channel in the presence of unusual conditions, causes the channel to terminate command chaining, if any, and report the unusual condition to the program. If, during command chaining, an unusual condition is recognized subsequent to device end having been accepted by the channel and before acceptance of the next command by the control unit, command chaining does not occur, and the device-end status previously presented is not made available to the program.

If an I/O device previously responded busy, device end is signaled on the path over which the initial-selection sequence was initiated when the I/O device becomes not busy. In this situation, device end is signaled only once, independent of the number of times the I/O device responded busy.

Device end is accompanied by attention and unit exception when a state change from not-ready to ready is signaled.<sup>4</sup> An I/O device is considered to be not ready when operator intervention is required in making the I/O device ready. A not-ready condition can occur, for example, because of any of the following:

1. An unloaded condition in a magnetic-tape unit.
2. Card equipment out of cards or stacker full.
3. Printer out of paper.
4. Error conditions that need operator intervention.
5. An I/O device having changed from the enabled to the disabled state.

When a not-ready-to-ready-state transition occurs on an I/O device that is shared by more than one channel, the I/O device must present the appropriate indication<sup>4</sup> to all attached channels.

Device end is accompanied by other status when conditions are recognized that are unrelated to the execution of an I/O operation. These conditions are presented to the channel as they occur.

## UNIT CHECK

Unit check indicates that the I/O device or control unit has detected an unusual condition that is detailed by information available to a basic sense command. The occurrence of unit check may indicate that a programming error or an equipment error has been detected, that the not-ready state of the device has affected the execution of the command, or that an exceptional condition other than the one identified by unit exception has occurred. The unit-check bit provides a summary indication of the conditions identified by sense data. (See also "Command-Retry Feature" in Chapter 3, "Features.")

An error condition causes the unit-check indication when the error condition occurs during execution of a command or during some activity associated with an I/O operation. Unless the error condition pertains to the activity initiated by a command or is of significance to the program, the condition does not cause the program to be alerted after device end has been cleared; a malfunc-

tion may, however, cause the device to become not ready.

Unit check is indicated when the existence of the not-ready state precludes a satisfactory execution of the command or when the command, by its nature, tests the state of the I/O device. When no status is pending for the addressed I/O device at the control unit, the control unit signals unit check when test I/O or the no-operation control command is issued to a not-ready I/O device. In the case of a no-operation, the command is not accepted, and channel end and device end do not accompany unit check.

Unless the command is designed to cause unit check, such as a command to rewind and unload magnetic tape, unit check is not indicated if the command is properly executed, even though the I/O device has become not ready during or because of the operation. Similarly, unit check is not indicated if the command can be executed with the I/O device not ready. A console could, for example, accept and execute the alarm-control command when the printer is not ready. Selection of an I/O device in the not-ready state does not cause a unit-check indication when the sense command is issued or when a status condition is pending for the addressed device at the control unit.

If, during the initial-selection sequence, the I/O device detects that the command cannot be executed before initial status is presented, unit check is presented to the channel and it appears without channel end, control-unit end, or device end. Such unit status indicates that no action has been taken at the device in response to the command. If the condition that precludes proper execution of the operation occurs after the command has been accepted, unit check is accompanied by channel end, control-unit end, or device end, depending on when the condition was detected. Errors detected after device end is cleared may be indicated by signaling unit check with attention, unit check with control-unit end, or unit check with device end.

Unit-check status presented either in the absence of or accompanied by other status indicates only that sense information is available to the basic sense command. Presentation of either channel end and unit check or channel end, device end, and unit check does not provide any indication as to the kind of conditions encountered by the control unit, the state of the I/O device, or whether execution of the operation has been started. Instead, descriptions of

<sup>4</sup> I/O devices designed prior to October 1, 1980 present status of only device end or device end, attention, and unit exception when signaling a not-ready-to-ready-state transition.

these conditions or states are provided in the sense information.

When unit check appears with channel end and without device end, the sense data and an available device path are preserved until after the device end and the sense data are accepted or reset.

Errors, such as invalid command code or invalid command-code parity, do not cause unit check when the device is working or contains a pending interruption condition at the time of selection. Under these circumstances, the I/O device responds by providing the busy bit and indicating any pending status. The command-code validity is not indicated.

Termination of an operation with the unit-check indication causes command chaining to be suppressed. (See "Command Retry Feature" in Chapter 3, "Features.")

**Note:** If an I/O device becomes not ready on completion of a command, the ending status can be cleared by test I/O without generation of unit check because of the not-ready state. Any subsequent test I/O issued to the I/O device causes a unit-check indication.

## UNIT EXCEPTION

Unit exception means that the I/O device detected an unusual condition that needs to be reported to the program. During execution of an I/O operation, unit exception has only one meaning for any particular command and type of I/O device. A sense operation is not required as a response to the acceptance of a unit-exception condition.

A unit-exception condition may be generated when the I/O device is executing an I/O operation, or when the device is involved with some activity associated with an I/O operation and the condition is of immediate significance. If a device detects a unit-exception condition during the initial-selection sequence, unit exception is presented to the channel without channel end, control-unit end, or device end. Such unit status indicates that no action has been taken at the device in response to the command. If the condition that precludes normal execution of the operation occurs after the command has been accepted, unit exception is accompanied by channel end, control-unit end, or device end, depending on when the condition is detected. Any unusual condition associated with an operation but detected after device end is cleared, is indicated by signaling unit exception with attention.

The unit-exception condition causes command chaining to be suppressed.

Unit exception is accompanied by device end and attention when a not-ready-to-ready-state transition is signaled. (See "Device End" in this chapter.)

## STATUS COMBINATIONS

The following rules indicate status combinations which are appropriate or inappropriate, depending on the state of an I/O operation when status is presented by the device. If a status byte is accepted by the channel which has valid parity, but which contains a combination of status bits that is inappropriate at the time the status is presented, an error condition may be recognized by the channel. If such an error is recognized, command chaining is suppressed. (Appropriate and inappropriate status combinations are summarized in tables contained in Appendix D.)

1. When status is presented during the short-busy sequence, the only appropriate status combinations are either (1) busy and status modifier, (2) busy, status modifier, and control-unit end, or (3) busy. All other status combinations presented during the short-busy sequence are considered inappropriate.

(See Figure D-1 in Appendix D.)

2. When status is presented during an initial-selection sequence which is not the result of command chaining, for any command other than test I/O, the following status combinations are considered inappropriate:
  - a. The status byte contains the device-end bit set to one and both the channel-end and busy bits set to zeros.
  - b. The status byte contains any combination of control-unit-end, status-modifier, and attention bits set to ones and no other status bit is set to one.
  - c. The status byte contains the control-unit-end bit set to one, busy, channel-end, and device-end bits set to zeros, and any combination of unit-check and unit-exception bits set to ones.

(See Figure D-2 in Appendix D.)

3. When status is presented during an initial-selection sequence which is

a result of command chaining, for any command other than test I/O, the following status combinations are considered inappropriate:

- a. The status byte contains the device-end bit set to one and both the channel-end and busy bits set to zeros.
- b. The status byte contains any combination of control-unit-end, status-modifier, and attention bits set to ones and no other status bit is set to one.
- c. The status byte contains the control-unit-end bit set to one, busy, channel-end, and device-end bits set to zeros, and any combination of unit-check and unit-exception bits set to ones.
- d. The status byte contains the busy bit set to one, except when only the busy and device-end bits are set to ones or when only the busy and attention bits are set to ones.
- e. The status byte contains the status-modifier bit set to one, any combination of unit-check and unit-exception bits set to ones, and all other status bits set to zeros.

(See Figure D-3 in Appendix D.)

4. At the first presentation of status after an initial status byte of zero has been accepted (provided that there is no intervening sequence initiated by the channel to select the device) the following status combinations are considered inappropriate:
  - a. The status byte contains the channel-end bit set to zero.
  - b. The status byte contains the busy bit set to one.

When the dynamic-reconnection feature is used, control-unit-end status alone, when presented during a control-unit-initiated sequence, does not cause the channel to recognize an error condition. See "Dynamic Reconnection" in Chapter 3, "Features."

(See Figure D-4 in Appendix D.)

5. At the first presentation of status after the channel accepts a status byte with either (1) only the channel-end bit set to one and command chaining was indicated, (2) only channel-end, unit-check,

and status-modifier bits set to ones when performing a command retry and command chaining was indicated, or (3) the channel-end bit set to one and the device-end bit set to zero and command chaining was not indicated (provided that there is no intervening sequence initiated by the channel to select the device) the following status combinations are considered inappropriate:

- a. The status byte contains the busy bit set to one.
- b. The status byte contains the channel-end bit set to one.
- c. The status byte does not contain either the device-end bit set to one or the combination of control-unit-end and unit-check bits set to ones, except under certain conditions when the status byte contains the control-unit-end bit alone, as described below.

If command chaining was not indicated, status of control-unit end alone does not cause the channel to recognize an error condition. If the dynamic-reconnection feature is in use, control-unit-end status alone, when presented during a control-unit-initiated sequence, does not cause the channel to recognize an error condition. See "Dynamic Reconnection" in Chapter 3, "Features."

For the purpose of determining whether status is appropriate or inappropriate after control-unit-end status alone is presented as described above, any subsequent status will be considered to have been presented without the intervening control-unit-end status.

(See Figure D-5 in Appendix D.)

### SENSE INFORMATION

Data transferred during a sense operation provides information concerning unusual conditions detected in a previous I/O operation and concerning the actual state of the I/O device. Information provided by the basic sense operation is more detailed than that supplied by the unit-status byte, and may describe reasons for the unit-check indication. It may also indicate, for example, that the I/O device is in the not-ready state, that a tape drive is in the file-protected state, or that magnetic tape is positioned beyond the end-of-tape marker.

Error information used for the recovery of the I/O operation, if any, normally is provided in the first byte of sense data. (See "Sense Byte.") All I/O devices provide at least the first sense byte and may transfer up to 31 additional sense bytes during execution of the basic sense command. (See "Sense" in this chapter.) The amount and the meaning of the additional sense data are model-dependent.

A device which normally operates in byte-multiplex mode and transfers a single byte during data transfer should transmit the maximum number of sense bytes possible during the 32-microsecond byte-multiplex timeout limitation.

The sense information that pertains to a previous I/O operation or other unit action at an I/O device may be reset any time after the completion of the basic sense command addressed to that I/O device. Except for the test I/O and no-operation commands, any other command addressed to the control unit may be allowed to reset the sense information, provided that the busy bit is not included in the initial status. The sense information may also be changed as a result of asynchronous actions, such as when not-ready-to-ready status is generated.

Sense information that results from more than one condition at the unit is not combined when this action would cause the program to misinterpret the original meaning and intent of the sense information. When a group of sense indicators is shared with different devices, the residual control-unit sense data that pertains to a previous command addressed to the control unit may be reset if the I/O device addressed is different from the I/O device which generated the sense data.

A command code with invalid parity causes the sense information to be replaced only if unit check is turned on as a result of the invalid parity.

## SENSE BYTE

The first six bits of the first sense data byte (sense byte 0) are common to all I/O devices. The six bits are independent of each other and, when set to ones, designate the following:

<u>Bit</u>	<u>Description</u>
0	Command reject
1	Intervention required
2	Bus-out check
3	Equipment check
4	Data check
5	Overrun

## SENSE CONDITIONS

### Command Reject

The command-reject condition occurs when the I/O device has detected a programming error. The command-reject condition is generated when a command is received which (1) the I/O device is not designed to execute (such as a read backward issued to a direct-access storage device), or (2) the I/O device cannot execute because of its present state (such as a write issued to a file-protected tape unit). In case 2, the program may have required use of an uninstalled optional feature or may have specified invalid control data. Command reject is also indicated when an invalid sequence of commands is recognized (such as a write to a direct-access storage device without the data block having previously been designated).

### Intervention Required

The intervention-required condition occurs when the command could not be executed because of a condition that requires intervention at the I/O device. It may indicate conditions such as an empty hopper in a card punch or a printer that is out of paper. The condition may also be generated when the addressed device is in the not-ready state, in test mode, or not provided on the control unit.

### Bus-Out Check

The bus-out check condition occurs when the I/O device or the control unit receives a data byte or a command byte with invalid parity over the I/O interface.

During writing, bus-out check indicates that a parity error was detected and that incorrect data may have been recorded at the I/O device. However, the condition does not cause the operation to be terminated prematurely, unless the operation is such that an error precludes meaningful continuation of the operation. No operation is initiated if the command code has a parity error.

### Equipment Check

The equipment-check condition occurs when an equipment malfunction has been detected logically between the I/O interface and the I/O medium. On output operations, this malfunction may have caused invalid data to be recorded. Detection of equipment check stops data transmission and terminates the opera-

tion prematurely when the error prevents any meaningful continuation of the operation.

### Data Check

The data-check condition occurs when invalid data has been detected by the

**This page is intentionally left blank.**



control unit or I/O device. When signaled during execution of a write command, the data-check condition indicates that incorrect data may have been recorded by the I/O device. When signaled during execution of a read command, the data-check condition means that the control unit may have forced correct parity on the data transferred to the channel.

Data errors on reading and writing cause the operation to be terminated prematurely only when the errors prevent meaningful continuation of the operation (loss of synchronization).

### Overrun

The overrun condition occurs when the channel fails to respond to the control unit in the anticipated time interval to a request for service from the I/O device. When the total activity initiated by the program exceeds the capability of the channel, an overrun may occur. The overrun may occur (1) when data is transferred to or from a control unit that does not have a buffer large enough to hold all the data transferred by the command, (2) when the data-streaming feature is used, or (3) when the I/O device receives the new command too late during command chaining.

On an output operation, overrun indicates that data recorded at the device may be invalid. In these cases, data overrun normally stops data transfer, and the operation terminates as when 'stop' is indicated.

## GENERAL SYSTEM CONSIDERATIONS

### INTERFACE-TIMEOUT CONSIDERATIONS

Except when data streaming occurs, signaling over the I/O interface is specified to be interlocked and is therefore not time-dependent. (See "Data-Streaming Feature" in Chapter 3, "Features.") Because the signaling may not always be time-dependent, a certain category of machine malfunctions may cause hangup of the channel and, unless detected and canceled, may result in hangup of the system. The timing considerations involved in determining malfunction cases are described in the following paragraphs.

All references in this section to particular time considerations represent

the maximum permissible time in worst-case situations. All control units are designed for minimum signal-sequence-response times within the limitations of the circuit family used and the sequencing method required for particular I/O devices. The maximum time contributed by a control unit to an initial-selection or a short-busy sequence, because of the control-unit circuitry, is no more than 32 microseconds. This restriction also applies to all byte-multiplex-mode connections. (See "Selection" in this chapter.)

A particular interface signal sequence may take longer because of other factors, such as delays introduced by the channel or a delay because of the need for a burst-mode I/O device to capture the interface before reaching the specified record area of the recording medium (not greater than 500 milliseconds).

In addition to excessive delays that may occur within a particular signal sequence, excessive delays may exist between data-transfer sequences during burst-mode operation. The maximum delay between data-transfer sequences in burst mode is 500 milliseconds. The channel does not indicate a malfunction unless the delay between data-transfer sequences exceeds approximately 30 seconds. Activity on the I/O interface may be absent because of other conditions (such as reading a long gap on magnetic tape created by successive erase commands).

### PROPAGATION OF SELECT OUT

During an initial-selection sequence with a device address not assigned to a control unit or, during a control-unit-initiated sequence, if the control unit does not require service or selection, 'select out' is propagated by the unit within 600 nanoseconds to meet performance requirements; in no case does it exceed 1.8 microseconds per control unit. (See "Initial-Selection Sequence" and "Control-Unit-Initiated Sequence" in this chapter.) If an initial-selection sequence with a device address assigned to a control unit is attempted and (1) the addressed I/O device is unavailable, and (2) 'select out' is to be propagated, the time required to propagate 'select out' may be the same as the time required for an initial-selection sequence for that control unit. This time, measured at the external cable connectors, extends from the rise of the incoming 'select out' to the rise of the outgoing 'select out'.

## SYSTEM CONFIGURATION

Number of Units: In the general system configuration, as many as eight control units can be directly connected to a single I/O interface.

Internal Cabling: Except for 'select out' and 'select in', the maximum allowable internal resistance (including all contact resistance) contributed by a channel or control unit, for every signal line, is 1.0 ohm. The combined 'select out' and 'select in' maximum resistance for the control unit is 1.5 ohms.<sup>5</sup> The maximum resistance for 'select out' and 'select in' in a control unit usually occurs when the control-unit power is off and electrical bypassing is effective.

For control units, the internal resistance is measured between the incoming and the outgoing pins on the external connectors. For channels, the measurement is made between the external connector pin and the corresponding channel driver or receiver.

With the exception of the 'select out'/'select in' line, the maximum signal delay measured between the external pins is 15 nanoseconds. The maximum skew between any two signal lines is 1 nanosecond. The delay for the bus lines does not exceed the delay for the tag lines.

External Cabling: The cable length available for the interconnection of channel and control units is primarily limited by the resistance to the interface lines which is contributed by the channel and the control units. For specific control units, the signal delays due to cable length require that the control unit be relatively close to the channel. However, the maximum allowable cable length is limited by total series resistance between drivers and receivers, if signal-cable delay requirements are met for each attached control unit.

The maximum external connector-to-connector cable length for unrestricted general systems configurations is determined by the combined internal resistance specification for 'select out'/'select in'. The maximum line resistance for the total 'select out'/'select in' loop is not greater than 52.5 ohms for worst-case conditions.

In certain customized installations where 'select out' is redriven at the end of the cable, the maximum cable length is determined by the 1.0-ohm internal control-unit and channel resistance specifications for the signal lines other than 'select out' and 'select in'. In this case, the maximum cable length can be calculated by using the 33-ohm driver-to-receiver maximum line resistance specification for worst-case conditions.

## OFFLINE/ONLINE

Because in an offline mode it must not interfere in the operation between the channel and other control units on the same I/O interface, the control unit provides the following:

1. A logical bypass for 'select out'
2. A gating off of all other line drivers to prevent interference on the I/O interface

The control-unit transition to or from an offline/online condition does not cause a machine malfunction. The following minimum conditions exist concurrently before the online-to-offline transition occurs:

1. The online/offline switch is set to offline.
2. The control unit is not actively communicating with the channel, either for executing an I/O operation ('operational in' is active) or for signaling a busy condition ('status in' is active during a short-busy sequence).
3. The control unit is not currently disconnected from the I/O interface during the execution of an I/O operation involving that control unit.
4. No stacked, pending, or forthcoming status exists for this unit.
5. No command chaining is currently indicated for this unit.

Existence of the preceding conditions ensures that no machine malfunctions occur when the operator throws the switch from online to offline.

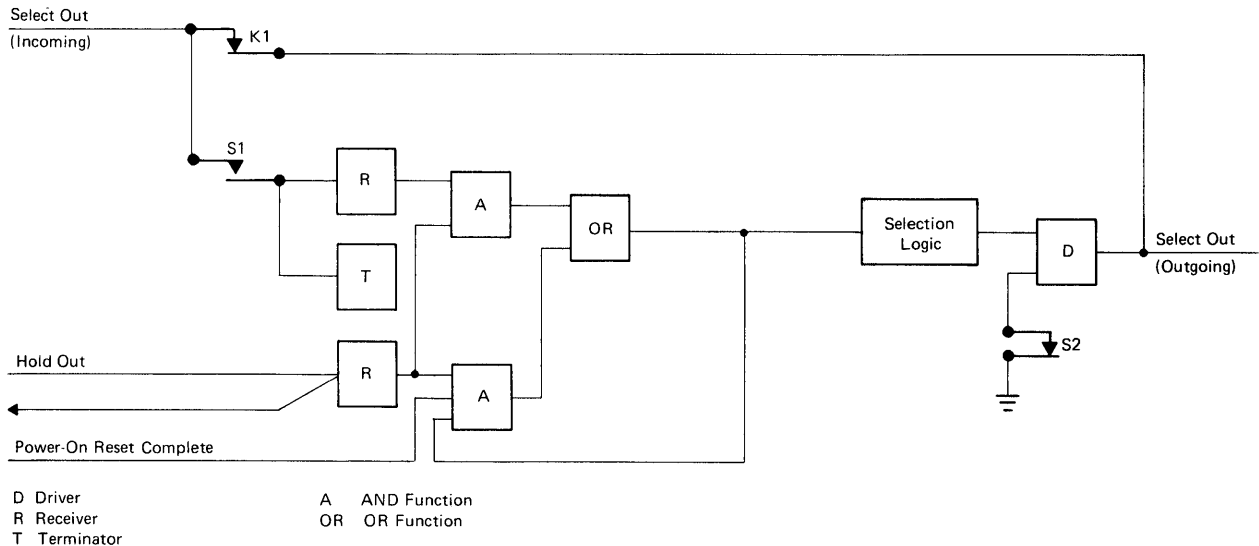
<sup>5</sup> For channels and control units designed before October 2, 1970, the maximum allowable internal resistance (including all contact resistance) contributed by a channel or control unit for every signal line except for 'select out' and 'select in' is 2.0 ohms. The combined 'select out' and 'select in' maximum resistance for a control unit is 3.0 ohms.

**POWER EFFECTS**

Steady State: The power-off state of any control unit does not affect any operations of other control units on the I/O interface. The control unit whose power is off provides an electrical bypass for 'select out', and all of its interface driver and receiver circuits are prevented from interfering with the I/O-interface signals. The incoming 'select out' signal terminator at the receiver is disconnected when the signal is electrically bypassed. When power is off in all units, 'select out' is propagated back to the channel.

Transient and Spurious Signals: Each control unit is designed so that, if proper procedures are followed, the process of individually powering on or off does not cause the driver or receiver circuits of the control unit to generate noise on the I/O-interface signal lines.

Transient and Select-Out Pulse Splitting: Each control unit ensures that the process of electrically bypassing 'select out' before power changes does not interfere with 'select out' propagation, except for possible short discontinuities (less than 1.8 microseconds) because of contact bounce coincident with signal delay through parallel logic circuits. Also, each control unit ensures that 'select out' discontinuities, which may occur when another control unit on the I/O interface is being powered on or off, do not affect the propagation of 'select out'. This is accomplished by the use of a special latch circuit. The latch is turned on by the coincidence of 'select out' and 'hold out' and is reset by the fall of 'hold out'. The circuit is in series with the remaining selection logic in the control unit and causes a constant 'select out' signal within the control unit and, therefore, to all following control units, regardless of variations in the input 'select out' signal (see Figure 2-1).



**Figure 2-1. Representative 'Select Out'/'Hold Out' Special Latch**

Power Off/On Sequence Requirements: The combination of proper procedures and circuitry provides the following sequence of events for powering off:

1. Logical disconnection of the unit from the I/O interface. When the unit becomes logically disconnected, a panel indicator (I/O interface disabled), in the proximity of the power-off control, is turned on. (The control unit can become logically disconnected when it and all connected I/O devices have completed all I/O operations, when no status is pending or stacked, and when chaining is not indicated.) This ensures that no unfinished operations exist that can cause indication of machine malfunction when power is turned off. Note that when a control unit is logically disconnected from the I/O interface, all its drivers except 'select out' are logically gated off. A logical disconnect may be accomplished as a result of going offline by use of the online/offline switch or the enable/disable switch.
2. Closing of the 'select out' bypass circuit (mechanical contact, K1, Figure 2-1). The normal logical electronic bypass of 'select out' is still active when the control unit is logically disconnected.
3. Opening of the connection from the line to the 'select out' receiver terminator (mechanical contact, S1, Figure 2-1)
4. Clamping of the interface driver gates to ground by means of a mechanical contact (S2, Figure 2-1), if gated drivers are used to avoid transient signals on the I/O-interface lines
5. Turning power off. See the System Library manual IBM System/360 and

System/370 Power Control Interface, Original Equipment Manufacturers' Information, GA22-6906.

For powering on, the sequence is reversed:

1. Turning power on. During the power-on sequence, a power-on reset automatically resets the control-unit circuitry, including resetting of the special 'select out' latch, regardless of 'hold out'.
2. Unclamping of driver gates. (Open mechanical contact, S2.)
3. Connection of the 'select out' receiver terminator (S1)
4. Opening of the 'select out' bypass circuit (K1)

The following step only occurs after the enable/disable switch is reset to enable:

5. Logical connection of the unit to the interface (I/O-interface-disabled panel indicator goes off).

If some method, such as automatic power sequencing rather than gated I/O-interface driver circuits, is used to eliminate transients on the signal lines, the steps in the sequence that refer to driver gates may be eliminated.

Note: The 'select out' bypass function (relay transfer) of a power-off or power-on sequence must be completed in one control unit attached to a channel before the 'select out' bypass function (relay transfer) is started in another control unit. Therefore, the 'select out' bypass function should be completely automatic or at least should be completed without interruption once it has started.

Unless otherwise noted, the definitions provided in Chapters 1 and 2 apply when any of the features described in this chapter are used.

### BUS-EXTENSION FEATURE

This feature extends the interface by providing an additional bus to allow information transfer of two bytes in parallel instead of one byte. The basic bus is designated bus 0, while the additional bus is designated bus 1. If control units using different bus widths are attached to the same I/O interface, the control units using the bus-extension feature must be attached first; the last control unit using the bus-extension feature must terminate bus 1.

The data lines of the additional bus are defined similarly to those of bus 0. In addition to data lines, 'mark in' and 'mark out' lines are defined for each bus, including bus 0, to indicate the presence of information on that bus. 'Mark in' and 'mark out' parity lines are also defined to permit a validity check of the mark lines.

The number of mark lines active when 'address out', 'address in', 'command out', or 'status in' are up indicates the number of bytes used for these functions. For output operations, the control unit must raise the 'mark in' lines before raising 'service in' or 'data in' to indicate the number of bytes required. The channel must then indicate by the 'mark out' lines the number of bytes provided on 'bus out' when 'service out' or 'data out' is raised. The bus width indicated during the early data-bus-width indication must be the maximum used by the control unit for that operation. (See "Early Data-Bus-Width Indication" in this chapter.) Transfers involving less than the maximum indicated bus width are permitted when the length of the block of information being transferred is not an integral multiple of the maximum transfer width. The partial transfer thus required is permitted as the last transfer of the block.

All information transferred, including that transferred by a read-backward command, must be gated so that the first byte is on bus 0 and the second byte is on bus 1.

### MARK-OUT LINES

During sequences involving the transfer of information from the channel to the control unit, the 'mark out' lines are valid for the same period of time as the information on 'bus out' and indicate the number of bytes provided by the channel on 'bus out'.

For sequences involving the transfer of data from the control unit to the channel, the 'mark out' lines indicate the number of bytes that the channel is accepting. The 'mark out' lines are valid from the rise of the outbound tag line until the resulting fall of the inbound tag line.

A channel capable of extended-bus operation ensures that the number of 'mark out' lines up, including 'mark out parity', is odd when the I/O device is addressed with 'address out' up. Additionally, if the control unit with which the channel is communicating is operating with extended-bus capabilities, the channel maintains odd mark-line parity whenever 'service out', 'data out' or 'command out' is up.

### MARK-IN LINES

During sequences involving the transfer of data from the control unit to the channel, the 'mark in' lines are valid for the same period of time as the information on 'bus in' and indicate the number of bytes provided by the control unit on 'bus in'.

For sequences involving the transfer of data from the channel to the control unit, the 'mark in' lines indicate the number of bytes requested by the control unit for that sequence and are valid from the rise of the inbound tag line until the rise of the corresponding outbound tag line.

The number of 'mark in' lines up, including 'mark in parity', must be odd whenever a control unit is operating with extended-bus capabilities and 'service in', 'data in', 'address in' or 'status in' is up. Control units operating with the one-byte interface do not necessarily activate any 'mark in' lines, except for command-retry indication. (See "Command-Retry Feature" in this chapter.)

## EARLY DATA-BUS-WIDTH INDICATION

The control unit indicates to the channel the maximum bus width to be used for an operation by raising the 'mark in' lines during the initial-selection sequence. These 'mark in' lines are valid from the rise of 'operational in' to the fall of 'address out'.

## I/O-ERROR-ALERT FEATURE

When a malfunction that affects the continued-execution capability of a control unit occurs, 'disconnect in' may be raised to alert the channel of the condition. An example of such a condition is one in which a microcoded control unit is communicating with the channel at the time a control-storage error is detected. Such a control unit may be unable to complete an interface sequence properly. Another example is one in which the control unit, while currently disconnected from the I/O interface, recognizes a control-storage error while communicating with an attached I/O device.

'Disconnect in' can be raised by a control unit only when it is connected to the channel (that is, when 'operational in' is up). If the control unit is currently disconnected from the I/O interface, a control-unit-initiated sequence is performed to establish a connection before 'disconnect in' is raised. When 'disconnect in' is used during a control-unit-initiated sequence, 'disconnect in' rises at the control unit at least 250 nanoseconds after 'address in' rises, or subsequent to the rise of 'command out' at the control unit, whichever occurs first.

The channel, in response to 'disconnect in', performs a selective reset. 'Disconnect in' does not fall before the reset nor remain up longer than 100 nanoseconds after the fall of 'operational in'.

## COMMAND-RETRY FEATURE

Command retry is a channel and control-unit procedure that can cause a command to be retried by the channel. The command-retry procedure is initiated by the control unit with a unique combination of status bits and the use of the 'mark 0 in' line.

A control unit may request the retry of a command in order to recover from a transient error or when conditions existing at either the control unit or

I/O device prevented execution of the command when it was previously issued.

A channel, upon accepting a request for command retry, repeats the execution of the channel program, beginning at the last command accepted by the control unit.

## COMMAND-RETRY SEQUENCE

If, during execution of a command, the control unit encounters a condition requiring retry, the control unit requests command retry by raising 'mark 0 in' and 'status in' while presenting unit check and status modifier together with (1) channel end alone (meaning the control unit or the device is not yet ready to retry the command), or (2) channel end and device end (meaning the control unit and device are prepared for immediate retry of the command). Device end, if not presented with channel end, is presented later, when the control unit is ready to retry the command.

The channel acknowledges the request for command retry by indicating command chaining. If device end accompanies the command-retry request, the channel immediately initiates an initial-selection sequence and reissues the previous command. If device end does not accompany the command-retry request, command chaining is indicated, but the retry is not immediately performed. When device end or device end with status modifier is presented to the channel, command chaining is indicated, and an initial-selection sequence is performed to reissue (1) for device end, the previous command and (2) for device end with status modifier, a new command from the modified sequence of commands. (See "Status Modifier" in Chapter 2, "Operational Description.")

A channel indicates a refusal to perform a command retry by accepting the status byte without indicating chaining or by stacking the status byte and thus not indicating command chaining. The stacked byte is treated as any stacked status. When the stacked status is subsequently accepted by the channel, command chaining is again not indicated, and the command-retry procedure is not performed.

## HIGH-SPEED-TRANSFER FEATURE

Some control units have data-transfer-rate requirements that exceed the capabilities provided through the use of only 'service in' and 'service out'. When longer I/O-interface-cable lengths

are used, the high-speed-transfer feature permits data transfers to take place at higher data rates than those achievable when only 'service in' and 'service out' are used. Specifically, higher data rates are made possible by alternating 'data in' with 'service in' and 'data out' with 'service out', respectively. Such alternation is termed high-speed transfer. However, the use of this feature does not require this alternation of tag lines. Situations may occur, for example, where successive data transfers are accomplished using only 'data in' and 'data out'.

The high-speed-transfer feature may also be used to allow placement of a control unit at a greater distance from the channel than is otherwise possible.

#### DATA IN

Note: When data streaming occurs, this "Data In" section does not apply. (See "Data-Streaming Feature" in this chapter.)

'Data in' is a tag line to the channel from all control units with this feature. It is used to signal the channel that the selected control unit requires the transmission of a byte of information. The nature of the information depends on the I/O operation and the I/O device. The channel responds to the rise of 'data in' by raising either 'data out' or 'command out'.

During execution of an I/O operation specified by read, read-backward, and sense commands, the control unit places a data byte on 'bus in' and raises 'data in'. During execution of a write command or control command, the control unit raises 'data in', and the channel places a data byte on 'bus out'. When 'data in' is alternated with 'service in', 'data in' may rise when 'service out' is raised in response to 'service in'. However, 'data in' is not considered valid until 'service in' is dropped. Similarly, 'service in' may rise when 'data out' is raised in response to 'data in'; however, 'service in' is not considered valid until 'data in' is dropped.

The conditions that apply to 'service in' concerning overrun and data suppression also apply to 'data in'.

'Command out' in response to 'data in' means "stop." "Stop" is signaled for the first 'service in' or 'data in' that is provided after the channel determines that the current operation is to be ended. After "stop" is signaled, the device does not send any further 'ser-

vice in' or 'data in' signals to the channel.

#### DATA OUT

Note: When data streaming occurs, this "Data Out" section does not apply. (See "Data-Streaming Feature" in this chapter.)

'Data out' is a tag line from the channel to all attached control units and is used only in response to the rise of 'data in'. 'Data out' in response to 'data in' means "accept data" during a read, read backward, or sense command, and means "data ready" during a write or control command.

When 'data out' is sent in response to 'data in' during execution of an I/O operation specified by a read, read-backward, or sense command, 'data out' rises after the channel accepts the information on 'bus in'. In these cases, the rise of 'data out' indicates that the information is no longer required to be valid on 'bus in' and is not associated with any information on 'bus out'. When 'data out' is sent in response to 'data in' during execution of a write or control command, the rise of 'data out' indicates that the channel has provided the requested information on 'bus out'. 'Data out' remains up, and the information on 'bus out' remains valid, until the fall of 'data in'.

#### DATA-STREAMING FEATURE

The data-streaming feature allows up to a 3-megabyte-per-second data-transfer rate on a one-byte data bus and up to 122 meters (400 feet) of interface cable. Whether a control unit can use this feature is determined by the data-rate and cable-length requirements it must meet. The interlocked method of transferring data that is described elsewhere in this manual imposes the least critical timing considerations on a control unit. The data-streaming feature, which is a noninterlocked method of transferring data, requires more precise timing implementations but offers higher data rates that are independent of the cable lengths.

The data-streaming feature is applicable to any operation that initiates data transfer except the basic sense (04) command. For the duration of each connection in which data streaming occurs, 'service in' is alternated with 'data in'.

The data-streaming feature requires that both control units and channels have no

greater than  $\pm 22.5$  nanoseconds of internal skew between the rise of the service or data tag and the rise of any associated bus-bit signal.

All timings stated in the definition for out tags or 'bus out' are measured at the channel tailgate. All timings stated in the definition for in tags or 'bus in' are measured at the control-unit tailgate.

Those control units that can transfer data by using the data-streaming feature signal the channel which method of data transfer is desired each time a connection is made. The following convention is used to identify the method of data transfer:

1. If the control unit initiates data transfer by raising 'data in', a data-streaming transfer occurs.
2. If the control unit initiates data transfer by raising 'service in', data is transferred by using the interlocked method. (High-speed transfer may occur.)

The channel must have information that the control unit has the data-streaming feature to properly interpret the in tags.

The maximum rate at which data is transferred by using the data-streaming feature is model-dependent for the I/O device and for the channel implementing this feature. The maximum rate is expressed as the minimum time from the rise of 'service in' or 'data in' until the next rise of 'service in' or 'data in', respectively. In addition, the channel must ensure that the rate of out-tag responses take place within  $\pm 15\%$  of the rate that the in-tag requests are received. That is, the interval from the rise of 'service out' or 'data out' until the next rise of 'service out' or 'data out', respectively, must be within  $\pm 15\%$  of the interval from the rise of 'service in' or 'data in' until the next rise of 'service in' or 'data in', respectively, that was actually received by the channel.

**Note:** When data streaming occurs, data must be valid on the appropriate bus for the specified amount of time. If the receiver of the data is delayed in sampling the bus, data may be lost. Thus, channels and I/O devices that previously did not have a time-dependency on the receipt of data (such as buffered I/O devices) might now overrun when this feature is implemented.

## DATA TRANSFER WHILE DATA STREAMING

When data streaming occurs, the rise and fall of 'service in' or 'data in' is independent of the rise and fall of 'service out' or 'data out', respectively. Furthermore, for 'service in' or 'data in', the uptime or downtime is not less than 270 nanoseconds, respectively. For 'service out' or 'data out', the uptime or downtime for either out tag is not less than 180 nanoseconds.

During execution of an I/O operation specified by a read, read-backward, or sense command other than basic sense (04), data is valid on 'bus in' from 22.5 nanoseconds after the rise of 'service in' or 'data in' until 247.5 nanoseconds or greater after the rise of 'service in' or 'data in'. Also, the overlap between 'service in' and the next consecutive 'data in' or between 'data in' and the next consecutive 'service in' is not greater than 22.5 nanoseconds.

The channel provides a delay in the inbound tag lines to accommodate skew caused by the channel circuitry (including its receivers) and, in addition, provides a delay of at least 100 nanoseconds. This delay compensates for skew caused by the cable, and, for most control units, for the skew caused by their drivers. When a control unit and cable can cause more skew than can be accommodated by the 100-nanosecond delay, the control unit provides the additional delay to eliminate this greater skew.

During execution of an I/O operation specified by a read, read-backward, or sense command other than basic sense (04), 'service out' or 'data out' rises in response to 'service in' or 'data in', respectively, indicating that the channel has accepted the information on 'bus in'.

When, during execution of an I/O operation specified by a write or control command, 'service out' or 'data out' rises in response to 'service in' or 'data in', respectively, the channel has placed data on 'bus out' and, after the appropriate 100-nanosecond delay, has raised the corresponding out tag for not less than 180 nanoseconds. Data is valid on 'bus out' from 100 nanoseconds before the rise of 'service out' or 'data out' until 100 nanoseconds or more after the rise of 'service out' or 'data out', respectively.

For any I/O operation in which data streaming occurs, once an in tag is recognized and acted upon by the channel, the out-tag signal is dropped only after the corresponding in-tag signal has fallen. 'Service out' and 'data out' may be up concurrently and



are not up with any other out tag except when 'command out' is used to signal stop, or during an interface-disconnect sequence, when 'address out' may be up.

When the channel does not respond in time to a preceding request for service, an error condition is recognized, and the I/O operation is then ended by the control unit.

### Notes

1. If a channel determines that the I/O device is requesting service faster than the channel can respond, the channel does not respond, thus forcing the I/O device to detect an error condition. When the I/O device detects the error condition, it initiates the ending sequence. When the ending sequence is initiated by the control unit, the channel responds by accepting or stacking the status as in a normal ending sequence.
2. Not requiring an interlocked sequence for the first byte of data transfer can cause problems for channels with buffered I/O devices attached. Nonbuffered I/O devices usually have some mechanical delay before the first data-transfer sequence, and some channels may rely upon this time to initialize their data-transfer hardware. However, a buffered I/O device may require data transfer immediately and possibly cause an overrun condition to be recognized.

The channel may delay the initiation of data transfer in one of two ways. The channel may (1) delay data transfer by the use of 'suppress out' or (2) delay dropping of the out tag immediately following the completion of either the initial-selection sequence or the control-unit-initiated sequence, as applicable.

When the channel delays the initiation of data transfer by the use of 'suppress out', the channel does so as defined in "Suppress Data While Data Streaming."

When the channel delays the initiation of data transfer by delaying the dropping of the out tag immediately following the completion of either the initial-selection sequence or the control-unit-initiated sequence, it follows this procedure:

- a. After an initial-selection sequence has been completed,

the channel delays the fall of 'service out' until it is ready to transfer data. Thus, an I/O device may not assume that the channel is capable of data transfer until it has recognized the fall of the 'service out' signal that is used to respond to 'status in'. If 'suppress out' is up when 'service out' falls, indicating suppress data, an I/O device that requires data transfer may create an overrun condition.

- b. During a control-unit-initiated sequence, the channel delays the fall of 'command out' until the channel is ready to transfer data. Thus, an I/O device may not assume that the channel is capable of performing a data-transfer sequence until it has recognized the fall of the 'command out' signal that is used to respond to 'address in'.

### STOP/COMMAND OUT WHILE DATA STREAMING

The stop sequence control is indicated by 'command out' in response to 'service in' or 'data in'.

'Command out' in response to 'service in' or 'data in' is used to signal the I/O device that the channel is ending the current operation. On receipt of the stop signal, the I/O device proceeds to its normal ending point without raising either 'service in' or 'data in'. The I/O device remains busy until the ending status is accepted by the channel.

When data streaming occurs and data transfer is to be stopped, 'command out' is raised in response to the first 'service in' or 'data in' that is raised after the channel determines to end the operation. 'Command out' remains up until the fall of the associated 'service in' or 'data in'. However, the uptime of 'command out' is not less than 180 nanoseconds.

Following the signaling of stop, one or more additional 'service in' or 'data in' signals may, because of propagation delay, be received by the channel and are responded to by the appropriate 'service out' or 'data out'. For each such signal, during a read, read-backward, or sense operation other than basic sense (04), the channel does not check 'bus in' for correct parity. For each additional 'service in' or 'data in' received during write or control operations, the channel responds with the appropriate 'service out' or 'data out' signal, and 'bus out' has a byte of

all zeros with undefined parity. 'Bus out' is not checked for parity or decoded by the control unit after stop has been signaled. The total number of channel responses equals the number of I/O-device requests. If they do not compare equal, the control unit sets the unit-check status indicator.

Note: Because 'command out' is substituted for 'service out' or 'data out' when stop is signaled, 'command out', 'service out', and 'data out' may be detected in any paired combination and be active concurrently.

#### SUPPRESS DATA WHILE DATA STREAMING

For a data-streaming operation, the definition of 'suppress data' is identical to that described elsewhere in this manual, except that if the control unit disconnects from the channel during data transfer, it recognizes 'suppress out' for the first data byte after reselection. Also, data transfer does not occur until either 'suppress out' drops or until further delay would result in the detection of an overrun condition, whichever occurs first.

If data streaming occurs during a control command or a sense command other than basic sense (04), suppress data is recognized when it is signaled.

When data streaming occurs, the channel may, because of propagation delay, recognize the rise and fall of one or more 'service in' or 'data in' signals after 'suppress out' has been raised. Thus, for read, read-backward, or sense commands other than basic sense (04), the channel is capable of accepting the additional bytes of data after 'suppress out' has been signaled.

Note: Channel designs must take into account that, when suppress data is signaled, additional data-transfer requests (depending upon the I/O-interface cable length) may be received after 'suppress out' is raised. Therefore, 'suppress out' should be raised prior to the time when data transfer should stop in order to ensure that all data requested is handled at the required rate.

#### INTERFACE DISCONNECT WHILE DATA STREAMING

When data streaming occurs, interface disconnect is signaled in the same manner as when the interlocked method of transferring data is used. Also, when

data streaming occurs, the rules for maintaining the validity of data during data transfer are unchanged during the signaling of interface disconnect (1) on 'bus in' for an I/O operation specified by a read, read-backward, or sense command other than basic sense (04), or (2) on 'bus out' for an I/O operation specified by a write or control command. However, when data-streaming occurs, one or more additional 'service in' or 'data in' signals may still be received by the channel after the signaling of interface disconnect. The channel does not respond to these signals with 'service out' or 'data out'. For each 'service in' or 'data in' received during a read, read-backward, or sense operation other than basic sense (04), the channel does not check 'bus in' for correct parity. For each additional 'service in' or 'data in' received during write or control operations, the channel does not respond with 'service out' or 'data out'. When 'operational in' drops, the channel may drop 'address out' to complete the interface-disconnect sequence control.

#### SELECTIVE RESET WHILE DATA STREAMING

Following the signaling of 'selective reset', one or more additional 'service in' or 'data in' signals may, because of propagation delay, be received by the channel. In this situation, the additional 'service in' or 'data in' signals received are not responded to by the rise of 'service out' or 'data out'. For each additional 'service in' or 'data in' during execution of an I/O operation specified by a read, read-backward, or sense command other than basic sense (04), the channel does not check 'bus in' for correct parity. For each additional 'service in' or 'data in' received during execution of an I/O operation specified by a write or control command, the channel does not respond with 'service out' or 'data out', respectively.

#### RESPONSE-TIME REQUIREMENTS WHILE DATA STREAMING

When data streaming occurs, during data transfer the control unit may or may not receive the corresponding response to the last 'service in' or 'data in' in the expected time interval. To avoid potential hangup conditions, the control unit may recognize an error condition and present unit-check status if at least 8 microseconds has elapsed from the rise of the in tag that is awaiting an out-tag response.

## DYNAMIC-RECONNECTION FEATURE

The dynamic-reconnection feature provides a device with the ability to select any available I/O interface in a group of I/O interfaces for transferring information to a system. Specifically, a device that has disconnected from an I/O interface during the execution of a command can reconnect to any one of a group of I/O interfaces to continue the execution of the current command. In addition, whenever the control unit or one of its attached I/O devices recognizes an asynchronous condition, the status can be presented to the system over any I/O interface in the group.

During execution of successive I/O operations, use of the dynamic-reconnection feature occurs only when (1) the I/O device has the feature, and (2) the channel subsystem has the companion feature. The capabilities of the channel subsystem when using this feature are model-dependent for the specific system.

When multiple I/O interfaces exist between the channel subsystem and the I/O device, they can be logically arranged into a group of I/O interfaces all of which are associated with a single system. Use of the dynamic-reconnection feature can then be employed among the I/O interfaces of the group. The method for establishing a group of I/O interfaces is model-dependent for the specific I/O device.

When the dynamic-reconnection feature is used, the I/O-interface signal sequences are unmodified from the sequences described elsewhere in this manual. However, in some cases, the I/O interface protocols are modified. These changes are described in the following paragraphs and apply whenever more than one I/O interface exists in the group. When only one I/O interface is in the group, the dynamic-reconnection feature has no effect on I/O interface protocols.

When the I/O device presents status, except control-unit end alone, on any I/O interface in the I/O-interface group after the I/O device has initiated an I/O operation, it is interpreted as status resulting from the I/O operation in progress. When control-unit end alone is presented on any interface in the group after the I/O device has initiated an I/O operation, it is not interpreted as status resulting from the I/O operation in progress. When status is stacked on one of the I/O interfaces, the stacked status may be presented on any I/O interface in the group.

When system reset is signaled and an I/O operation has been initiated but is not actively in progress ('operational in'

not active) on that I/O interface, the reset signal has no effect on the I/O operation. When system reset is signaled over any I/O interface of the group and status is pending, the status remains pending and is presented on any I/O interface in the group. When system reset is signaled over an I/O interface of the group and status is being retained because it was previously stacked on that I/O interface, the stacked status is reset. If stacked status is being retained for an I/O interface of the group other than the one signaling system reset, the stacked status is not reset. When system reset is signaled over all I/O interfaces in the group while an I/O operation is in progress or when status is pending, the corresponding I/O operation or pending status, as applicable, is reset.

When selective reset is received on one I/O interface of the group while an I/O operation for the same device is actively being executed ('operational in' active) on another I/O interface of the group or while command chaining is being signaled for the same device on another I/O interface in the group, then selective reset has no effect on the I/O operation. At any other time while an I/O operation is in progress, selective reset received on any I/O interface of the group resets the I/O operation. If status is stacked when selective reset is signaled, the status may or may not be reset.

When interface disconnect is received on one I/O interface of the group while an I/O operation for the same device is actively being executed ('operational in' active) on any other I/O interface in the group or while command chaining is being signaled for the same device on another I/O interface in the group, then the interface-disconnect signal has no effect on the I/O operation. When interface disconnect is received at any other time, the device disconnects from the I/O interface, goes to its normal ending point, and, if appropriate, status information is generated and presented on any I/O interface in the group.

If a device responds to a command with busy status, it subsequently presents device end to indicate the end of the busy period. The device presents only one device-end indication per I/O interface group, regardless of the number of times it responded busy over the collection of I/O interfaces in the group. The device end may be presented on any I/O interface in the group, regardless of whether the associated device-busy indication had been signaled on that I/O interface. This device end must be accepted or reset before any new I/O operation can be accepted by the device from any I/O interface in the group.

When the control-unit-busy period no longer exists, a control unit presents a control-unit-end status indication over each I/O interface over which it previously returned status of control-unit busy. If a control unit returns a control-unit end over an I/O interface in a group, the control unit may use any legitimate address associated with the control unit, even if the device associated with the address is currently executing a command.

Whenever an error condition that is described by sense information has been

recognized, the control unit keeps available the device path over which the unit-check status was signaled until the associated sense information is reset. Prior to the reset of the sense information, any initial-selection attempts for that device on any other I/O interface of the I/O interface group receive either a control-unit-busy or device-busy indication. If the device presents device-end status separate from the presentation of channel end and unit check, the device-end status must be presented over the I/O interface on which the unit check was presented.

## APPENDIX A. ELECTRICAL SPECIFICATIONS

### PHYSICAL REQUIREMENTS

#### MULTIPLE DRIVERS AND RECEIVERS

Up to 10 receivers are driven by one driver. The driver is located at one of the extreme ends.

Up to 10 drivers can be dot-ORed to drive one receiver. The receiver is located at one of the extreme ends.

Receivers are spaced at least 914 millimeters (36 inches) apart. No minimum requirement is set for the spacing between drivers. No minimum requirement is set for the spacing between a terminator and driver or receiver if the terminator is placed on the outermost end of the line.

The maximum stub length from the line to a driver or receiver on the circuit card is 152 millimeters (6 inches).

**Note:** An end-of-line driver or receiver may be placed beyond the terminator. In this case, the maximum distance between the end-of-line driver or receiver and the terminator is 152 millimeters (6 inches).

### GENERAL ELECTRICAL REQUIREMENTS

#### VOLTAGE LEVELS

The dc voltage consists of two logical levels. A dc line voltage of +2.25 volts or more denotes a logical one state, and a dc line voltage of +0.15 volt or less denotes a logical zero state. These voltages are relative to the driver ground.

#### CABLE

All lines have a characteristic impedance of  $92 \pm 10$  ohms and, with the exception of 'select out', are terminated at each extreme end in their characteristic impedance by a terminating network. (For 'select out'/'select in', see "Electrical Specifications for Select-Out Circuitry.")

Cable length may be limited by special conditions but never exceeds a maximum

line resistance of 33 ohms. The 33-ohm line resistance includes all contact resistance, internal cable resistance, and external cable resistance.

#### TERMINATING NETWORKS

The terminating network presents a resistance of 95 ohms  $\pm 2.5\%$  between the signal line and ground and is capable of dissipating 390 milliwatts.

#### GROUND SHIFT AND NOISE

The maximum noise (measured at the receiver input) coupled to any signal line does not exceed 400 millivolts.

The maximum allowed ground shift between any active driver and any receiver of the same interface line is 150 millivolts. Therefore, the maximum shift (coupled noise plus ground shift) allowed on any line is 550 millivolts. The logical levels defined in "Voltage Levels" and the receiver threshold levels specified in "Receivers" under "Interface Circuit Requirements" allow for this 550-millivolt shift. That is, a negative noise pulse of 400 millivolts coupled with a positive receiver ground shift of 150 millivolts occurring during a one state (+2.25 volts minimum) guarantees a receiver input of +1.70 volts or more. (See Figure A-1.)

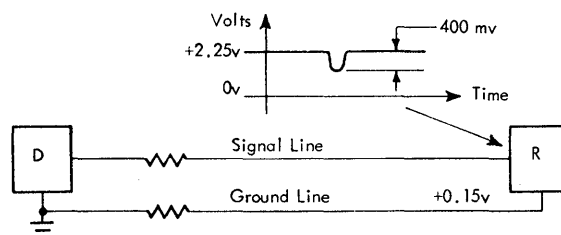


Figure A-1. Negative Noise

Also, a positive noise pulse of 400 millivolts coupled with a negative receiver ground shift of 150 millivolts occurring during a zero state (+0.15 volt maximum) guarantees a receiver input of +0.70 volt or less. (See Figure A-2.)

**Note:** The noise measurements are made at the input to the receiver. A combination of the normal dc level plus

ground shift plus noise does not exceed +0.70 volt for the down level and is not less than +1.70 volts for the up level.

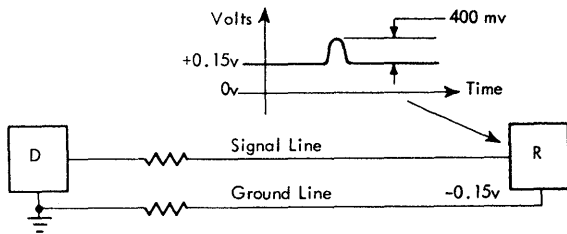


Figure A-2. Positive Noise

### INTERFACE-CIRCUIT REQUIREMENTS

#### RECEIVERS

An input voltage (relative to receiver circuit ground) of 1.70 volts or more is interpreted as a logical one; an input of +0.70 volt or less is interpreted as a logical zero. An open-circuited input is interpreted as a logical zero.

The receiver is not damaged by:

1. A dc input of +7.0 volts with power on in the receiver
2. A dc input of +6.0 volts with power off in the receiver
3. A dc input of -0.15 volt with power on or off

To reduce the loading effect on the line, the receiver input resistance is larger than 7,400 ohms across the input voltage range of +0.15 volt to +3.9 volts, and the negative receiver input current does not exceed -240 microampere at an input voltage of +0.15 volt (See Figure A-3 for the definition of receiver-current polarity.)

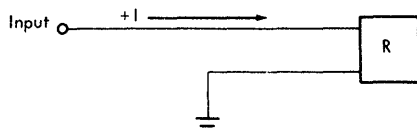


Figure A-3. Receiver-Current Polarity

Receivers are designed to ensure that no spurious noise is generated on the line during a normal power-on or power-off sequence.

#### DRIVERS

In the logical-zero state:

1. The output voltage does not exceed +0.15 volt at a load of +240 microamperes. (See Figure A-4 for the definition of driver-current polarity.)

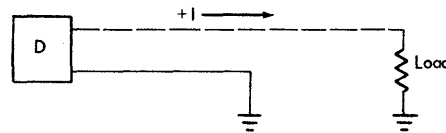


Figure A-4. Driver-Current Polarity

In the logical-one state:

1. The output voltage is +3.11 volts or more at a load of +59.3 milliamperes.
2. The output voltage does not exceed +5.85 volts at a load of +30 microamperes.
3. The output voltage does not exceed +7.0 volts at a load of +123.0 milliamperes during an over voltage internal to the drivers.

Drivers are designed to ensure that no spurious noise is generated on the line during a normal power-on or power-off sequence. For the driver, this may be accomplished by one of the following methods:

1. Sequencing the power supplies.
2. Building noise suppression into the circuit.
3. Providing an externally controlled gate. (See Figure A-5).

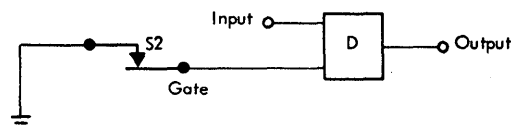


Figure A-5. Driver Gate

For a normal power-off sequence:

1. Logically ensure that the driver is in the zero state.

2. Close contact S2. (See Figure A-5.)
3. Turn power off.

For a normal power-up sequence:

1. Ensure that contact S2 is closed.
2. Turn on power.
3. Logically ensure that the input level causes the driver output to be in the zero state.
4. Open contact S2.

### FAULT CONDITIONS

A grounded signal line does not damage drivers, receivers, or terminators. A signal line may be grounded (shorted) at any point along its length. This means the resistance value of a shorted signal line may be anywhere between 24.5 ohms and 0.0 ohms with respect to ground.

With one driver transmitting a logical one, loss of power in any single circuit driver, receiver, or terminator on the line does not cause damage to other components.

With both terminators connected, line operations are not affected by power off in any drivers or receivers on the line.

### CIRCUITS

Figures A-6 through A-8 show representative circuits used to drive, receive, and terminate the lines between the channel and attached control units.

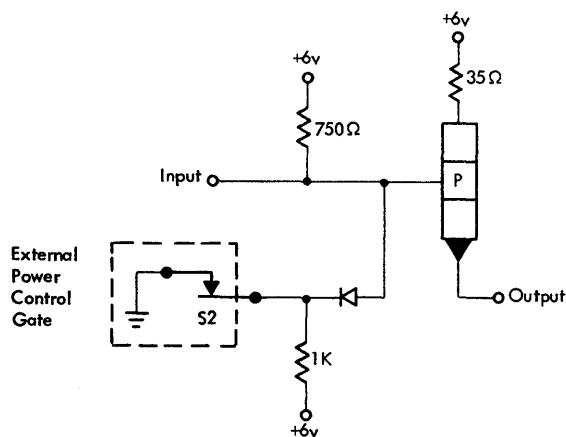


Figure A-6. Line Driver

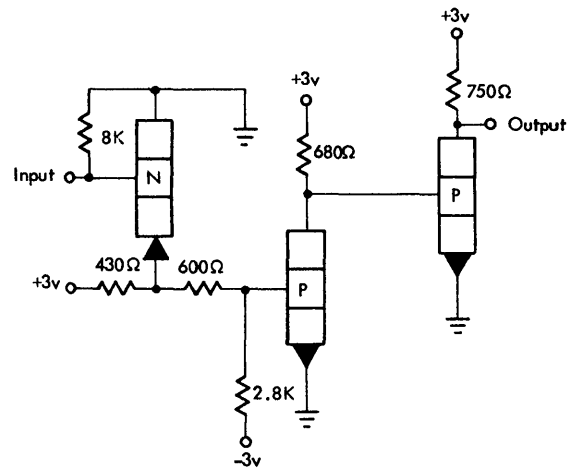


Figure A-7. Line Receiver

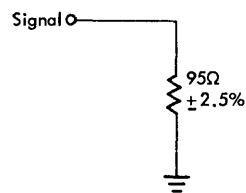


Figure A-8. Line Terminator

### ELECTRICAL SPECIFICATIONS FOR SELECT-OUT CIRCUITRY

#### GENERAL

The 'select out' line has a single-driver to single-receiver configuration, with only the receiver end of the line terminated in the characteristic impedance.

A dc line voltage of +1.85 volts or more denotes a logical-one state, and a dc line voltage of +0.15 volt or less denotes a logical-zero state. These voltages are relative to the driver ground.

All electrical requirements specified in "General Electrical Requirements" that are not redefined in this section are also applicable to 'select out'

**Note:** Because of the nature of the 'select out'/'select in' line, a negative noise tolerance is unnecessary.

## RECEIVER

The 'select out' receiver satisfies all requirements given in "Receivers" under "Interface-Circuit Requirements."

## DRIVER

The 'select out' driver can withstand a short-circuit-to-ground output condition, while in either the logical one or zero state, without damage to the driver circuit.

For the logical-zero state:

1. The output voltage of a 'select out' driver does not exceed +0.15 volt at a load of 1 milliamperes.

For a logical-one state:

1. Output voltage of a channel driver or the driver of a control unit contained within a channel frame exceeds +3.9 volts at a load of 41 milliamperes.
2. The output voltage of a control-unit driver not contained within a channel frame exceeds +3.7 volts at a load of 41 milliamperes.

The output voltage of a 'select out' driver does not exceed:

1. +5.8 volts at a load of 0.3 milliamperes.
2. +7.0 volts at a load of 72 milliamperes during an overvoltage internal to the driver.

## TERMINATOR

A 95-ohm  $\pm 2.5\%$ , 390-milliwatt terminator to ground is placed at each receiver for each line segment along the 'select out'/'select in' path, including the receiver end of 'select in' located in the channel.

The driver end of each segment of 'select out'/'select in' is not terminated, including the driver end of 'select out' located in the channel. Also, the jumpered 'select out' or 'select in' and the bypassed 'select out' or 'select in' path is not terminated.

## MEASURING I/O-INTERFACE TIMINGS

The definition of the I/O interface specifies several different timings. These timings are measured in the following manner.

### UPTIME

The uptime is measured from when the signal goes above the least-positive-up level (LPUL) to when it goes below the LPUL. (See Figure A-9.) The LPUL for an active driver at its output is +3.11 volts at +59.3 milliamperes. When the signal is measured at the unit's external connector, the voltage drop caused by the +59.3 milliamperes flowing through the internal cable should be taken into account and the measurement voltage of +3.11 adjusted accordingly. This LPUL value ensures that the input to all receivers on the interface meets or exceeds the requirements of the timing diagrams when measured at the receiver's LPUL of +1.70 volts.

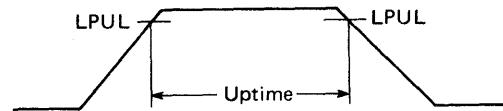


Figure A-9. Uptime

### DOWNTIME

The downtime is measured from the point at which the signal goes below the most-positive-down level (MPDL) to the point at which it goes above the MPDL. (See Figure A-10.) The MPDL for an active driver is +0.15 volt. This MPDL value ensures that the input to all receivers on the interface meets or exceeds the requirements of the timing diagrams when measured at the receiver's MPDL of +0.70 volt.



Figure A-10. Downtime



## VALID DATA

Valid data is measured from when the signal goes above the LPUL for a logical one or below the MPDL for a logical zero.

## OVERLAP NOT GREATER THAN

The point of measurement of an "overlap not greater than" is from when the first signal goes above the LPUL to when the second signal goes below the LPUL. (See Figure A-11.)

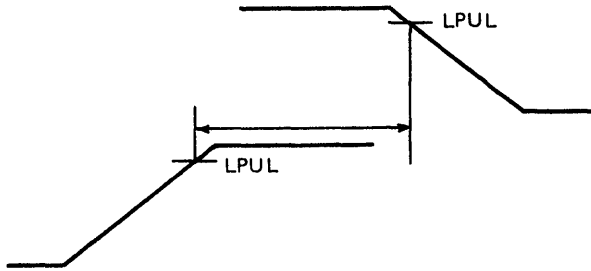


Figure A-11. Overlap Not Greater Than

## 100-NANOSECOND DELAY

The 100-nanosecond delay is measured from when the data is valid until the

corresponding out tag goes above the MPDL. (See Figure A-12.)

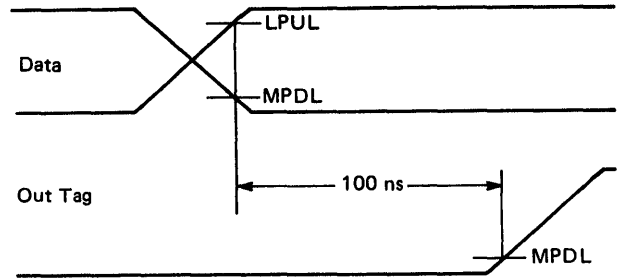
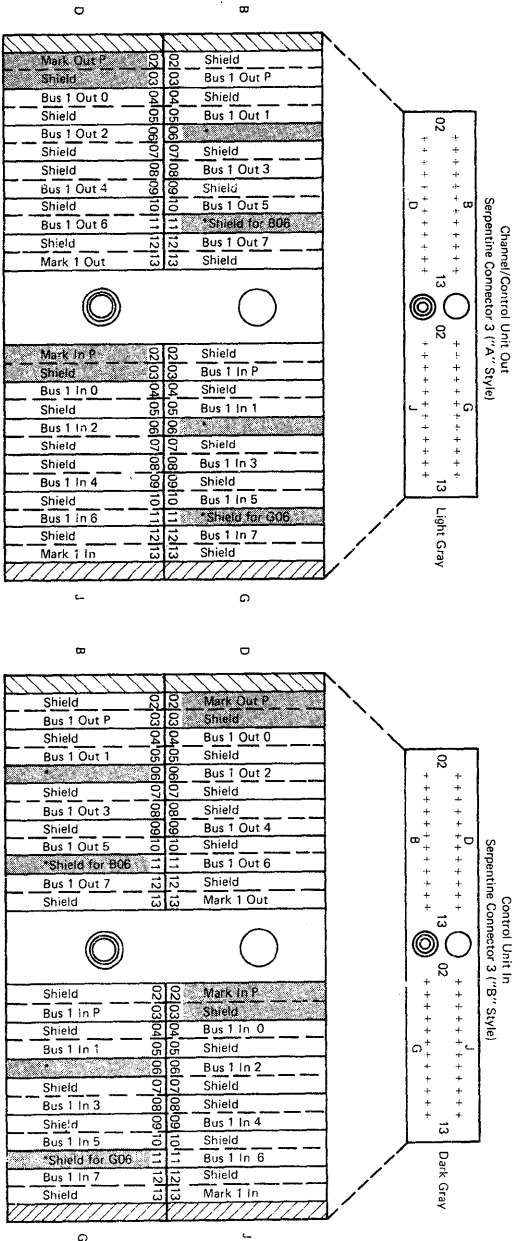
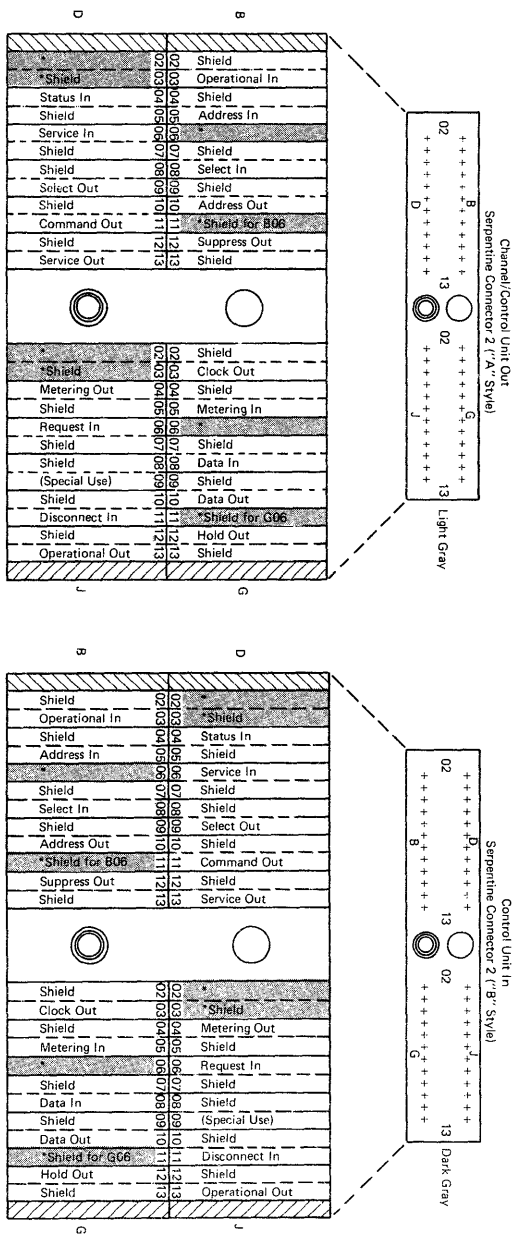
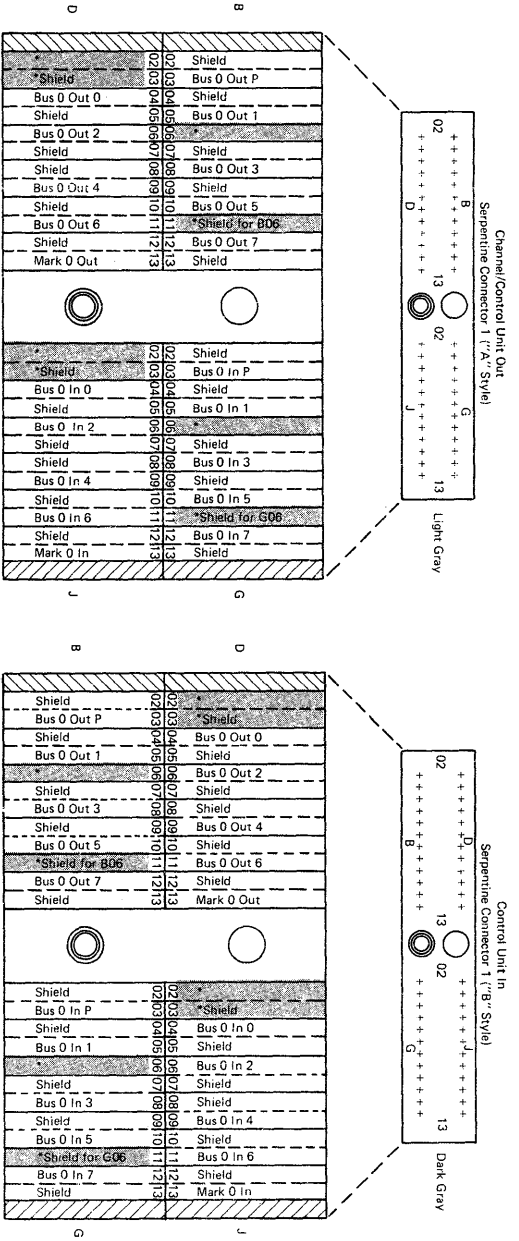


Figure A-12. 100-Nanosecond Delay

## INTERFACE-CONNECTOR PIN ASSIGNMENTS

Pin assignments for the I/O interface are shown in Figure A-13 as viewed from the connector side of the channel and control-unit tailgates. Two interface cables (three cables for the optional bus-extension feature) connect the channel to the first of a group of control units. Signals are assigned corresponding pin numbers on the channel and control-unit connectors (see Appendix B).



\* Reserved for future use.  
Note: Shading indicates those signals passed only with 48-position cable.

Figure A-13. Interface-Connector Pin Assignment

## APPENDIX B. SIGNAL CABLING AND CONNECTORS

### CABLING

#### CABLE HALVES

A screwdriver is required for joining the cable halves. The connecting screw is spring-loaded to prevent damage to mating block contacts. Blocks should be aligned to ensure proper parallel contact mating. After the blocks are keyed, push the screw forward to engage the insert; then tighten the screw securely. For assistance in removing individual contacts, field personnel should use a No. 12 crocheting needle, available from IBM Field Engineering as part 450540.

#### MOUNTING

In normal applications, the blocks are mounted in the horizontal plane to provide a smoother bend into the coaxial cables. (See Figure B-1.) Vertical mounting produces unusual bending configurations that require careful routing and strain relieving of the external cables.

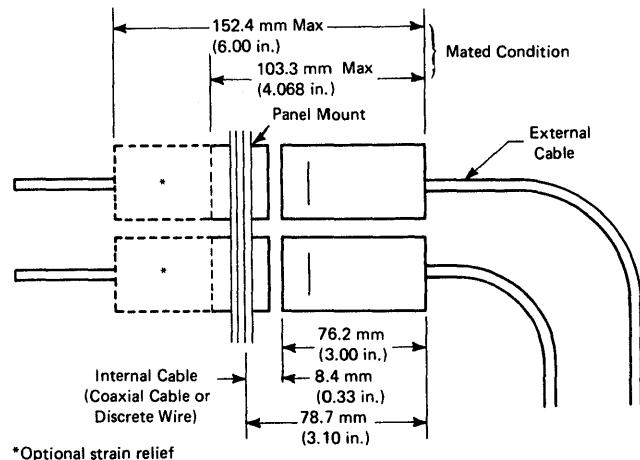


Figure B-1. Mounted Cable Halves

Figure B-2 shows an exploded view of the panel mounting.

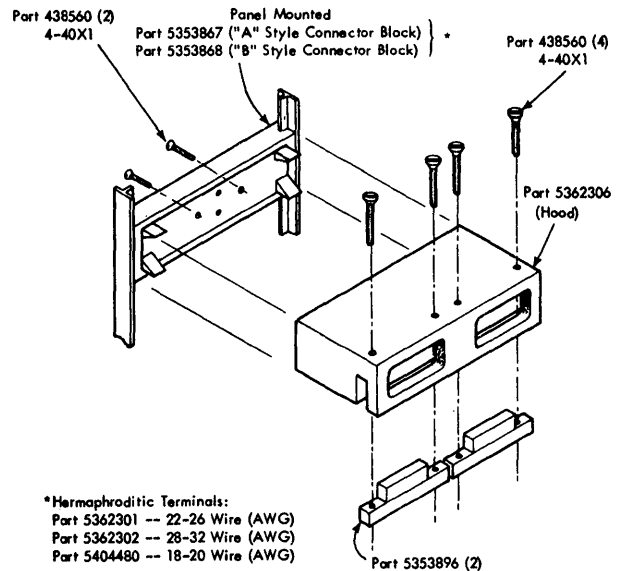


Figure B-2. Panel Mount with Optional Strain Relief

#### OFFLINE UTILIZATION

This cable connector has the feature of being able to mate "cable half" to "cable half" for offline utilization or for the physical bypassing of machine units. (See Figure B-3.)

When mating cable halves, it is only necessary to use the screw on the "B-" connector. This allows the "A" connector's screw to fall within the empty insert location in the "B-" block and permits easier connection.

Typical connections are shown in Figures B-3 and B-4. (Information concerning the power-control-interface cable shown in Figure B-4 is contained in the System Library manual IBM System/360 and System/370 Power Control Interface Original Equipment Manufacturers' Information, GA22-6906.)

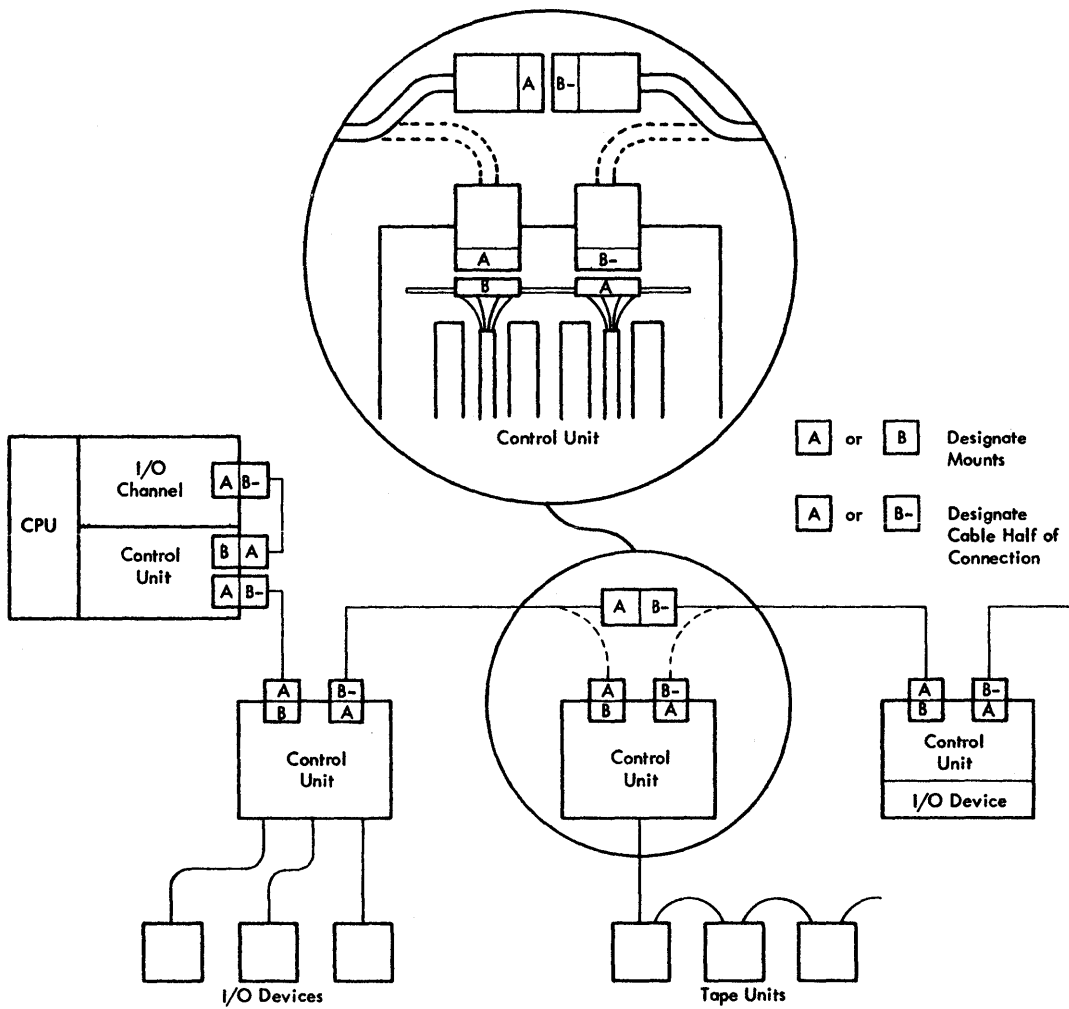
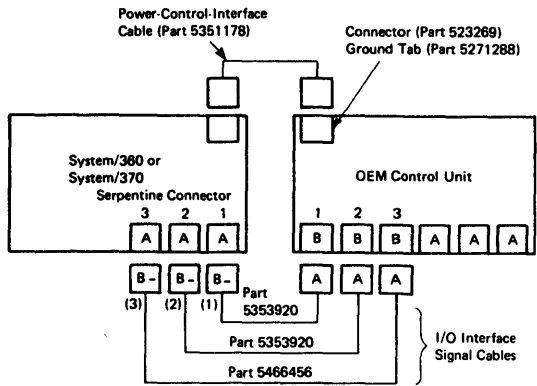


Figure B-3. Offline Utilization



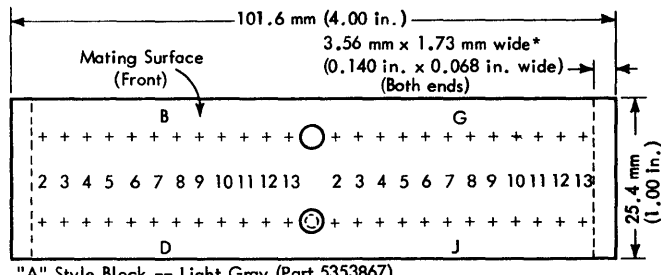
- A** Panel Mount (Part 5353867)
- B** Panel Mount (Part 5353868)

Cables (1) and (2), (40-position cable Part 5353920), are required in all cases. Cable (3), (48-position cable Part 5466456), is required for the bus-extension feature. (Cable Part 5466456 may also be used in the cable (1) and cable (2) positions. See "Reserved Lines" in Chapter 1, "Functional Description.")

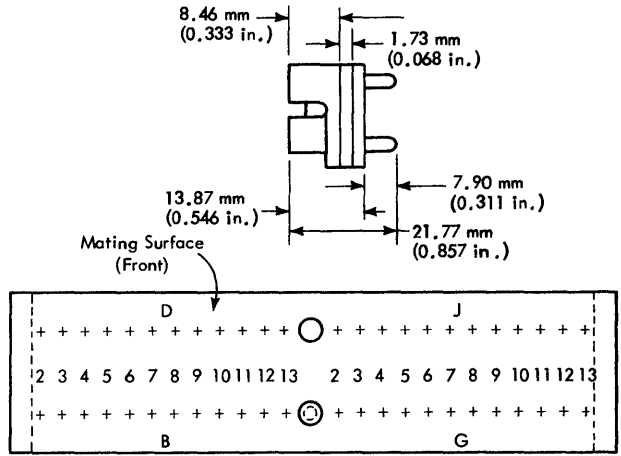
Terminals:	
Power-Control-Interface Cable Connector (6 Required)	14-18 AWG - Part 523267 20-22 AWG - Part 5351177
Signal Cable Panel Mount (40 Required per Connector for Cable 5353920; 48 Required per Connector for Cable 5466456)	22-26 AWG - Part 5362301 28-32 AWG - Part 5362302 18-20 AWG - Part 5404480

Figure B-4. Typical Connection

the connector blocks. Application of the connector is limited only by the number of coaxial wires (shielded wires or twisted pairs) used and the method used to ground the shields of the coaxial wires. In the I/O interface, for connectors 1 and 2, 40 serpentine contacts are required on each connector for individually terminating the shield and signal wires of 20 coaxial wires. For connector 3, 48 serpentine contacts are required for terminating the shield and signal wires of 24 coaxial wires. (See "Interface-Connector Pin Assignments" in Appendix A.)



"A" Style Block -- Light Gray (Part 5353867)



"B" Style Block -- Dark Gray with Threaded Insert (Part 5353868)  
"B-" Style Block -- Dark Gray, No Threaded Insert (Part 5353869)

\*Mounting Slots

Figure B-5. Connector Blocks and Contact Location

**CONNECTORS**

**CONNECTOR BLOCKS**

Three styles of connector blocks are available: "A," "B," and "B-." The "A" and "B" designations are used to identify correct mating arrangements because the physical hardware is identical. The two styles are differentiated by the color coding of the blocks: the "A" is light gray, and the "B" is dark gray. (See Figure B-5.)

The "B-" block is the same as the "B" but does not have a threaded insert.

When mating connectors, take care to prevent accidental mismatching of two "A" or two "B" connectors since letter positions would then be transposed. Connectors of the same color are never mated.

Both the "A" and "B" blocks are used for panel mounts. The "A" may also be used for a cable end. The "B-" is used only as a cable end. (See "Offline Utilization" under "Cabling.")

**I/O CABLE CAPACITY**

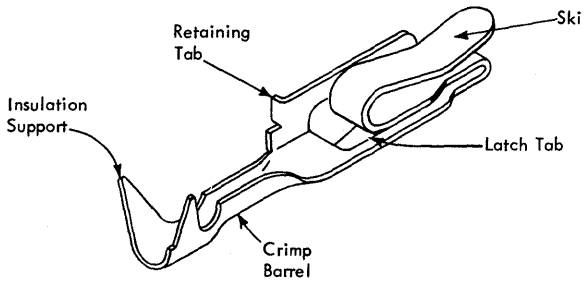
Forty-eight individual positions (serpentine contacts) are provided for in

**SERPENTINE CONTACTS**

Description

The serpentine contact is a hermaphroditic, gold-plated phosphor bronze, dual-mating-surface contact. (See Figure B-6.)

A. Part Identification

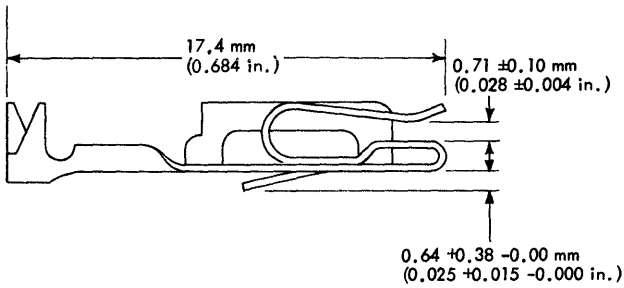


Wire Termination

Terminations are made by the bare-wire crimp method. Three contacts are available to cover the range of solid or stranded wire sizes required:

<u>IBM Part</u>	<u>Wire Size (AWG)</u>	<u>Insulation Range</u>
5404480	18-20	0.042 to 0.103
5362301	22-26	0.028 to 0.103
5362302	28-32	0.026 to 0.090

B. Dimensions



C. Mating Configuration

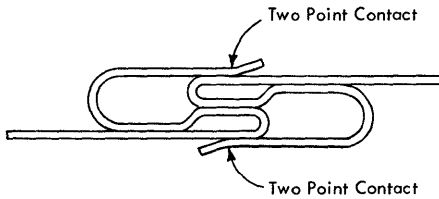
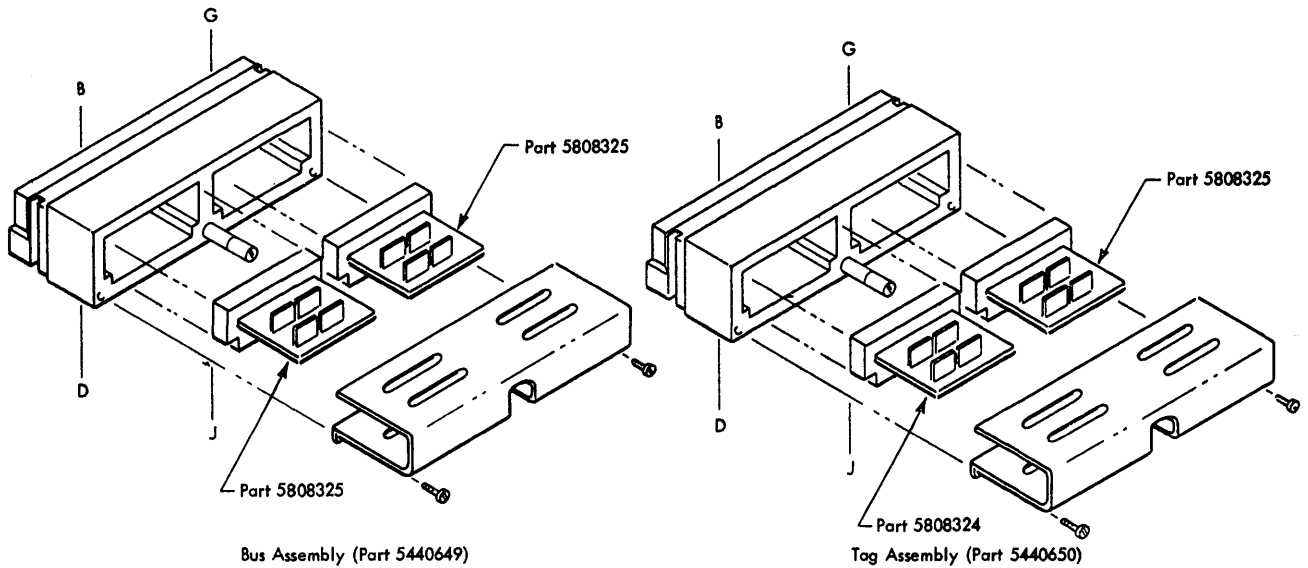
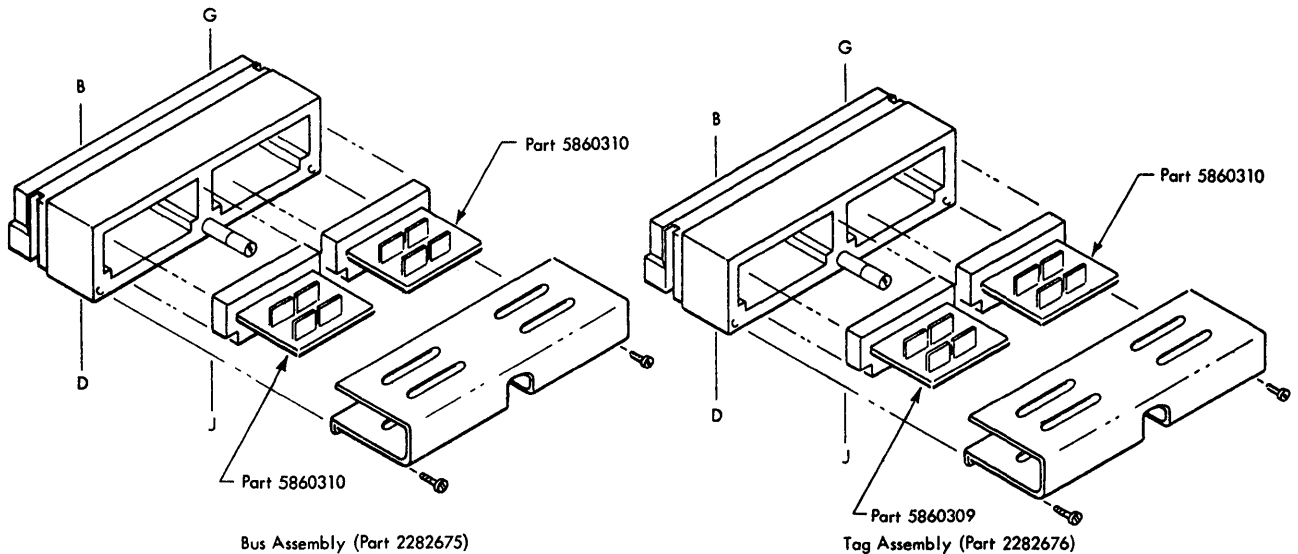


Figure B-6. Serpentine Contact



These assemblies are used with cable Part 5353920.



These assemblies are used with cable Part 5466456.

Figure B-7. Terminator Assemblies

#### ELECTRICAL SPECIFICATIONS

**Voltage Rating:** The maximum voltage rating of this connector is 24 volts ac or dc. For applications above 24 volts, contact the local IBM representative.

**Current Rating:** The maximum continuous current rating of each contact is 6 amperes. The contacts are not intended for interrupting current.

**Resistance:** The termination-to-termination resistance (includes 2 crimps and mated contacts) does not exceed:

1. 0.020 ohms when installed on #22 AWG and larger wire.
2. 0.030 ohms when installed on #24-#26 AWG wire.
3. 0.040 ohms when installed on #28-#32 AWG wire.

**Insulation Resistance:** The contact-to-contact insulation resistance is 100 megohms (minimum) measured at a test potential of 100 volts dc, after exposure of one hour at a temperature of 38° C and 85-90% relative humidity.

Grounding: If all surfaces of the cable connectors are nonconductive plastic, then no grounding is necessary. If any part of the surface of the cable connectors is conductive, then these surfaces must be grounded.

assemblies shown in Figure B-7.

A bus assembly is used to terminate cable (1) and, if the bus-extension feature is used, cable (3). A tag assembly is used to terminate cable (2).

#### TERMINATORS

The System/360 and System/370 I/O interface line termination is provided by the



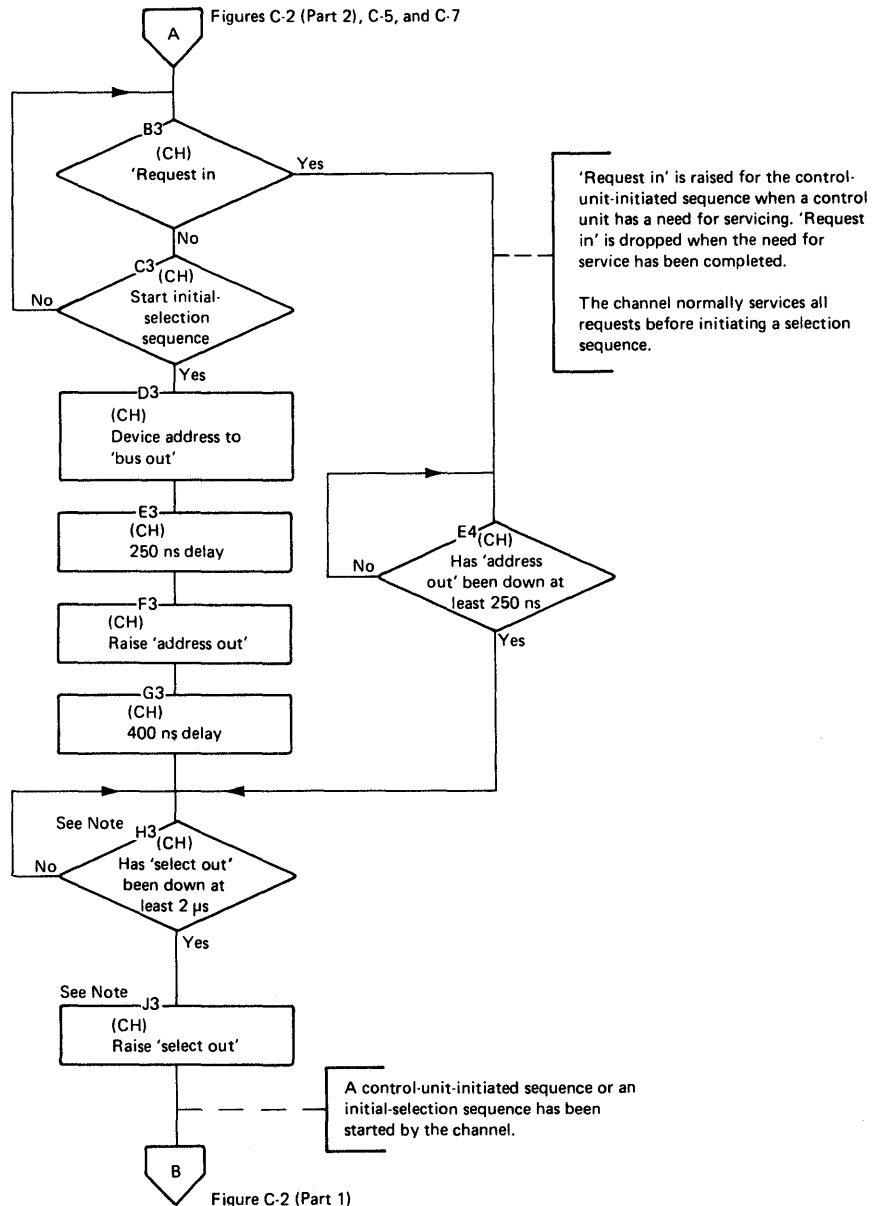
## APPENDIX C. SUPPLEMENTARY APPLICATION INFORMATION

The diagrams in this appendix are examples only and do not constitute a precise description of the I/O-interface architecture.

In the flowcharts of this appendix, the decision block involving signal lines

(with few exceptions) poses the following question: Is the line logically up?

The presence of CH (channel) or CU (control unit) in a block specifies responsibility for the action taken or decision made.



Note: All mention of 'select out' assumes proper operation of 'hold out' as well. Therefore, 'select out' up means 'select out' and 'hold out' are up, and 'select out' down means either 'select out' or 'hold out' is down.

Figure C-1. Initiation of Polling or Selection

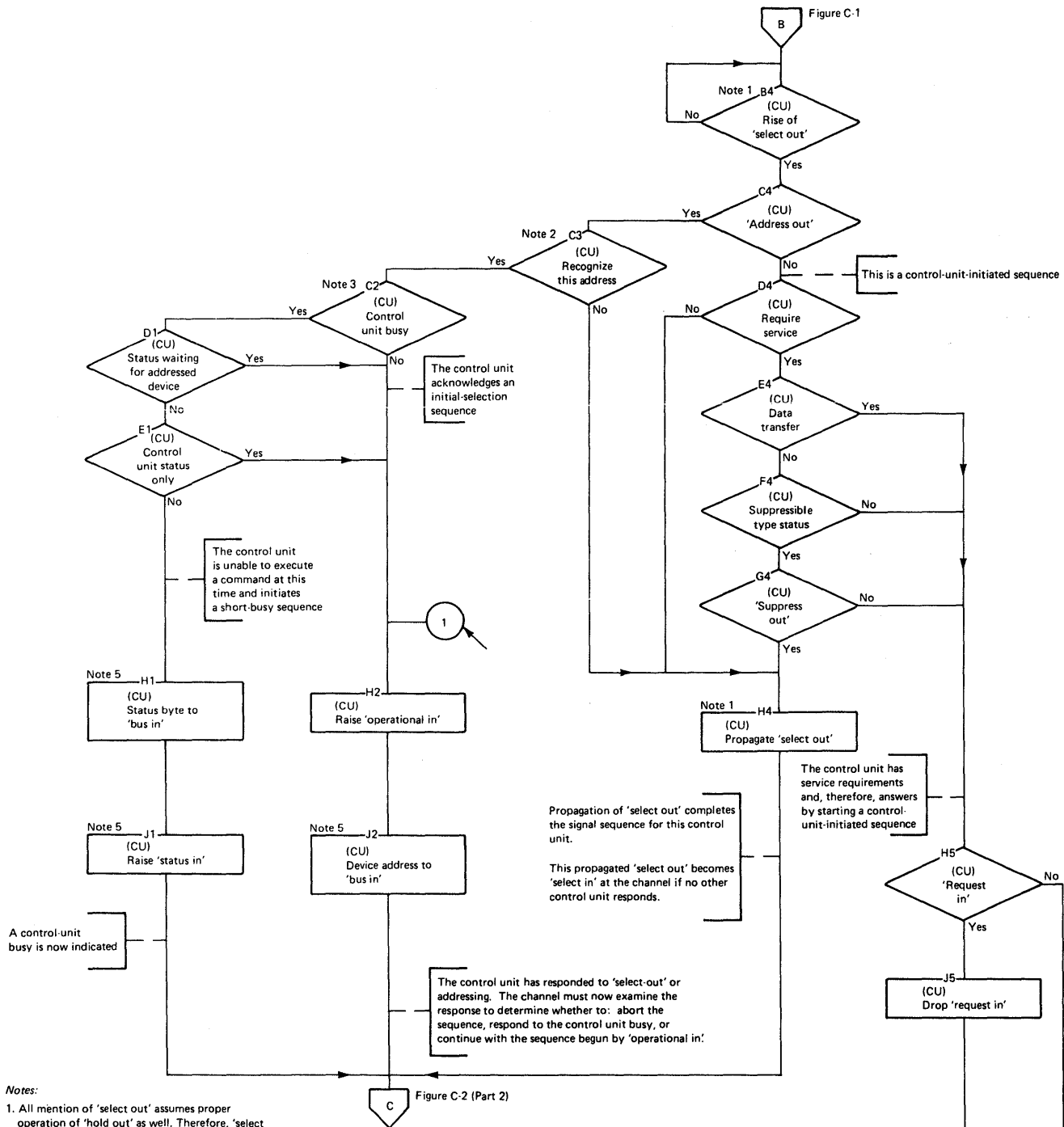


Figure C-2 (Part 1 of 2). Control Unit Response to Select Out

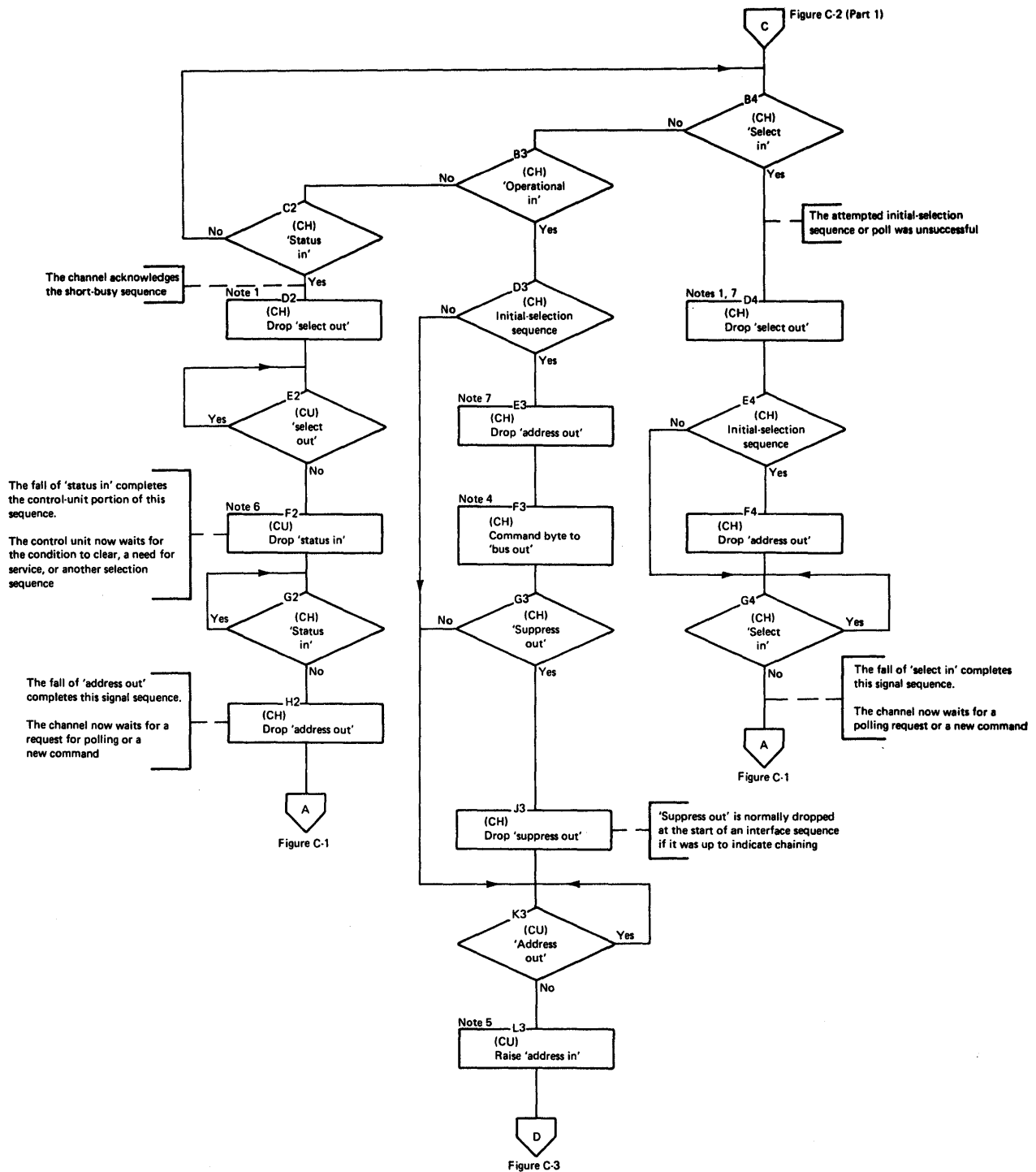


Figure C-2 (Part 2 of 2). Control Unit Response to Select Out

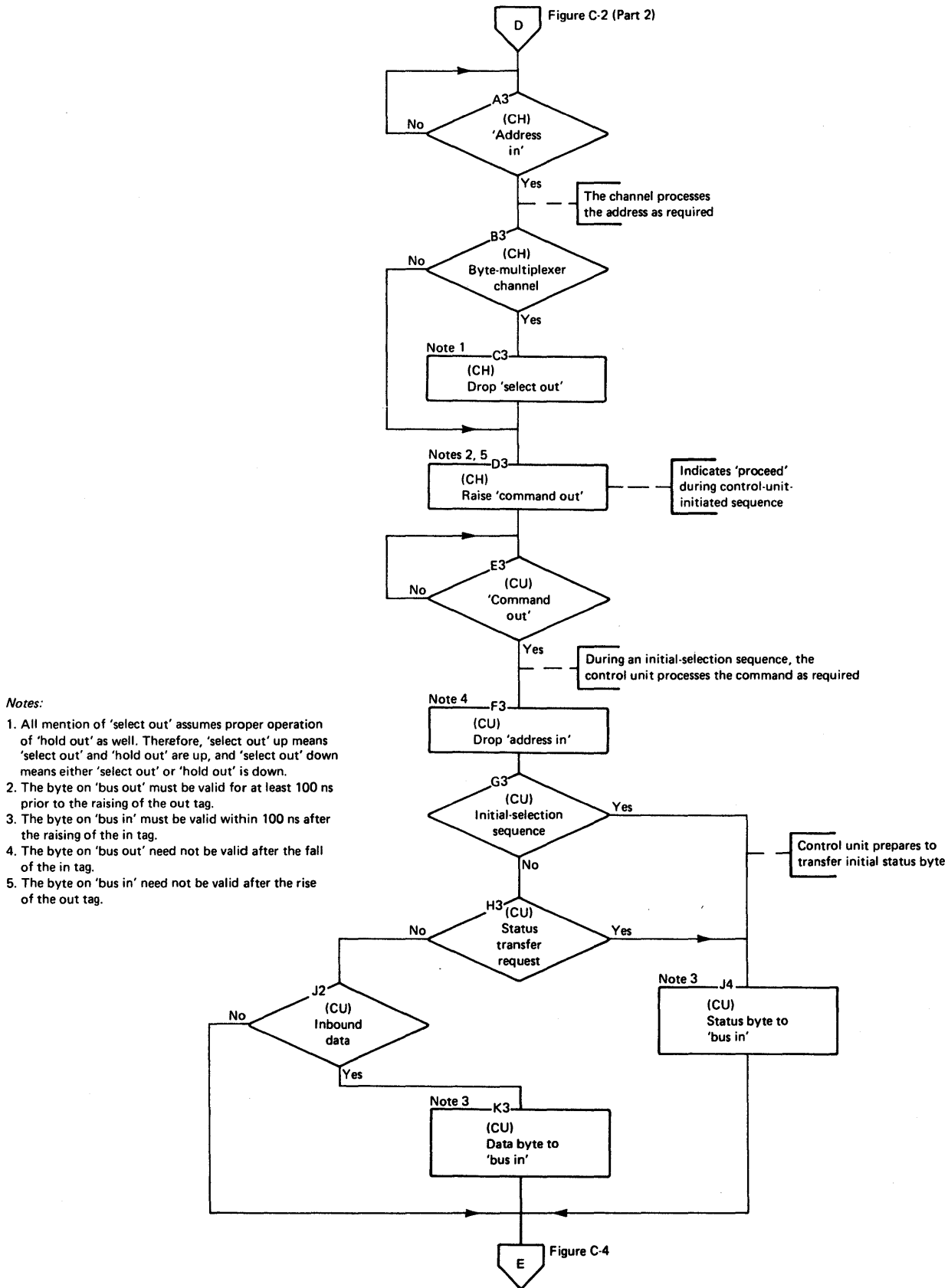
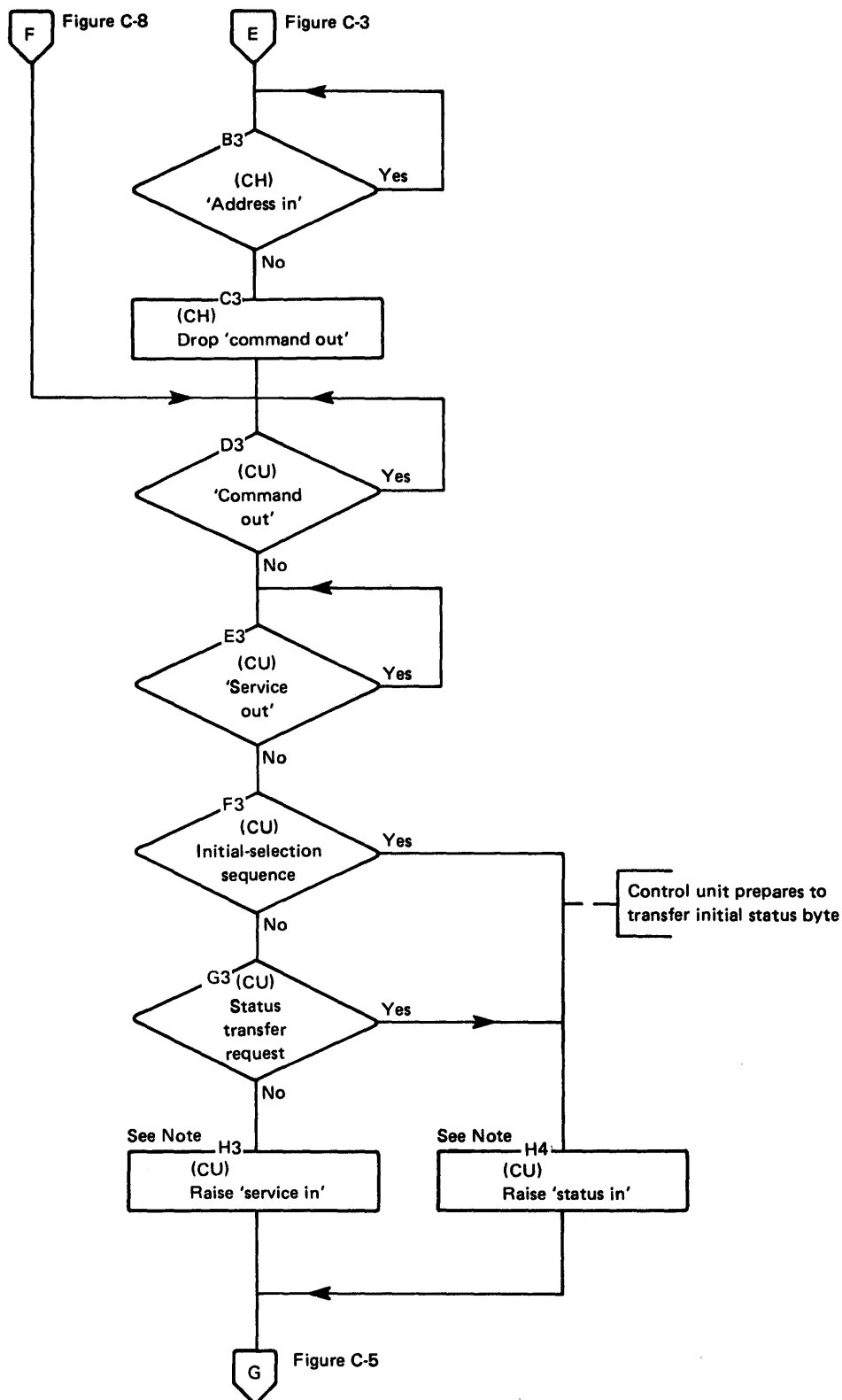


Figure C-3. Command Transfer



Note: The byte on 'bus in' must be valid within 100 ns after the raising of the in tag.

Figure C-4. Status/Data Presentation

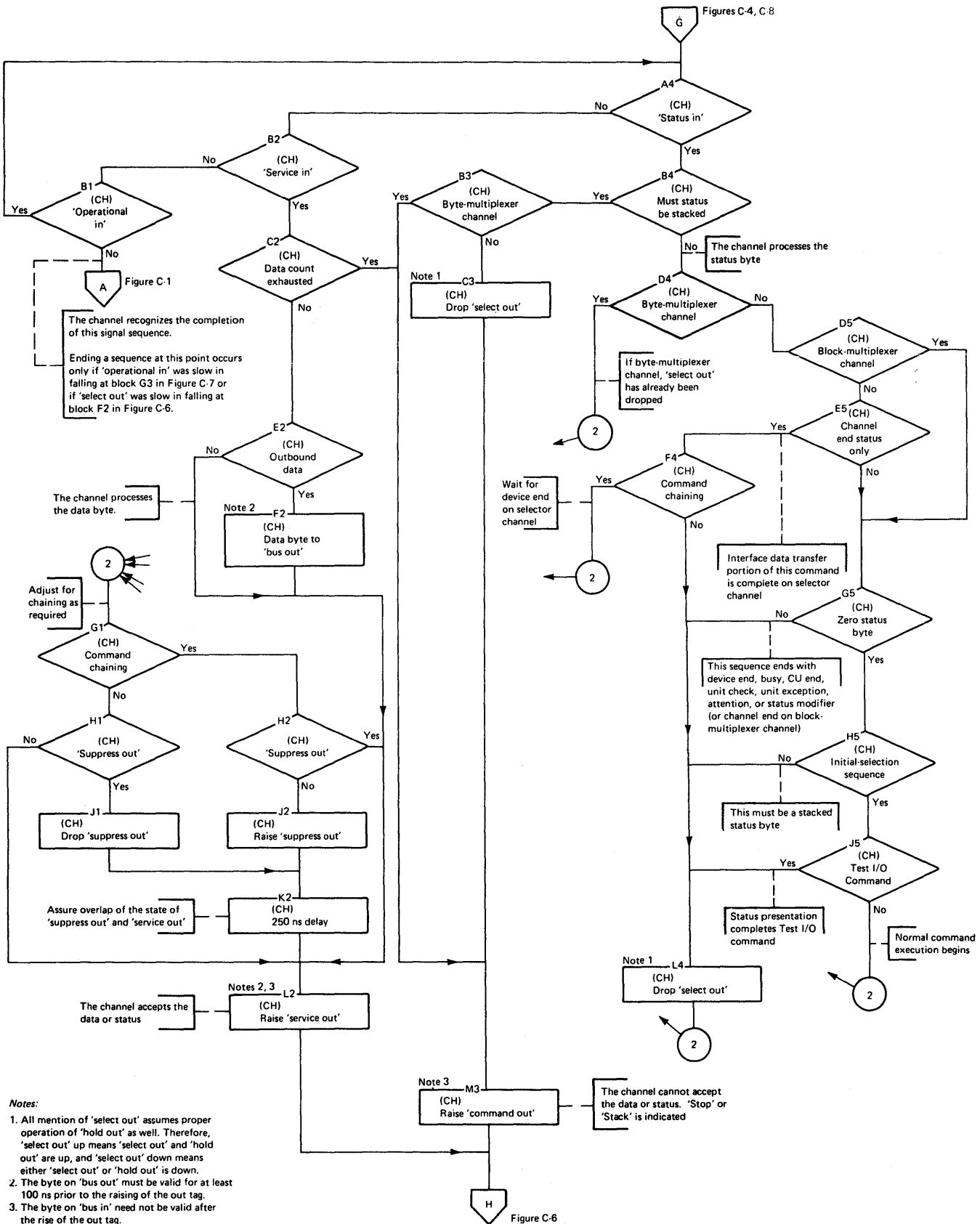


Figure C-5. Response to Status/Data Presentation

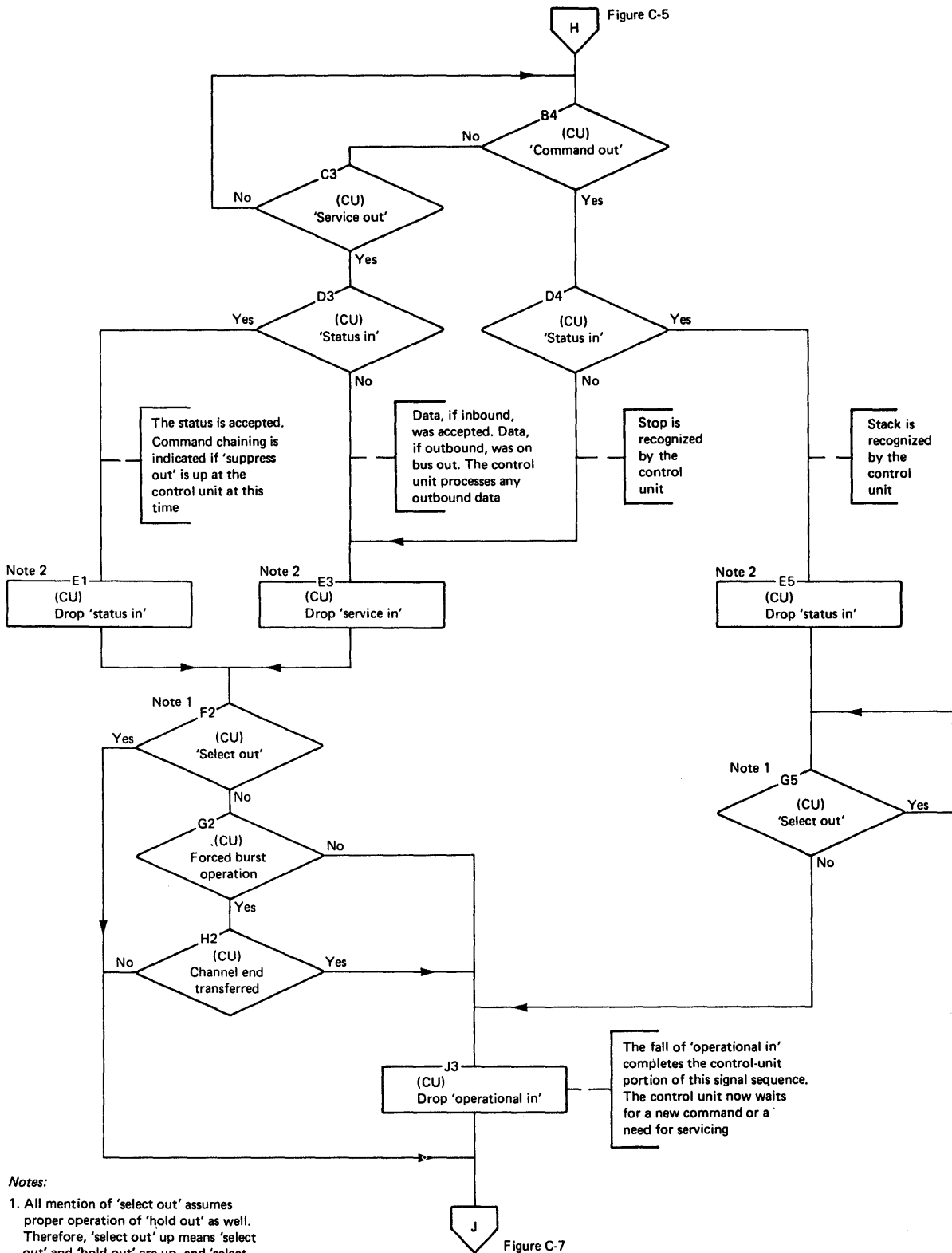
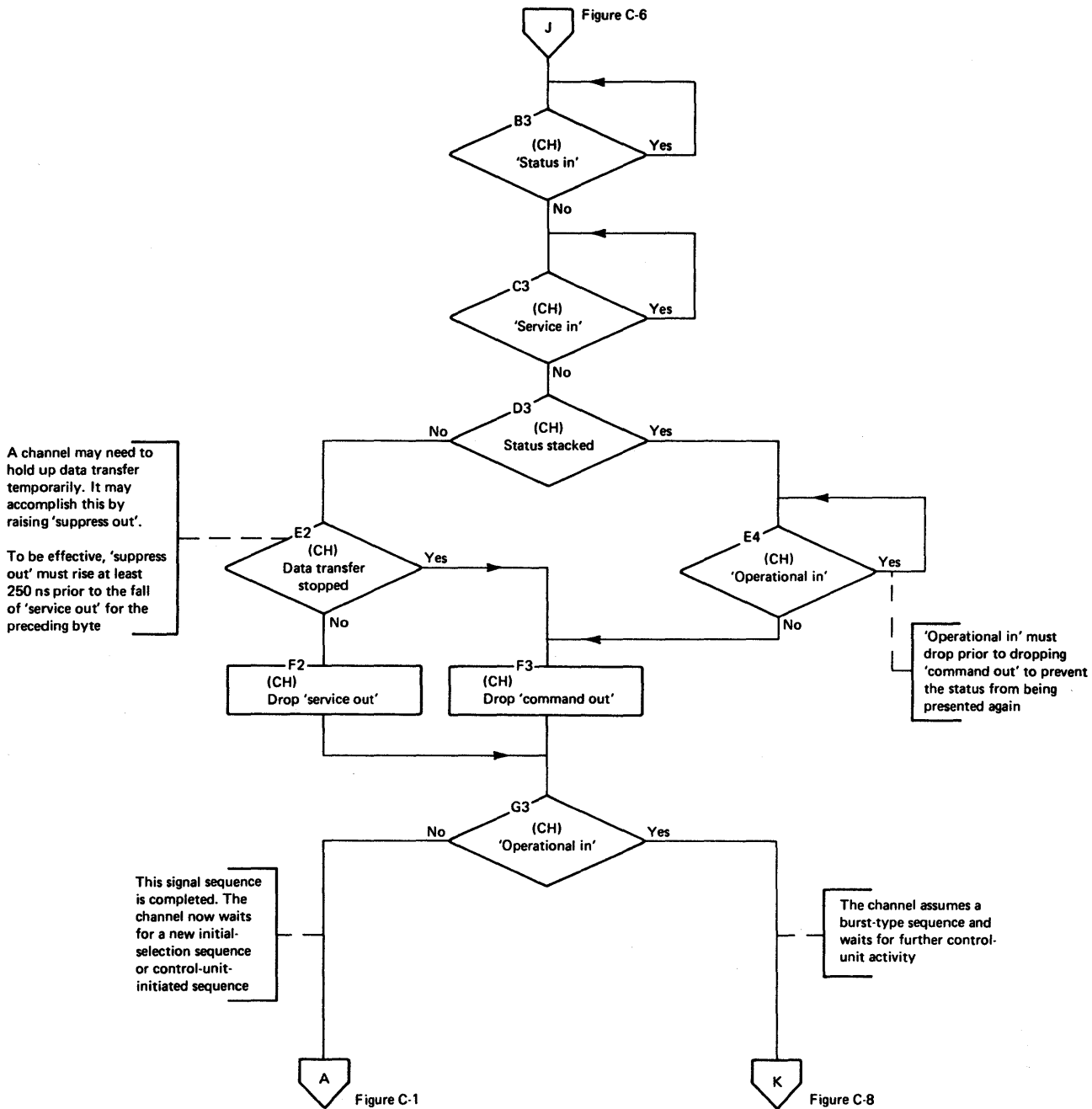


Figure C-6. Response to Stack/Stop/Accept



**Figure C-7. Response to Fall of Status In or Service In**



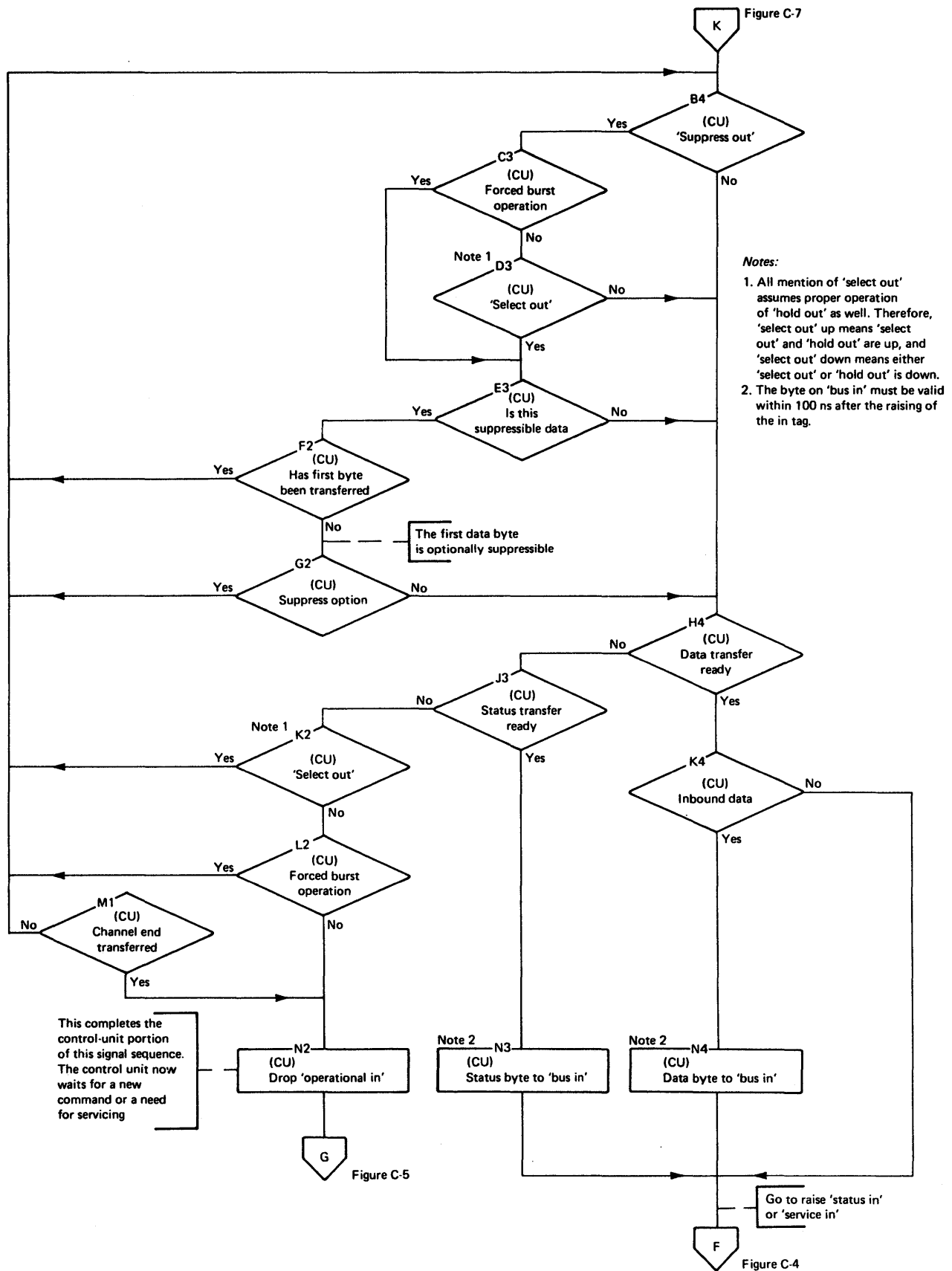


Figure C-8. Burst Mode Waiting Loop



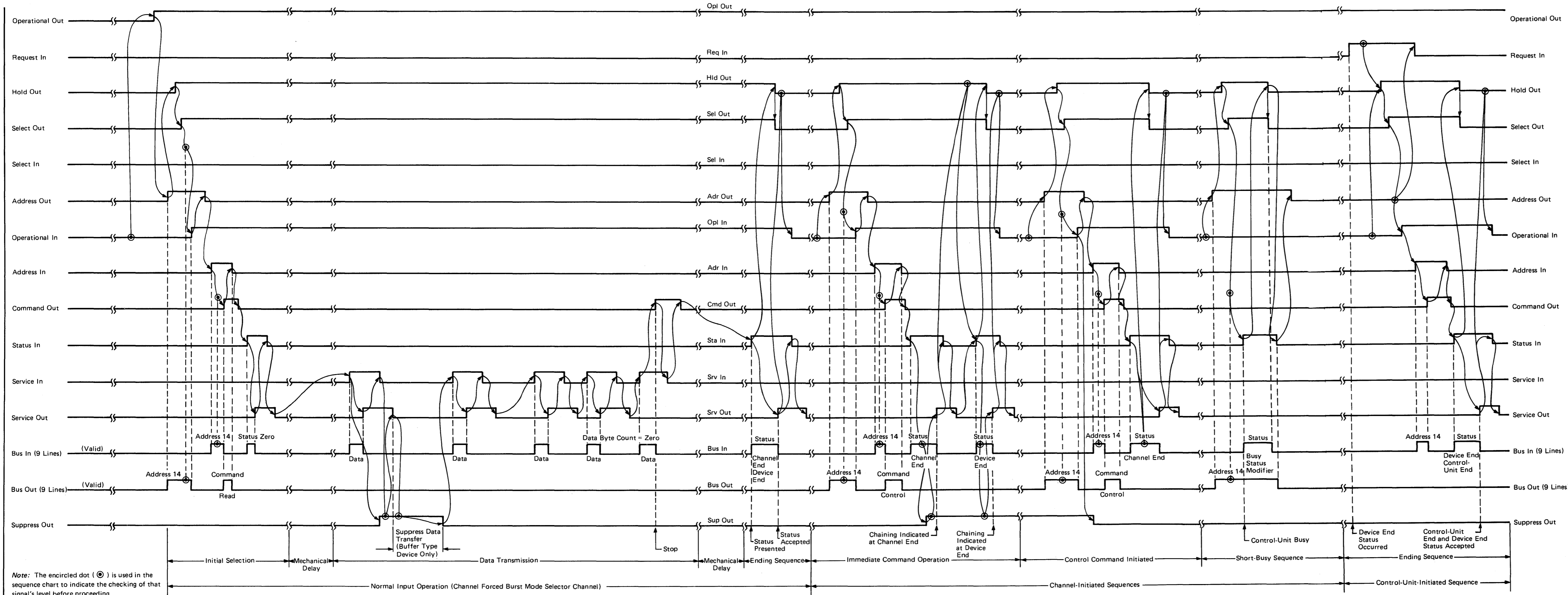
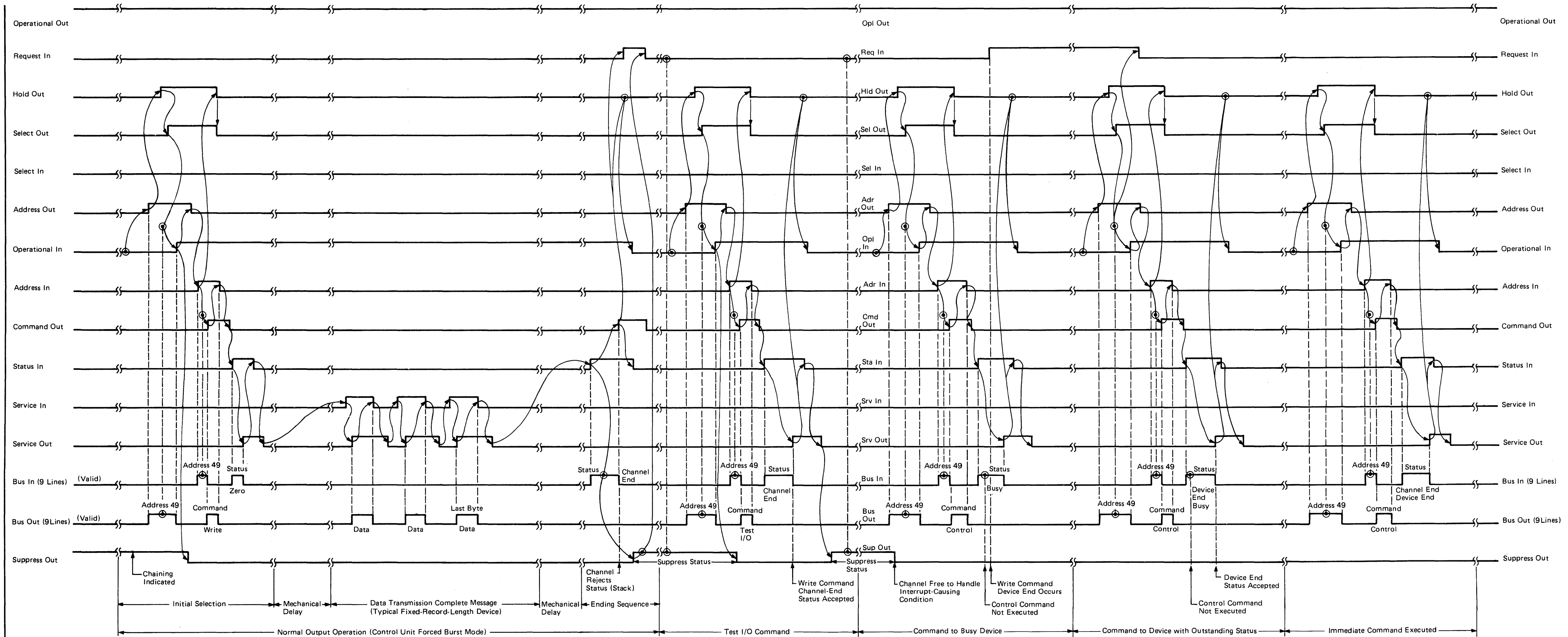


Figure C-9. Selector Channel Operations



Note: The encircled dot (⊙) is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Figure C-10. Control-Unit-Forced Burst-Mode Operation-on-Byte Multiplexer Channel  
Appendix C. Supplementary Application Information C-13

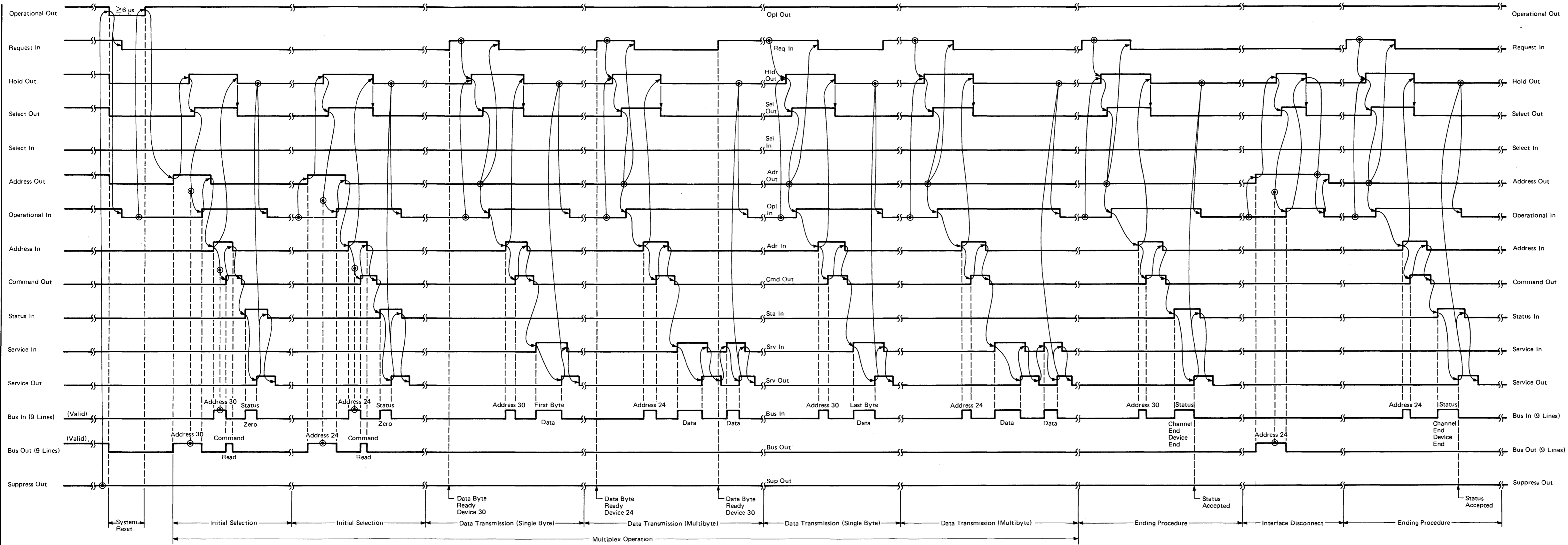


Figure C-11. Multiplexing Operation on Byte-Multiplexer Channel

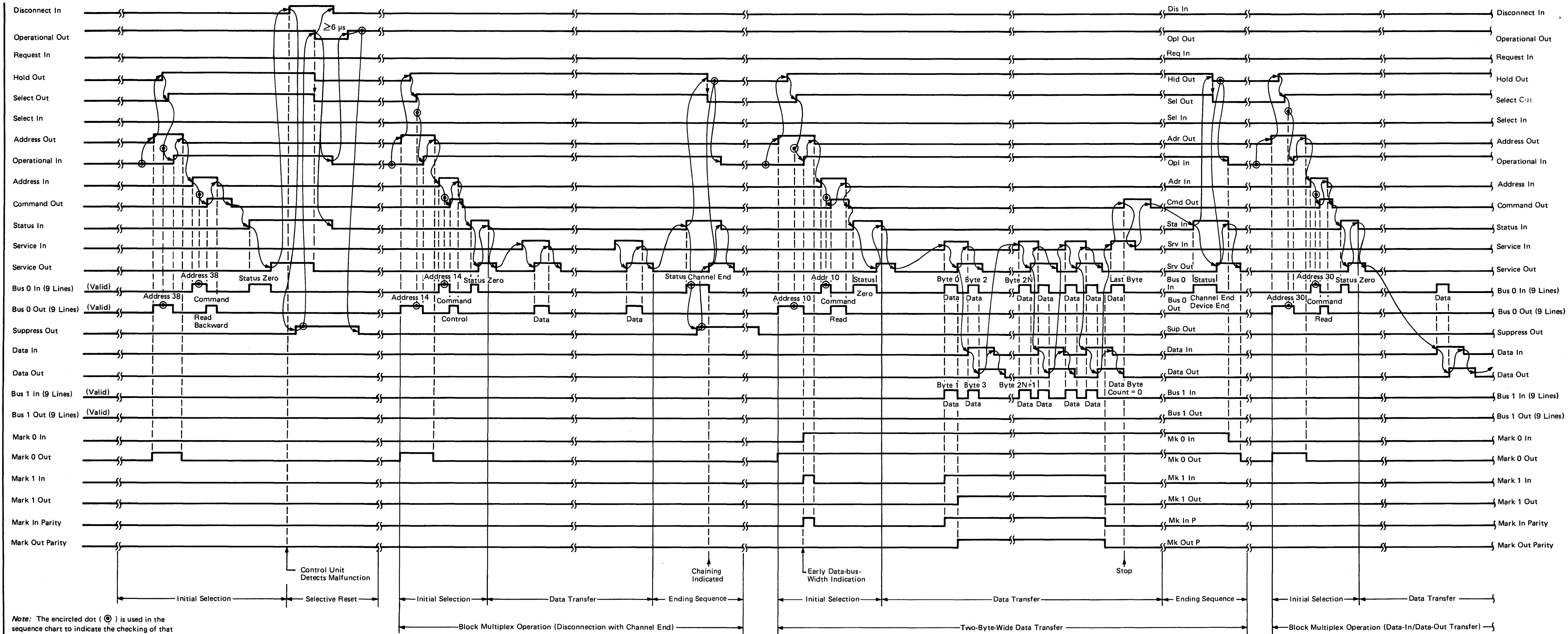
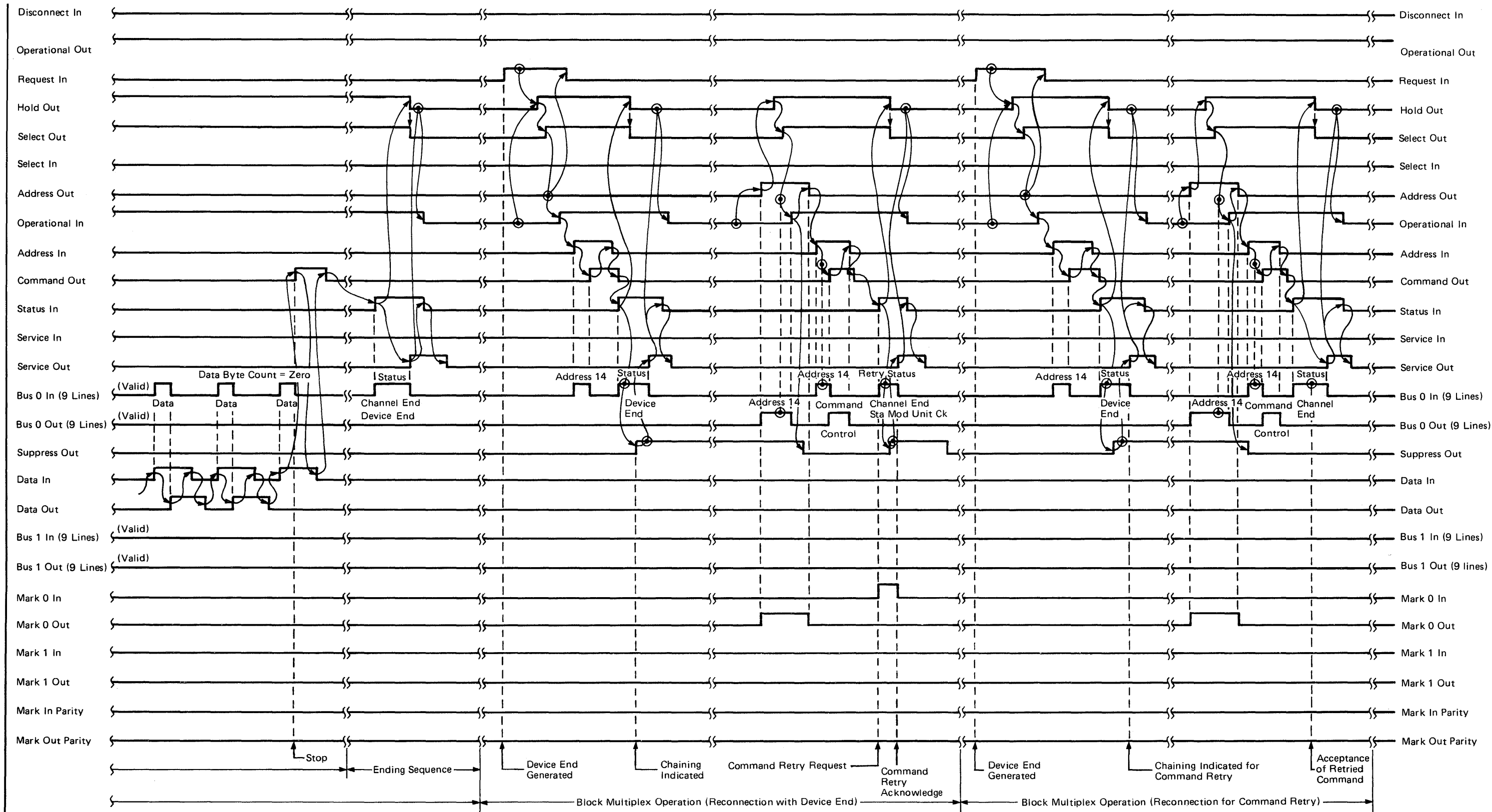
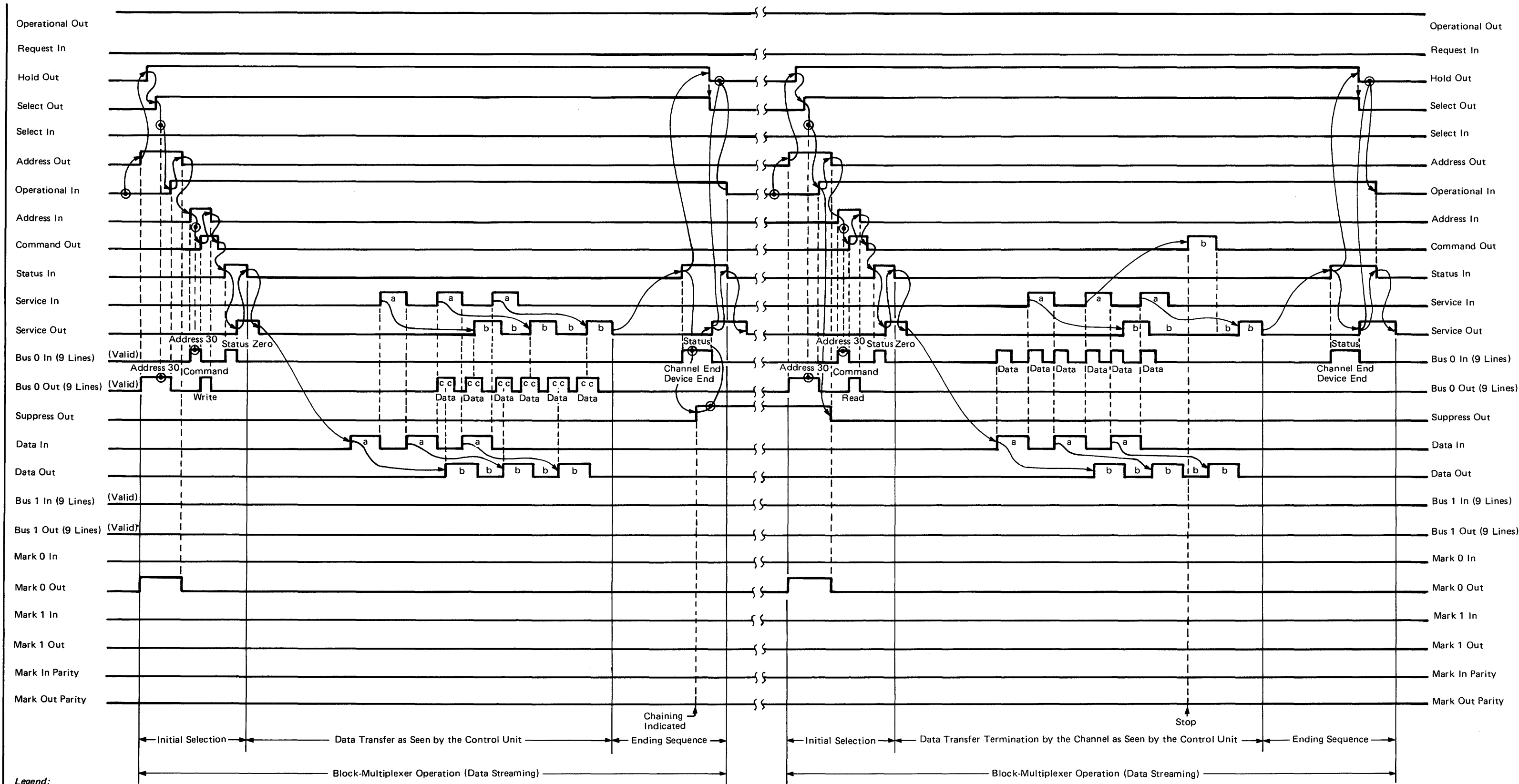


Figure C-12 (Part 1 of 2). Block-Multiplexer Channel Operation



**Note:** The encircled dot (⊙) is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Figure C-12 (Part 2 of 2). Block-Multiplexer Channel Operation



**Legend:**  
 a ≥ 270 nanoseconds  
 b ≥ 180 nanoseconds  
 c ≥ 100 nanoseconds

**Note:** The encircled dot (⊙) is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Figure C-13. Data-Streaming Operation on Block-Multiplexer Channel



Figures D-1 through D-5 summarize appropriate and inappropriate status combinations that are further described in the section "Status Combinations" in Chapter 2, "Operational Description." The condition represented by each bit position is defined in the section "Status Byte," also in Chapter 2.

First Hex Digit	Second Hex Digit															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1_	A	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5_	A	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7_	A	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
8_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
9_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
A_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
B_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
D_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
E_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
F_	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

**Explanation:**

This figure shows the 256 possible combinations of bits that may be presented as the device-status byte during the short-busy sequence. For each status combination, the figure specifies:

- Combination of bits is inappropriate for the device-status byte
- A Combination of bits is appropriate for the device-status byte

The device-status byte is represented in the figure by a two-digit hexadecimal value where the 16 possible values for each digit are shown as follows:

First hex digit: Represents bit positions 0-3 (attention, status modifier, control-unit end, and busy, respectively)

Second hex digit: Represents bit positions 4-7 (channel end, device end, unit check, and unit exception, respectively)

Figure D-1. Status Presented during Short-Busy Sequence

First Hex Digit	Second Hex Digit															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0_	A	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
1_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
2_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
3_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
4_	•	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
5_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
6_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
7_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
8_	•	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
9_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
A_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
B_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
C_	•	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
D_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
E_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
F_	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A

**Explanation:**

This figure shows the 256 possible combinations of bits that may be presented as the device-status byte for initial status when not command chaining. For each status combination, the figure specifies:

- Combination of bits is inappropriate for the device-status byte
- A Combination of bits is appropriate for the device-status byte

The device-status byte is represented in the figure by a two-digit hexadecimal value where the 16 possible values for each digit are shown as follows:

First hex digit: Represents bit positions 0-3 (attention, status modifier, control-unit end, and busy, respectively)

Second hex digit: Represents bit positions 4-7 (channel end, device end, unit check, and unit exception, respectively)

Figure D-2. Initial Status When Not Command Chaining

First Hex Digit	Second Hex Digit															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0_	A	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
1_	•	•	•	•	A	•	•	•	•	•	•	•	•	•	•	•
2_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
3_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
4_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
5_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
6_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
7_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
8_	•	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
9_	A	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
A_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
B_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
C_	•	A	A	A	•	•	•	•	A	A	A	A	A	A	A	A
D_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
E_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
F_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

**Explanation:**

This figure shows the 256 possible combinations of bits that may be presented as the device-status byte for initial status when command chaining. For each status combination, the figure specifies:

- Combination of bits is inappropriate for the device-status byte
- A Combination of bits is appropriate for the device-status byte

The device-status byte is represented in the figure by a two-digit hexadecimal value where the 16 possible values for each digit are shown as follows:

First hex digit: Represents bit positions 0-3 (attention, status modifier, control-unit end, and busy, respectively)

Second hex digit: Represents bit positions 4-7 (channel end, device end, unit check, and unit exception, respectively)

Figure D-3. Initial Status When Command Chaining

First Hex Digit	Second Hex Digit															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
1_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
2_	#	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
3_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
4_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
5_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
6_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
7_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
8_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
9_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
A_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
B_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
C_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
D_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
E_	•	•	•	•	•	•	•	•	A	A	A	A	A	A	A	A
F_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

Explanation:

This figure shows the 256 possible combinations of bits that may be presented as the device-status byte for the first status after zero initial status. For each combination, the figure specifies:

- Combination of bits is inappropriate for the device-status byte
- # Combination of bits is inappropriate for the device-status byte except when the dynamic-reconnection feature is in use
- A Combination of bits is appropriate for the device-status byte

The device-status byte is represented in the figure by a two-digit hexadecimal value where the 16 possible values for each digit are shown as follows:

First hex digit: Represents bit positions 0-3 (attention, status modifier, control-unit end, and busy, respectively)

Second hex digit: Represents bit positions 4-7 (channel end, device end, unit check, and unit exception, respectively)

Figure D-4. First Status after Zero Initial Status

First Hex Digit	Second Hex Digit															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0_	•	•	•	•	A	A	A	A	•	•	•	•	•	•	•	•
1_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
2_	#	•	A	A	A	A	A	A	•	•	•	•	•	•	•	•
3_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
4_	•	•	•	•	A	A	A	A	•	•	•	•	•	•	•	•
5_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
6_	•	•	A	A	A	A	A	A	•	•	•	•	•	•	•	•
7_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
8_	•	•	•	•	A	A	A	A	•	•	•	•	•	•	•	•
9_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
A_	•	•	A	A	A	A	A	A	•	•	•	•	•	•	•	•
B_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
C_	•	•	•	•	A	A	A	A	•	•	•	•	•	•	•	•
D_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
E_	•	•	A	A	A	A	A	A	•	•	•	•	•	•	•	•
F_	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

**Explanation:**

This figure shows the 256 possible combinations of bits that may be presented as the device-status byte for the first status after channel end without device end. For each status combination, the figure specifies:

- Combination of bits is inappropriate for the device-status byte
- # Combination of bits is inappropriate for the device-status byte except when the dynamic-reconnection feature is in use or when command chaining was not indicated
- A Combination of bits is appropriate for the device-status byte

The device-status byte is represented in the figure by a two-digit hexadecimal value where the 16 possible values for each digit are shown as follows:

First hex digit: Represents bit positions 0-3 (attention, status modifier, control-unit end, and busy, respectively)

Second hex digit: Represents bit positions 4-7 (channel end, device end, unit check, and unit exception, respectively)

Figure D-5. First Status after Channel End without Device End

- A**  
 accept-data indication 2-9  
 accept-data sequence control 2-9  
 accept status  
   indication 2-10  
   sequence control 2-10  
 accepted, command 2-1  
 address  
   assignment 2-4  
   decoding 2-5  
   legitimate 2-15  
   nonsequential 2-5  
 address in 1-9  
 address out 1-7  
 addressing 2-4  
 appropriate status 2-18  
 assignment, address 2-4  
 asynchronous conditions 2-13  
 attention status bit 2-13
- B**  
 basic operations 2-6  
 basic read 2-6  
 basic sense 2-7  
 blocks, connector (See connector)  
 burst mode 2-3  
 bus-extension feature 3-1  
   early data-bus-width indication 3-2  
   mark-in lines 3-1  
   mark-out lines 3-1  
 bus in 1-6  
   data streaming, valid data 3-4  
   signal skew 1-7  
   valid data 1-6  
 bus out 1-6  
   data streaming, valid data 3-4  
   signal skew 1-6  
   valid data 1-6  
 bus-out-check sense bit 2-20  
 buses  
   general 1-5  
   information 1-5  
   line parity 1-5  
   unused lines 1-5  
 busy  
   control-unit 2-14  
   I/O-device 2-16  
   short sequence 2-1  
   status bit 2-15  
   temporary control-unit 2-15  
 busy status (See busy-status bit and status-modifier bit)  
 byte  
   command 2-6  
   status 2-12  
 byte-multiplex mode 2-3
- C**  
 cable  
   characteristics A-1  
   connectors, mounting B-1  
   halves B-1  
 cabling B-1  
   external 2-22  
   internal 2-22  
 cabling and connectors, signal B-1  
 cancellation of I/O-device selection 2-11  
 channel-end status bit 2-16  
 check, unit 2-17  
 circuit requirements, I/O interface A-2  
 circuits A-3  
   line driver A-3  
   line receiver A-3  
   line terminator A-3  
 clock out 1-11  
 codes, modifier 2-6  
 command accepted 2-1  
 command byte 2-6  
   bit definitions 2-6  
 command-chaining sequence control 2-10  
 command out 1-9  
 command-reject sense bit 2-20  
 command-retry feature 3-2  
 command-retry sequence 3-2  
 commands 2-6  
   control 2-7  
   read 2-6  
   read backward 2-7  
   sense 2-7  
   test 2-8  
   write 2-7  
 commands, special diagnostic 2-6  
 conditions, unit-status 2-13  
 configuration, system 2-22  
 connection  
   burst-mode 2-2  
   byte-multiplex-mode 2-2  
   I/O-interface 2-2  
 connector B-3  
   capacity B-3  
   electrical specifications B-5  
   I/O-interface, pin assignments A-5  
   serpentine contacts B-3  
   wire termination B-4  
 contacts, serpentine B-3  
 control command 2-7  
   no-operation 2-7  
 control unit  
   model number 2-8  
   selection 1-8  
   type number 2-8  
 control-unit-end status bit 2-14  
 control-unit-initiated sequence 2-2  
 control-unit type 2-4  
   type-1 2-4  
   type-2 2-4  
   type-3 2-4  
 controls and tag lines, selection 1-7
- D**  
 data  
   bus definition 1-6  
   in tag line 3-3  
   out tag line 3-3  
   ready indication 2-9  
   transfer  
     and channel end 2-16  
     and ready indication (data ready) 2-9

- data-check sense bit 2-20
- data-ready sequence control 2-9
- data-streaming feature 3-3
  - data transfer 3-4
    - interface disconnect 3-6
    - interface-line overlap 3-4
    - response-time requirements 3-6
    - selective reset 3-6
    - stop/command out 3-5
    - suppress data 3-6
    - timeout considerations 3-6
  - valid data
    - bus in 3-4
    - bus out 3-4
- data transfer 2-2
  - data streaming 3-4
- data-transfer sequence 2-2
- decoding, address 2-5
- device (See I/O device)
- device-end status bit 2-16
- diagnostic commands, special 2-6
- disconnect, interface 2-11
- disconnect in 3-2
- downtime A-4
- driver, line
  - normal A-2
  - select-out A-4
- drivers and receivers, multiple A-1
- dynamic-reconnection feature 3-7

## E

- early data-bus-width indication 3-2
- electrical
  - connector, serpentine-contact B-3
  - requirements, general A-1
  - specifications A-1
    - electrical requirements A-1
    - physical requirements A-1
  - select out A-3
- end
  - channel 2-16
  - control-unit 2-14
  - device 2-16
- ending sequence 2-3
- ending status (ending sequence) 2-3
- equipment-check sense bit 2-20
- exception, unit 2-18
- external cabling 2-22
- external line resistance A-1

## F

- fault conditions A-3
- features 3-1
  - bus extension 3-1
  - command retry 3-2
  - data streaming 3-3
  - dynamic-reconnection 3-7
  - high-speed transfer 3-2
  - I/O-error alert 3-2
- functional description 1-1

## G

- general electrical requirements A-1
- general system considerations 2-21
- ground, circuit and machine A-3

- ground shift and noise A-1

## H

- high-speed-transfer feature 3-2
  - data in 3-3
  - data out 3-3
- hold out 1-8

## I

- I/O device 1-1
  - address 2-4
  - model number 2-8
  - selection 1-7
  - type number 2-8
- I/O-error-alert feature 3-2
- I/O interface
  - circuit requirements A-2
  - connector, pin assignments A-5
  - lines (See I/O-interface lines)
  - sequences (See sequences)
  - timeout considerations 2-21
- I/O-interface connection 2-2
- I/O-interface line, overlap, data streaming 3-4
- I/O-interface lines 1-3
  - address in 1-9
  - address out 1-7
  - bus in 1-6
  - bus out 1-6
  - clock out 1-11
  - command out 1-9
  - data in 3-3
  - data out 3-3
  - disconnect in 3-2
  - hold out 1-8
  - mark in 3-1
  - mark out 3-1
  - metering in 1-11
  - metering out 1-11
  - operational in 1-9
  - operational out 1-7
  - request in 1-7
  - reserved 1-11
  - select in 1-8
  - select out 1-8
  - service in 1-10
  - service out 1-10
  - status in 1-10
  - suppress out 1-11
- immediate, operation 2-6
- inappropriate status 2-18
- initial-selection sequence 2-1
- initial status 2-1
- interface-disconnect sequence control 2-11
- interface disconnect while data streaming 3-6
- interlock summary, signal 1-11
- interlocking rules 1-11
- internal
  - cabling 2-22
  - delay 2-22
  - resistance 2-22
  - skew 2-22
- intervention-required sense bit 2-20
- IPL, basic read 2-6

J  
jump status (status modifier) 2-14

## L

line  
  driver  
    normal A-2  
    select-out A-4  
  receiver  
    normal A-2  
    select-out A-4  
  resistance  
    external A-1  
    internal 2-22  
  termination A-1  
  terminator block B-6  
  total resistance 2-22  
line definition 1-3  
lines (See I/O-interface lines)  
  reserved 1-11  
logic levels (See voltage levels)  
logical zero and logical one A-1

## M

mark in 3-1  
mark out 3-1  
mark 0 in (See mark in and command  
  retry)  
maximum  
  internal line resistance 2-22  
  internal skew 2-22  
  skew, data streaming 3-3  
  total line resistance 2-22  
measuring I/O-interface timings A-4  
metering controls 1-11  
  clock out 1-11  
  metering in 1-11  
  metering out 1-11  
metering in 1-11  
metering out 1-11  
mode  
  burst 2-3  
  byte-multiplex 2-3  
modifier  
  bits 2-6  
  codes 2-6  
  status 2-14  
mounting, cable connector B-1  
multiple drivers and receivers A-1  
multiplex mode, byte 2-3

## N

no-operation control command 2-7  
noise, ground shift and A-1  
not ready 2-16  
number of units, system configuration  
  2-22

## O

offline utilization B-1  
online/offline  
  cabling B-1  
  mode 2-22  
  transition 2-22  
operation, immediate 2-6  
operational description

  data transfer 2-2  
  ending 2-3  
  selection 2-1  
operational in 1-9  
operational out 1-7  
operations, basic 2-6  
outbound tag lines (See tags)  
overlap not greater than A-5  
overlapping sequence 1-9  
overrun 1-10  
  sense bit 2-21

## P

parity on bus lines 1-5  
pending status 2-13  
physical requirements A-1  
pin assignments, interface connector  
  A-5  
power effect 2-23  
power off/on sequence requirements 2-24  
priority selection 1-8  
proceed sequence control 2-8  
propagation of select out 2-21

## R

read-backward command 2-7  
read command 2-6  
receiver, line  
  normal A-2  
  select-out A-4  
receivers and drivers, multiple A-1  
request in 1-7  
requirements  
  electrical A-1  
  physical A-1  
reserved lines 1-11  
reset  
  selective 2-12  
  system 2-12  
resistance  
  maximum internal line 2-22  
  maximum total line 2-22  
retry, command 3-2

## S

select in 1-8  
select out 1-8  
  driver A-4  
  electrical specifications A-3  
  propagation of 2-21  
  pulse splitting 2-23  
  receiver A-4  
  terminator A-4  
selection  
  I/O-device 1-7  
  priority 1-8  
selection controls 1-7  
  disconnect in 3-2  
  hold out 1-8  
  operational in 1-9  
  operational out 1-7  
  request in 1-7  
  select in 1-8  
  select out 1-8  
  suppress out 1-11  
selective-reset sequence control 2-12  
selective reset while data streaming  
  3-6



sense 2-19  
   bits  
     but-out check 2-20  
     command reject 2-20  
     data check 2-20  
     equipment check 2-20  
     intervention required 2-20  
     overrun 2-21  
   byte 2-20  
   command 2-7  
     basic sense 2-7  
     sense-ID 2-7  
   conditions 2-20  
 sense ID 2-7  
 sequence  
   command-retry 3-2  
   control-unit-initiated 2-2  
   controls 2-8  
   data-transfer 2-2  
   ending 2-3  
   initial-selection 2-1  
   overlapping 1-9  
   short-busy 2-1  
 sequence controls 2-8  
   accept data 2-9  
   accept status 2-10  
   command chaining 2-10  
   data ready 2-9  
   interface disconnect 2-11  
   proceed 2-8  
   selective reset 2-12  
   stack status 2-9  
   stop 2-8  
   suppress data 2-9  
   suppress status 2-10  
   system reset 2-12  
 sequence requirements, power off/on 2-24  
 serpentine contacts B-3  
 service in 1-10  
 service out 1-10  
 short-busy sequence 2-1  
 short-term busy (See temporary control-unit busy)  
 signal  
   cabling and connectors B-1  
   interlock summary 1-11  
   line definition 1-3  
   lines (See I/O-interface lines)  
   measuring timings A-4  
   spurious 2-23  
   transition time 1-12  
   voltage level  
     driver A-2  
     receiver A-2  
     select-out driver A-4  
     select-out receiver A-4  
 skew  
   bus-in 1-7  
   bus-out 1-6  
   maximum, data streaming 3-3  
   maximum internal 2-22  
 special diagnostic commands 2-6  
 specifications, electrical A-1  
 spurious signals 2-23  
 stack-status sequence control 2-9  
 stacked status 2-13  
 status  
   accept 2-10  
   appropriate 2-18  
   bits  
     attention 2-13  
     busy 2-15  
     channel end 2-16  
     control-unit end 2-14  
     device end 2-16  
     jump (See status modifier)  
     status modifier 2-14  
     temporary control-unit busy (See control-unit end)  
     unit check 2-17  
     unit exception 2-18  
   byte 2-13  
   conditions 2-13  
   ending 2-3  
   inappropriate 2-18  
   information 2-12  
   initial 2-1  
   modifier 2-14  
   pending 2-13  
   stack 2-9  
   stacked 2-13  
   suppress 2-10  
   status combinations 2-18  
   status in 1-10  
   status-modifier status bit 2-14  
   stop sequence control 2-8  
   stop/command out, data streaming 3-5  
   suppress-data sequence control 2-9  
   suppress data while data streaming 3-6  
   suppress out 1-11  
   suppress-status sequence control 2-10  
   system  
     configuration 2-22  
     reset sequence control 2-12  
   system configuration, number of units 2-22  
  
 T  
 tag lines 1-7  
   address in 1-9  
   address out 1-7  
   command out 1-9  
   data in 3-3  
   data out 3-3  
   service in 1-10  
   service out 1-10  
   status in 1-10  
 temporary control-unit busy 2-15  
 termination  
   line A-1  
   wire B-4  
 terminator  
   block B-6  
   networks A-1  
   select out A-4  
 test command, test I/O 2-8  
 test I/O, test command 2-8  
 timeout, I/O-interface considerations 2-21  
 timeout considerations, data streaming 3-6  
 transient  
   and select-out pulse splitting 2-23  
   and spurious signals 2-23  
   transition time, signal 1-12  
   type-1, control-unit type 2-4  
   type-2, control-unit type 2-4  
   type-3, control-unit type 2-4  
  
 U  
 unit-check status bit 2-17  
 unit-exception status bit 2-18  
 unit-status conditions 2-13

unused bus lines 1-5  
uptime A-4

V

valid data A-5  
  bus in 1-6  
  bus out 1-6  
  data streaming  
    bus in 3-4  
    but out 3-4

voltage level  
  driver A-2  
  receiver A-2  
  select-out driver A-4  
  select-out receiver A-4

W

wire termination B-4  
write command 2-7

GA22-6974-08



GA22-6974-08

