

IBM Systems Reference Library

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

This Original Equipment Manufacturers' Information (OEMI) manual provides the definitions and functional descriptions of the interface lines for the IBM I/O Interface – Channel to Control Unit. In addition, it contains electrical, mechanical, and cabling considerations and specifications of this interface.



Preface

The I/O Interface (Channel to Control Unit) is the communication link between the CPU channel and the various I/O control units in the IBM System/360. It employs information formats and control signal sequencing to provide uniform means for attaching and controlling various types of control units.

Information, in the form of data, status and sense information, control signals and I/O device addresses, is transmitted over the 34 time- and function-shared lines of this interface. All transmissions of information are interlocked with corresponding response signals.

The design of this interface provides these important features:

- A degree of consistency of input/output programming over a wide range of control units.
- Ready physical connection to System/360 channels of control units designed by any manufacturer to operate with this interface.
- Ability to physically accommodate future control units designed to meet the parameters of this interface.

An interlocked interface operation that is not very time-dependent; this feature increases the range of control units that may be attached.

An operation applicable to both multiplex and burst mode operations as well as many control operations and channel-to-channel transmissions.

Up to eight control units serviced per set of lines.

This Original Equipment Manufacturers' Information (OEM I) Manual provides a functional description of these lines together with the electrical, mechanical and cabling considerations, and specifications of the interface. It does not define the interface between control units and I/O devices.

This publication is open-ended to permit future extensions to the I/O interface. The document also provides additional information of interest to designers and manufacturers of equipment to be attached to the IBM System/360.

The I/O interface, or any aspect of the I/O interface, may be altered from time to time by IBM or may be withdrawn by IBM in part or in whole.

Fourth Edition

This edition, Form A22-6843-3, obsoletes Form A22-6843-2. Significant changes have been made throughout the manual, and this new edition should be reviewed in its entirety.

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

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by the IBM Systems Development Division, Product Publications, Dept. B98, P.O. Box 390, Poughkeepsie, N.Y. 12602. Address comments concerning the manual to this address.

Contents

Functional Description	5	Unit Exception	23
Line Definition	7	Sense Information	23
Input/Output Interface Lines	7	Sense Byte	24
Buses — General	7	Sense Conditions	24
Bus Out	7	Command Reject	24
Bus In	8	Intervention Required	24
Selection Controls and Tag Lines	8	Bus-Out Check	24
Operational Out	8	Equipment Check	24
Request In	8	Data Check	24
Address Out	9	Overrun	24
Select Out/Hold Out and Select In	9	General System Considerations	25
Operational In	10	Interface Time-Out Considerations	25
Address In	10	Propagation of Select Out	25
Command Out	10	System Configuration	25
Status In	11	Off-Line/On-Line	26
Service Out	11	Power Effects	26
Service In	11	Appendix A. Electrical Specifications	A-1
Suppress Out	11	Physical Requirements	A-1
Clock Out	11	Multiple Drivers and Receivers	A-1
Metering In	11	General Electrical Requirements	A-1
Metering Out	12	Voltage Levels	A-1
Reserved Lines	12	Cable	A-1
Signal Interlock Summary	12	Terminating Networks	A-1
Operational Description	13	Ground Shift and Noise	A-1
Interface Sequences	13	Interface Circuit Requirements	A-2
Initial Selection Sequence	13	Receivers	A-2
Control Unit Busy Sequence	13	Drivers	A-2
Control-Unit-Initiated Sequence	14	Fault Conditions	A-2
Data Transfer	14	Electrical Specifications for Select-out Circuitry	A-3
Multiplex Mode	14	General	A-3
Burst Mode	14	Receiver	A-3
Ending Procedure and Asynchronous Status	14	Driver	A-3
Addressing	15	Terminator	A-3
Address Assignment	15	Circuits	A-3
Address Decoding	15	Pin Assignments	A-4
Commands	16	Appendix B. Signal Cabling and Connectors	B-1
Command Byte	16	Cabling	B-1
Basic Operations	16	Cable Halves	B-1
Read	16	Mounting	B-1
Read Backward	16	Off-Line Utilization	B-1
Write	17	Connectors	B-1
Control	17	Connector Blocks	B-1
Sense	17	Capacity	B-2
Test I/O	17	Serpent Contacts	B-3
Sequence Controls	17	Description	B-3
Proceed	17	Wire Termination	B-3
Stop	17	Terminators	B-3
Stack Status	18	Electrical Specifications	B-3
Suppress Data	18	Appendix C. Supplementary Application	C-1
Accept Data	18	Information	C-1
Data Ready	18	Flow Diagram 1. Initiation of Polling or Selection	C-1
Suppress Status	18	Flow Diagram 2. Control Unit Response to Select Out	C-2
Accept Status	18	(Part 1 of 2)	C-2
Command Chaining	18	Flow Diagram 2. Control Unit Response to Select Out	C-3
Interface Disconnect	19	(Part 2 of 3)	C-3
Selective Reset	19	Flow Diagram 3. Command Transfer	C-4
System Reset	20	Flow Diagram 4. Status/Data Presentation	C-5
Status Information	20	Flow Diagram 5. Response to Status/Data Presentation	C-6
Status Byte	20	Flow Diagram 6. Response to Stack/Stop/Accept	C-7
Unit Status Conditions	20	Flow Diagram 7. Response to Fall of Status/Service In	C-8
Attention	21	Flow Diagram 8. Burst Mode Waiting Loop	C-9
Status Modifier	21	Sequence Chart 1. Selector Channel Operation	C-11
Control Unit End	21	Sequence Chart 2. Control Unit Forced Burst Mode-	C-13
Busy	21	Multiplexer Channel	C-13
Channel End	22	Sequence Chart 3. Multiplexing Operation on Multi-	C-15
Device End	22	plexer Channel	C-15
Unit Check	22		

Functional Description

Input/output devices provide external storage and a means of communication between data processing systems or between a system and the external world.

Input/output (I/O) devices are attached to the central processing unit (CPU) by means of channels and control units in various configurations (Figure 1).

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

The control unit provides the logical capability necessary to operate and control an I/O device, and adapts the characteristics of each I/O device to the standard form of control provided by the channel. A

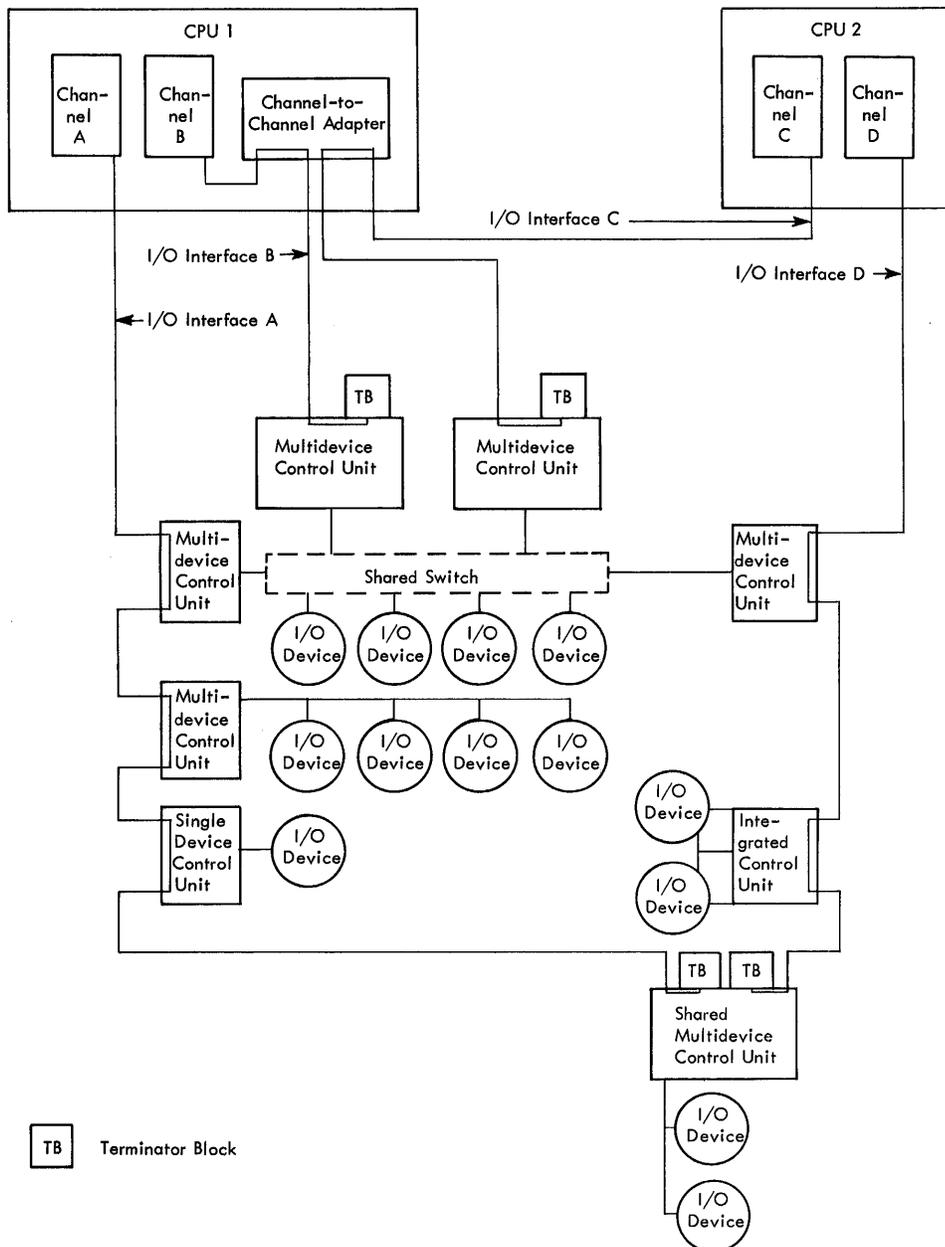


Figure 1. Input/Output Interface - Multiple Configurations

control unit may be housed separately or it may be physically and logically integral with the I/O device.

An I/O device attached to the control unit may be designated to perform only certain limited operations; for example recording data and moving the recording medium. To accomplish these functions the I/O device needs detailed signal sequences peculiar to that type of I/O device. The control unit decodes the commands received from the channel, interprets them for the I/O device, and provides the signal sequence for executing the operation.

The connection between the channel and the control unit is called the I/O interface. The I/O interface provides an information format and a signal sequence common to all control units. The interface consists of signal lines that connect a number of control units to a channel (Figure 1.) Except for signals used to establish selection control, all communications to and from the channel occur over a common bus; i.e., any signal provided by the channel is available to all control units (Figure 2). At any one time, however, only

one control unit can be logically connected to the channel. Selection of a control unit for communication with the channel is controlled by a signal (passing serially through all control units) that permits, sequentially, each control unit to respond to the signals provided by the channel. A control unit remains logically connected to the interface until it transfers the information it needs or has, or until the channel signals it to disconnect.

The interface can accommodate up to 256 directly addressable I/O devices (limit set by addressing facilities). The number of control units that can be accommodated is limited only by timing and electrical considerations within the given addressing limitations.

The multiplexing facilities of the interface permit the possibility of operating any number of the 256 I/O devices concurrently on a single interface; portions of various messages can be transmitted over the interface in an interleaved fashion to or from different I/O devices, or a complete message can be transmitted in

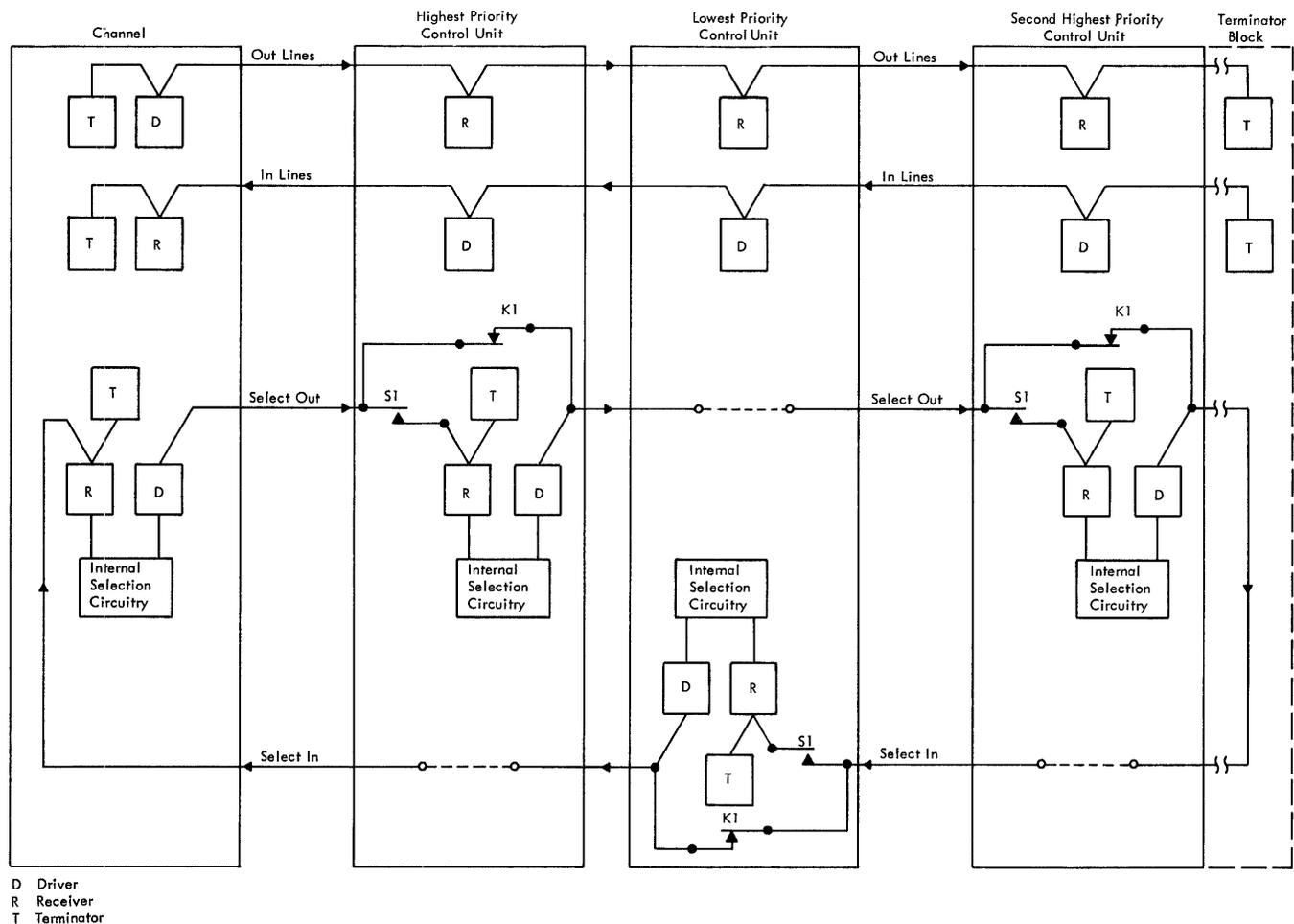


Figure 2. Interconnections on the Input/Output Interface

a single interface operation. The operation is determined by the particular channel and control unit.

The rise and fall of all signals transmitted over the interface are controlled by interlocked responses. Interlocking removes the dependence of the interface on circuit speed, and makes it applicable to a wide variety of circuits and data rates. Further, interlocking permits connecting control units of different circuit speeds to a single channel.

Line Definition

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

Input/Output Interface Lines

The I/O interface lines and their uses are:

NAME OF LINE	ABBREVIATION	USES
Bus Out Position P	Bus Out P	<i>Bus Out</i> — Used to transmit information (data, I/O device address, commands, control orders) from the channel to the control unit.
Bus Out Position 0	Bus Out 0	
Bus Out Position 1	Bus Out 1	
Bus Out Position 2	Bus Out 2	
Bus Out Position 3	Bus Out 3	
Bus Out Position 4	Bus Out 4	
Bus Out Position 5	Bus Out 5	
Bus Out Position 6	Bus Out 6	
Bus Out Position 7	Bus Out 7	
Bus In Position P	Bus In P	<i>Bus In</i> — Used to transmit information (data, selected I/O device identification, status information, sense data) from the control unit to the channel.
Bus In Position 0	Bus In 0	
Bus In Position 1	Bus In 1	
Bus In Position 2	Bus In 2	
Bus In Position 3	Bus In 3	
Bus In Position 4	Bus In 4	
Bus In Position 5	Bus In 5	
Bus In Position 6	Bus In 6	
Bus In Position 7	Bus In 7	
Address Out	Adr Out	<i>Tags</i> — Used for interlocking and controlling information on the buses, and for special sequences.
Address In	Adr In	
Command Out	Cmd Out	
Status In	Sta In	
Service Out	Srv Out	
Service In	Srv In	
Operation Out	Opl Out	<i>Selection Controls</i> — Used for the scanning of, or the selection of, attached I/O devices.
Operational In	Opl In	
Hold Out	Hld Out	
Select Out	Sel Out	
Select In	Sel In	
Suppress Out	Sup Out	
Request In	Req In	
Metering Out	Mtr Out	<i>Metering Controls</i> — Used for the conditioning of usage meters located in the various attached units.
Metering In	Mtr In	
Clock Out	Clk Out	

NOTE: The validity of information on the buses and the timing of the signals on the tag lines is specified 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 low-order bit within an eight-bit byte. The highest-order bit is in position 0 and intervening bits are in descending order from position 1 to position 6.

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

Bus Out

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

1. When 'address out' is up during the channel-initiated selection sequence, 'bus out' specifies the address of the I/O device with which the channel wants to communicate.

2. When 'command out' is up during the channel-initiated selection sequence, 'bus out' specifies a command.

3. When 'service out' is up in response to 'service in' during the execution of a write or control command, the nature of the information on 'bus out' depends on the type of operation. For example, during a write operation it contains data to be recorded by the I/O device. During a control operation it can specify an order code or a second-level address within the control unit or I/O device.

The period during which information on 'bus out' is valid is controlled by the tag lines. During transmission of the I/O device address, information on the bus need be valid from the rise of 'address out' until the rise of 'operational in,' 'select in,' or, in the case of the control-unit-busy sequence, until 'status in' drops. 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' must be accommodated by the channel. (Except as noted in the section "Address Out," the channel must delay raising of the signal on the outbound tag lines by a time duration which ensures that the information on 'bus out' will precede

Table 1. Organization of Information on Bus

BUS LINE	BCD (POSITION VALUE)	PACKED NUMERIC (POSITION VALUE)	UNPACKED NUMERIC (POSITION VALUE)	EBCDIC (BIT POSITIONS)	USASCII-8 (BIT POSITIONS)	BINARY (POSITION VALUE)
<i>P</i>	<i>P</i>	<i>P</i>	<i>P</i>	<i>P</i>	<i>P</i>	<i>P</i>
0	0	8	0	0	7	128
1	0	4	} Digit } X	1	6	64
2	B	2		0	2	X
3	A	1	} Digit } X + 1	3	5	16
4	8	8		8	4	4
5	4	4	4	5	3	4
6	2	2	2	6	2	2
7	1	1	1	7	1	1

X = Higher order digit
X + 1 = Lower order digit
0 = Logical Zero

the signal on the outbound tag line by not less than 100 nanoseconds (ns) when measured at the cable connectors at the channel location under the worst-case skew conditions.) The channel thus must provide a delay that accommodates skew caused by its own circuitry and, in addition, must provide a delay of at least 100 ns. This delay compensates for skew caused by the cable, and for most control units, will also be sufficient to accommodate the skew caused by the interface receivers. When a control unit can cause more skew, the control unit must provide the additional delay to compensate for it.

Bus In

'Bus in' is used to transmit addresses, status, sense information, 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 case of the control-unit-busy sequence.

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' is up during execution of a read 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 sense operation, the bus contains a set of bits describing the detailed status of the I/O device and the conditions under which the last operation was terminated.

The period during which information on 'bus in' is valid is controlled by the tag lines. Information on the bus must be valid within 100 ns after the rise of the associated inbound tag and must remain valid until the rise of the responding outbound tag, or in a control-unit-busy sequence, until 'select out' drops. The 100-

ns 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 must provide a delay in the inbound tag lines to accommodate skew caused by the channel circuitry (including its receivers) and, in addition, must provide a delay of at least 100 ns. This delay compensates for skew caused by the cable, and, for most control units, 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-ns delay, the control unit must provide 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 must drop and any operation currently in process over the interface must be reset. Under these conditions, all control-unit generated interface signals must be down within 1.5 microseconds (μsec) after the fall of 'operational out' at the control unit. See "Selective Reset" and "System Reset."

Request In

'Request in' is a line from all attached I/O control units to the channel and indicates that the control unit is ready to present status information or data and thus is requesting a selection sequence. See "Control Unit Initiated Sequence." 'Request in' should be dropped when 'operational in' rises, unless additional selection

sequences are required, or when the control unit is no longer ready to present the status information or data, or when the selection requirement is satisfied by another path. 'Request in' must in no case fall later than 250 ns after the fall of 'operational in' if the sequence satisfies the service requirements of the control unit.

'Request in' must not remain up when 'suppress out' is up if the request for status presentation is suppressible (see "Suppress Status"). Under this condition 'request in' must fall at the control unit within 1.5 μ sec 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. It provides two functions:

1. *I/O Device Selection.* 'Address out' is used to signal all the control units to decode the I/O device address on 'bus out.' The control unit, when recognizing the address, must respond by raising its 'operational in' line when its incoming 'select out' (see "Select Out/Hold Out and Select In") rises with 'address out' still up (except in the case of control unit busy). The rise of 'address out' precedes the rise of 'select out' by a minimum of 400 ns and follows the placing of the I/O device address on 'bus out' by at least 250 ns. If 'address out' falls before 'select out' rises, this control-unit selection is canceled.

'Address out' can rise only when 'select out' (hold out), 'select in,' and 'operational in' are down at the channel (except as otherwise noted in this paragraph). Ultimate use of the address on 'bus out' at the control unit is timed by the next rise of 'select out' at the addressed control unit. The rise of 'address out' must be delayed at least 250 ns after the address is placed on 'bus out.' Once 'address out' and 'select out' (hold out) are up, 'address out' must stay up until either 'select in' or 'operational in' rises, or until 'status in' falls, in the case of the control-unit-busy sequence. During I/O device selection, 'address out' cannot be up concurrently with any other outbound tag line.

2. *Disconnect Operation.* If 'hold out' is down and 'address out' rises or 'address out' is up and 'hold out' falls, the presently connected control unit must drop its 'operational in,' thus disconnecting from the interface. 'Address out' remains up until 'operational in' drops. 'Operational in' must drop within 6 μ sec after receiving the disconnect indication. Mechanical motion in process continues to a normal stopping point. Status information will be generated and presented to the channel when appropriate. See "Interface Disconnect." 'Address out,' in this case, may be up concurrently with another outbound tag line.

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), 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.' A selection priority is established because the rise of 'select out' is effective only to the first control unit on the line, and if selection is not required, is in turn propagated by each control unit to the next control unit on the line. This priority is in a descending sequence from the channel through each control unit with selection circuitry attached to 'select out' followed, in descending order back to the channel, by the remaining control units with selection circuitry attached to 'select in.' (See Figure 2.)

Note: For purposes of clarity, this document assumes that the selection circuitry of all control units is attached to 'select out.'

Each control unit must ensure that the process of electrically by-passing 'select out' before power changes does not interfere with the propagation of 'select out.' Each control unit must therefore ensure that 'select out' discontinuities, which may occur when another control unit on the interface is powered up or down, do not affect the propagation of 'select out.' This 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 3.)

Throughout the following description, 'select out' assumes the proper operation of the latch; i.e., the rise of 'select out' at the control unit presumes that 'hold out' is up and the fall of 'select out' is a result of the fall of 'hold out.'

Select Out is a line from the channel to the control unit that has highest priority, and from any control unit to the control unit next lower in priority. This line, together with 'select in,' provides a loop for scanning the attached control units. A control unit can raise its 'operational in' only at the rise of its incoming 'select out.' If a control unit does not require selection, it must propagate the signal to the next control unit. Once a control unit propagates 'select out,' it cannot raise its 'operational in' or respond with a control-unit-busy sequence until the next rise of the incoming 'select out.'

When an operation is being initiated by the channel,

the rise of 'address out' must precede the rise of 'select out' by a minimum of 400 ns.

The channel must keep 'select out' up until either 'select in,' or 'address in' and 'operational in,' or 'status in' rises. When 'select in' rises, 'select out' must drop and may not rise again until after 'select in' falls. A control unit becomes selected only when it raises its 'operational in.' 'Select out' must drop so that 'operational in' may drop. However, after the drop of 'select out,' the control unit must keep 'operational in' up until the current signal sequence is completed. Except for the interface disconnect sequence, channels that force burst mode keep 'select out' up until the end of the operation. A rise of the incoming 'select out' in a control unit signals that the control unit can become selected to the channel by raising 'operational in.' If a control unit raises 'operational in,' it must suppress the propagation of 'select out' to the next control unit. If the control unit does not require selection, it must propagate 'select out' to the next control unit within 1.8 μ sec. See "Interface Time-Out Considerations."

Hold Out is a line from the channel to all attached control units and is used in conjunction with 'select out' to provide synchronization of 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 must not rise for at least 4 μ sec in general system configurations. The minimum down time of this signal may be optionally adjusted at installation time to a minimum of 2 μ sec to handle high-speed channel configurations. In all cases, the channel must be capable of providing the 4 μ sec timing for general system configuration (see "System Configuration").

Select In is a line that extends the 'select out' signal from the jumper in the terminator block (see Figure 2) to the channel. 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 to the channel that an I/O device has been selected (except for the control-unit-busy sequence). It must stay up for the duration of the selection. The selected I/O device is identified by the address byte transmitted over 'bus in.'

The rise of 'operational in' indicates that a control unit is selected and is communicating with the channel. 'Operational in' can rise only when the incoming 'select out' to the control unit is up and the outgoing

'select out' is down; i.e., the control unit must raise 'operational in' (except for the control-unit-busy sequence) in response to the rise of 'select out' and must block 'select out' from emanating to the next control unit. 'Operational in' can drop only after 'select out' drops.

When 'operational in' is raised for a particular signal sequence, it must stay up until all required information is transmitted between the channel and the control unit. If 'select out' is down, 'operational in' must drop after the rise of the outbound tag associated with the transfer of the last byte of information. With the exception of 'request in' or 'metering in,' all inbound signals must be down within 1.5 μ sec of the fall of 'operational in' at the control unit.

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.' The channel responds to 'address in' by 'command out.'

The rise of 'address in' indicates that the address of the currently selected I/O device is available on 'bus in.' 'Address in' must stay up until the rise of 'command out.' 'Address in' must fall so that 'command out' may fall. 'Address in' cannot be 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,' or 'service in.' A signal on 'command out' as a response to 'address in' during the initial-selection sequence 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.") This rise of 'command out' indicates that information on 'bus in' is no longer required to be valid. 'Command out' must stay up until the fall of the associated 'address in,' 'status in,' or 'service in.' It cannot be up concurrently with any other outbound tag line, except possibly during an interface disconnect sequence in which case 'address out' may be up. See "Interface Disconnect."

'Command out in response to 'address in' means *proceed* during a control-unit-initiated sequence. In the case of a channel-initiated-selection sequence, 'command out' indicates that 'bus out' defines the operational command to be performed. 'Command out' in response to 'service in' always means *stop* (see "Stop"). 'Command out' in response to 'status in' means *stack* (see "Stack Status").

When 'command out' is raised to indicate *proceed*,

stack, or *stop*, 'bus out' must have a byte of all zeros, but need not necessarily have correct parity. 'Bus out' is not to be checked for parity or decoded 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"). The channel responds with either 'service out' or 'command out' depending on whether it accepted the status byte.

The rise of 'status in' indicates that a byte of status information is available on 'bus in.' 'Status in' cannot be up concurrently with any other inbound tag line. 'Status in' must stay up until the rise of an out-tag, or in the control-unit-busy sequence until 'select out' falls. It must fall so that the responding out-tag may fall. During the control-unit-busy sequence, status information on 'bus in' must be valid until 'select out' (hold out) falls.

Service Out

'Service out' is a tag line from the channel to all attached control units and is used to signal the selected I/O device in recognition of a signal on 'service in' or 'status in.' 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.'

When 'service out' is sent in response to 'service in' during a read, or sense operation, or to 'status in,' the 'service out' signal must rise after the channel accepts the information on 'bus out.' In these cases, the rise of 'service 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 'service out' is sent in response to 'service in' during a write or control operation, the rise of 'service out' indicates that the channel has provided the requested information on 'bus out.' In this case, the signal must rise after the information is placed on the bus. 'Service out' must stay up until the fall of the associated 'service in' or 'status in.' 'Service out' cannot be up concurrently with any other out-tag, except during an interface-disconnect sequence (when 'address out' may be up).

A 'service out' response to 'status in' while 'suppress out' is up indicates to the control unit that the operation is being chained and that this status is accepted by the channel. See "Command Chaining."

Service In

'Service in' is a tag line from all attached control units to the channel and is used to signal to the channel

when the selected I/O device wants to transmit 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 must respond to 'service in' by 'service out,' 'command out,' or, during an interface disconnect, by 'address out.'

During read, read-backward, and sense operations, 'service in' rises when information is available on 'bus in.' During write and control operations, 'service in' rises when information is required on 'bus out.' 'Service in' cannot be up concurrently with any other inbound tag line. 'Service in' must stay up until the rise of either 'service out,' 'command out,' or 'address out.'

When, in the case of cyclic I/O devices, the channel does not respond in time to the preceding 'service in,' an overrun condition occurs. This condition must be recognized by the I/O device. In any case, 'service in' must not drop if an out-tag has not risen, nor may it rise if 'service out' has not dropped.

An overrun condition causes both the unit-check status indicator and the overrun sense indicator to be set. Data transfer stops after an overrun condition. For I/O devices that may overrun, the critical signal timings involved must be part of the SRL I/O device specifications.

Suppress Out

'Suppress out' is a line from the channel to all attached control units and 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 will be described in subsequent sections.

Clock Out

'Clock out' is a line from the channel to all attached control units and is used to provide the CPU interlock control necessary for changing the *enable/disable* states of the units (signal must be down to permit changing states). In addition, the control unit's transition between the enabled and disabled state requires the same prevailing conditions as for the off-line/on-line transition. (See "Off-Line/On-Line.")

The down state of 'clock out' must be at least 1 μ sec in duration.

Metering In

'Metering in' is a line from all attached control units and is used to condition the CPU meter for operation. The 'metering in' signal originates from each I/O device and/or control unit and is generated by the device from the time of acceptance of a command until the generation of 'device end' for that command. 'Metering in' may be raised concurrently with 'operational in' for any interface signaling sequence that does not involve 'device end,' such as 'test I/O' or a

control-unit-initiated status presentation. If raised, the duration of the signal must not exceed that of 'operational in.' 'Metering in' may be signaled by more than one control unit at a time. Refer to the SRL publication pertinent to each I/O device.

'Metering in' will not be raised:

1. Between the generation and acceptance of 'device end.'

2. Between the generation of 'device end' and the acceptance of the next command during chaining.

3. While a device is awaiting initiation of an automatic start. For example, transmission control units do not necessarily activate 'metering in' during the idle portion of prepare commands.

Metering Out

'Metering out' is a line from the channel to all attached control units and is used to condition all other meters in assignable units and I/O units. 'Metering out' is raised whenever the CPU meter is recording time.

Reserved Lines

Some of the 40 signal lines in the I/O interface are reserved. (See "Interface Connector Pin Assignments.")

All 40 signal transmission lines, with the exception of 'select out' ('select in' if low priority), must be carried through the control units from the IN cable connector to the OUT cable connector.

Signal Interlock Summary

The following rules for direct-current interlocking of signals must be used in the design of channels and control units using this I/O interface:

1. During the interface disconnect sequence, no more than one out-tag may be up at any given time, except for 'address out.'

2. No more than one in-tag may be up at any given time.

3. An in-tag may rise only when all out-tags are down, except for the control-unit-busy sequence.

4. An in-tag may fall only after the rise of a responding out-tag, except for 'status in' in the control-unit-busy sequence.

5. 'Service out' and 'command out' may rise only in response to the up level of an in-tag.

6. 'Address out' for a channel-initiated selection sequence may rise only when 'select in' and 'select out' are down at the channel.

7. Once 'address out' and 'select out' have risen for a channel-initiated selection sequence, 'address out' must stay up until after the rise of 'select in' or 'operational in' or the fall of 'status in.'

8. Once 'address out' has risen for the interface-disconnect control sequence, it must 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' can rise only if 'operational in' and 'select in' are down.

11. 'Operational in' cannot fall until either:

- a. 'Select out' falls and an out-tag response is sent for the last in-tag of any given signal sequence, or

- b. 'Operational out' falls, or

- c. An interface-disconnect sequence is given.

12. 'Operational in' cannot rise unless 'operational out' is up and must drop if 'operational out' drops.

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, due to wide variation in circuit tolerance or physically integrated channel control units, where transition time must be considered.

The general design rule is that the unit originating the signal (channel or control unit) must fully recognize internally the transition of the signal from one state to the other before any result of the transition can be returned from the receiving unit.

Interface Sequences

This section describes the detailed signal sequence for a complete I/O operation. These sequences include: initial selection, data transfers, and ending procedures. See Sequence Charts 1-3 and Flow Diagrams 1-8 in Appendix C.

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 the bus; 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 issues 'select out'; and when the incoming 'select out' appears at the addressed control unit, the control unit blocks its propagation and raises the 'operational in' line. When 'operational in' rises, the channel responds by dropping 'address out.' After 'address out' falls and the I/O device address is on 'bus in,' 'address in' may rise. For a multiplex operation, 'hold out' with 'select out' may drop any time after this point. After the channel checks the address, it responds by placing the command on 'bus out' and signaling on the 'command out' line. The selected control unit processes the command and drops 'address in,' which allows 'command out' to fall. After 'command out' drops, the control unit places the status information on 'bus in' and raises 'status in.' (Note that, if selection had been for a start I/O instruction, sufficient information is available at this point to complete instruction execution.)

If the channel accepts this status condition, it responds with 'service out.' 'Service out' allows 'status in,' to fall, completing the initial selection sequence. A 'command out' response from the channel also allows 'status in' to fall. See "Stack Status."

If during this channel-initiated selection sequence the I/O device is operating, the control unit presents busy status. When the control unit has, for the addressed device, status information outstanding from a previous operation or an externally initiated status condition, it presents the busy status (except to the all-zero command) along with the other status conditions in the status byte. See "Unit Status Conditions."

If the command is rejected by the control unit, for example, as a result of the detection of an invalid command, the control unit presents the unit-check status condition. No operation is initiated at the control unit and no ending status is generated.

An immediate-type command or command-immediate is a command whose execution meets the following requirements:

1. Execution requires no more information than that in the command byte; that is, no data or information bytes are transferred.

2. Channel-end time coincides with initial-status time; and on a normal operation, at least 'channel end' instead of zero status will be in the initial-status byte.

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

Control Unit Busy Sequence

If an I/O device is addressed and the control unit to which it is attached is busy or has status pending for an I/O device other than the one addressed, the control unit responds to 'select out' from the channel with 'status in.' The status byte indicates the busy condition. See "Busy" under "Status Byte." The control unit can present this status byte in either of two ways, depending on the design of the control unit; it can present status information as in the initial selection sequence or it can respond with the shorter control-unit-busy sequence.

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

The control-unit-busy sequence begins when the channel places the I/O device address on 'bus out' and raises 'address out.' 'Select out' is then raised. 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.

After accepting the status byte, the channel drops 'select out.' The control unit responds by dropping 'status in' and disconnecting from the interface. The channel must keep 'address out' up until 'status in' drops, thus completing the control-unit-busy sequence.

Control-Unit-Initiated Sequence

When any control unit requires service, it raises 'request in' to the channel. The next time 'select out' rises at any control unit requiring service and no I/O selection is being attempted by the channel ('address out' down), the control unit places the address of the I/O device on 'bus in,' and signals both 'address in' and 'operational in.' When the channel recognizes the address, 'command out' is sent to the control unit, indicating *proceed*. After 'address in' drops, the channel responds by dropping 'command out.'

If the service request is for data, the sequence proceeds as described for "Data Transfer." The control-unit-initiated selection for data transfer occurs in multiplex mode.

If the service request is for status information, the sequence proceeds as defined for the status cycle in the ending procedure. See "Ending Procedure."

Data Transfer

Data transfer may be requested by a control unit after a selection sequence. 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' must be maintained until an outbound tag is raised in response.

To request data from the channel, 'service in' is raised, and the channel places the data on 'bus out' and signals with 'service out.' The channel maintains the validity of 'bus out' until 'service in' falls. After 'service in' falls, the channel responds by dropping 'service out.'

After selection, the control unit remains connected to the channel for the duration of the transfer of information. The information can be a single byte of data, a status report, an initiation of a new command, a string of data bytes, or a complete operation from initiation to reception of the final status report.

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 cannot disconnect from the interface before 'select out' (hold out) falls. However, the control unit may preserve the logical connection after the channel permits the control unit to disconnect — 'select out' (hold out) down — by holding up 'operational in.' In this manner the control unit can force burst mode. See "Burst Mode."

Depending on the duration of the connection, one of two modes of operation is established: multiplex or burst. (These modes are established so that the program can schedule concurrent execution of multiple I/O operations.)

If 'operational in' remains up for longer than the multiplexing time-out limit, selection is in burst mode. See "Interface Time-Out Considerations." If the selection time is less than this time-out limit, the selection is in multiplex mode.

Multiplex Mode

The 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. Channels that are not capable of operating in multiplex mode force burst mode by holding up 'select out' (hold out) until presentation of channel-end status conditions. The transfer of one or more than one data byte during a single interface sequence, where the time contributed by the control unit is less than 32 μ sec, is considered multiplex mode.

Burst Mode

Burst 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 multiplex operation. Medium-speed or buffered I/O devices, which may normally operate in either mode as determined by channel data-rate capabilities, 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. An interface-disconnect executed by the channel overrides the force-burst-mode condition of a control unit. See "Interface Disconnect."

Some channels can tolerate an absence of data transfer during a burst-mode operation, such as when reading a long gap on tape, for not more than approximately one-half minute. Equipment malfunction may be indicated when an absence of data transfer exceeds this time.

Ending Procedure and Asynchronous Status

The ending procedure may be initiated by either the I/O device or the channel. If the procedure is initiated by the I/O device, the end of operation is completed in one signal sequence, assuming that both channel-end and device-end status conditions occur together. If the procedure is initiated by the channel, the I/O device may still require time to reach the point where the proper status information is available, in which case a second signal sequence is necessary to complete the ending procedure.

One of three situations may exist at the initiation of the ending procedure (assume 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 the 'service in' line. The channel responds with '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 where it normally would send 'channel end,' the control unit places the ending status on 'bus in' and raises 'status in.' The channel responds with 'service out,' unless it is necessary to stack the status condition 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 operation.

For situations 2 and 3, status information is available at the control unit. The control unit places the ending status on 'bus in' and raises 'status in.'

If 'device end' does not occur with 'channel end,' 'device end' is presented when it is available and an additional status sequence is required.

Some status conditions are unrelated to any previous program-initiated command. One of these conditions is 'attention,' which is normally on console or communication devices. Another condition is 'device end,' which is generated whenever 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 and are subject to the same rules as far as presentation to the channel and stacking are concerned.

Addressing

An eight-bit address byte (plus parity) is used over the 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. Devices that do not share a control unit with other devices may be assigned any device address in the range 0-255, provided the address is not recognized by any other control unit. Logically, such devices are not distinguishable from their control unit, and both are identified by the same address.

2. Devices sharing a control unit (magnetic tape units and disk storage units) are assigned addresses within sets of contiguous numbers. The size of a set of contiguous numbers is equal to the maximum number of devices that can share the control unit, or 16, whichever is smaller. Furthermore, the set of addresses

starts with an address in which the number of low-order zeros is at least equal to the number of bit positions required to specify the size of the set. The high-order bit positions of an address within a set identify the control unit; the low-order bit positions designate the device on the control unit.

3. Control units designed to accommodate more than 16 devices may be assigned nonsequential sets of addresses. Each set consists of 16 addresses, or the number required to make the total number of assigned addresses equal to the maximum number of devices attachable to the control unit, whichever is smaller. The 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 devices attached by more than 15. For example, if a communication-type control unit 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 I/O device addresses can be assigned. However, if the full capacity of 56 I/O devices is to be installed, exactly 56 addresses will be assigned.

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

Except for the rules described, the assignment of channel and device addresses is arbitrary. The assignment is made at the time of installation and the addresses normally remain fixed thereafter.

Address Decoding

Control units must recognize an I/O device address that meets the following conditions:

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

The control unit does not respond to any address *outside* its assigned set or sets. For example, if a control unit is designed to control devices that have only bits 0000-1001 in the low-order position 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 an address ('select out' is propagated through all control units, and back to the channel on 'select in'), the device appears as not operational. Not operational may include (in addition to addresses outside an assigned set):

1. An I/O device address not installed.
2. An I/O device address partitioned out of the system by the program, operator, or customer engineer (off-line, disabled, etc.).

The control unit must respond to those addresses *in* the set which is:

1. Ready.
2. Not ready but which can be made ready by means of an ordinary manual intervention. A not-ready device is indicated by the unit-check status and intervention-required sense indicator.

The control unit may respond to all addresses *in* the assigned set, regardless of whether the device associated with the address is installed. If a control unit responds to an address for which no device is installed, the unit-check status indicator must be turned on (as well as the appropriate sense indicator(s)).

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

Single-device-address control units must decode all eight bits of the address byte and the decoder can be set to any bit combination at the time of installation. See item 1 under "Address Assignment."

Commands

When 'command out' is up, the information on 'bus out' is the command byte. A channel issues 'command out' to initiate, continue, or terminate an operation in an I/O device.

Command Byte

The command byte requires decoding by the control unit only during a channel-initiated selection sequence (when the channel addresses the I/O device). At all other times, the byte is zero (parity is immaterial). The low-order bit positions indicate the type of operation; the high-order bit positions indicate a modification code which expands the basic operation at the control unit or I/O device level.

The actual modifier codes and the particular modes set or the controls performed for them must be specified in the SRL publication for each control unit and/or I/O device.

The command byte on the interface is defined as follows:

	BIT POSITION								
	P	0	1	2	3	4	5	6	7
Test I/O*	1	0	0	0	0	0	0	0	0
Sense	P	M	M	M	M	0	1	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
Control	P	M	M	M	M	M	M	1	1

M Modifier bit

P Parity bit

*Although test I/O may be a CPU instruction, it appears over the interface as a command. All references to test I/O in this manual are treated as a command.

'Sense,' 'read,' and 'control' commands with all-zero modifier bits are decoded on all I/O devices as follows:

	BIT POSITION								
	P	0	1	2	3	4	5	6	7
Basic Sense	0	0	0	0	0	0	1	0	0
Basic Read*	0	0	0	0	0	0	0	1	0
No-Operation Control	1	0	0	0	0	0	0	1	1

*Used for initial program load operations in addition to basic read operations.

Basic Operations

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

The low-order bit positions of this command byte specify the type of operation. (See command byte format.) The high-order bit positions (modifier code) expand the interpretation of the basic operation at the control unit or I/O device level.

Note: The basic operations are expressed by the commands: 'read,' 'read backward,' 'write,' 'control,' 'sense,' and 'test I/O.' This manual treats 'test I/O' as a command on the I/O interface, although it may originate as either a CPU instruction or a channel-initiated sequence for the purpose of clearing or stacking interruption conditions.

A command with invalid parity is not recognizable and therefore not executable.

Note: Special diagnostic commands which by intent permit errors to occur on the interface or introduce the possibility that subsequent commands may be executed erroneously, must be interlocked in such a fashion as to prevent inadvertent use of them.

Read

The 'read' command initiates execution of data transfer from the control unit to the channel and the data is obtained, for example, from the record source of the particular I/O device in operation.

A 'read' command, with all modifiers set to zero, is a basic 'read' command that is also used as an initial-program-loading (IPL) 'read' command. This command, if executed on an I/O device after a system reset, performs the initial-program-loading read as specified for that particular I/O device as stated in the appropriate SRL publication.

Read Backward

The 'read backward' command initiates an operation in the same manner as the 'read' command, except that the data bytes are transferred to main storage by the channel in reverse order to that for 'read' command. The control unit may be designed to cause mechanical motion in the I/O device in the opposite direction to

that for a 'read' command or may be designed to operate in the device precisely as with a 'read' command.

Unless otherwise noted, any description that refers to 'read' also applies to 'read backward.'

Write

The sequence of signals over the I/O interface to perform a write operation is the same as for a read operation. For a write operation, however, the data is sent from the channel to the control unit for operations such as recording or comparing by the selected I/O device.

Control

The control operation proceeds similarly as in a write operation, except that the command modifier bits received by the control unit are decoded to determine which of several possible functions is to be performed. The function 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 involving the transfer of data, the channel-end status may be presented during the initial selection sequence.

The timing in the control unit for the bytes transferred during a control operation is normally such that the byte rate for this operation is no faster than the normal read or write rate for the same I/O device. Input/output devices may be excepted from this restriction as determined on an individual basis, as specified in the SRL publications.

A 'control' command with all-zero modifiers performs no operation at the I/O device, except to satisfy any previously indicated chaining operations and to allow 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 proceeds exactly as a 'read' command, except that the data is obtained from sense indicators rather than from a record source.

The purpose of the 'sense' command is to extract information that concerns unusual conditions detected during the last operation and the actual state of the device.

The basic sense command (modifier bits set to zero) initiates a sense operation on all I/O devices. The basic sense command must not initiate any operation other than to sense the sense indicators. The basic sense command to an available control unit must be accepted even though the addressed I/O device is in a not-ready state (mechanically unable to execute other operations). See "Sense Information." If the control

unit detects an error during the sense operation, 'unit check' is sent with the channel-end status condition.

The I/O devices that can provide special diagnostic sense information or can be instructed to perform other special functions by means of 'sense' may define modifier bits for the 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 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.

Test I/O

The 'test I/O' command relieves the addressed I/O device path of pending status information. If no pending status information is encountered along the I/O path being tested, a zero status byte for the selected I/O device is sent to the channel for processing. If status information is available, pending status bits for the selected I/O device are transmitted to the channel.

Note that the busy condition is defined differently with respect to 'test I/O' than for other commands. See "Busy."

The signal sequence to accomplish 'test I/O' is the same as any other channel-initiated-selection process, except that no operation is initiated.

Sequence Controls

The sequence controls described in this section are determined by the sequence of the signals over the interface.

Proceed

Whenever 'command out' responds to 'address in' at any time other than during a channel-initiated-selection sequence, it means *proceed*.

Proceed indicates to the I/O device to continue the normal servicing sequences on the interface.

Stop

Stop is indicated by 'command out' in response to 'service in' or as a result of an interface disconnect, which occurs before the normal 'channel end' for the operation in process. See "Interface Disconnect."

Stop 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 must proceed to its normal ending point without sending any further 'service in' signals to the channel. The I/O device remains busy until the ending status is available, presented to, and accepted by the channel.

During data operations, 'command out' is transmitted in response to the first 'service in' that is provided after the designated operation is complete. (If 'select out' is down or goes down after this sequence, 'operational in' must drop on force-burst-mode type operations on I/O devices that cannot meet the time-out requirements as indicated in "Interface Time-Out Considerations." Also, burst-mode I/O devices that have relatively long times between stop and ending status and have no time-dependent chaining requirements must drop 'operational in' at this time. See appropriate SRL publication.

Stack Status

'Stack status' is indicated by a 'command out' response to 'status in.'

The 'stack status' signal causes retention of status information at the control unit or I/O device until that status is accepted on a subsequent status cycle with a 'service out.' When 'stack status' occurs, the control unit disconnects from the interface after 'select out' is down. 'Command out' remains up until 'operational in' falls. An attempt by the control unit to initiate a selection sequence to present the status again is under control of 'suppress out.' See "Suppress Status." 'Command out' is not signaled to a zero-status byte that has been provided in response to a command other than 'test I O.'

Note: Rather than delay completion of an interface sequence, the channel will either stack or accept status.

Suppress Data

Operations whose rate of data transfer can be adjusted without overrunning (completely buffered I/O devices, start-stop devices, and the transfer of data for the basic sense command may fall in this category) are subject to suppression of data by 'suppress out' as follows:

1. 'Suppress out' is ignored for the first data byte of any selection sequence, unless the data transfer is contiguous with initial selection; that is, no deselection and reselection between initial selection and data transfer.

2. 'Suppress out' must be up either 250 nanoseconds before the rise of 'service out' or at least 250 nanoseconds before 'service out' falls to ensure suppression of subsequent data.

3. When 'suppress out' is up at the control unit and the operation is in burst mode (whether because 'select out' is up or because the control unit is forcing burst mode), the control unit must not raise 'service in' for subsequent suppressible data.

Accept Data

Raising 'service out' in response to 'service in' during a read, read backward, or sense operation indicates that the information placed on 'bus in' has been accepted by the channel.

Data Ready

Raising 'service out' in response to 'service in' during a write or control operation indicates 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 handle immediately interruption-causing status conditions, 'suppress out' may be raised. With this line up, the control unit must not attempt to initiate a selection sequence to present suppressible-type status information. Status is suppressible if 'stack status' is received. Status that contains 'channel end' is not suppressible until stacked, except when 'interface disconnect' is received for that address. Status that contains 'device end' is not suppressible, when chaining has been indicated, until it is stacked. Other asynchronous status may be suppressible at the option of the particular control unit without being stacked.

'Suppress out' must be up at least 250 nanoseconds before 'select out' rises at the control unit if suppression of status is to be insured. 'Suppress out' suppresses only the initiation of the selection of the interface by the control unit. If 'suppress out' rises after a status sequence has been started, the status sequence will proceed normally.

The relationship between 'request in' and 'suppress out' has been described in "Request In."

Accept Status

Raising 'service out' in response to 'status in' indicates that the status placed on 'bus in' has been accepted by the channel.

Command Chaining

'Command chaining' is indicated if 'suppress out' is up when 'service out' is raised in response to 'status in.' 'Command chaining' means that another command for the I/O device in operation will immediately follow the presentation of 'device end,' provided 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.

When command chaining is indicated at the time 'device end' is presented, this indication will be valid until reselection is made or until 'suppress out' falls

(minimum down-level to ensure break – 250 nanoseconds). Reselection of any I/O device attached to the control unit will reset the chained-command condition in the control unit. Status conditions such as 'unit check,' 'unit exception,' 'control unit end,' or 'attention' will terminate command chaining in the channel.

A multidevice control unit which is to operate on more than one subchannel must not:

1. Reset the chained-command condition on an address other than the one being selected.
2. Present 'control unit end' with a chaining device address.

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

If chaining is indicated when 'channel end' is presented ('device end' not generated) from a control unit shared by two or more devices, the I/O device that presents 'channel end' must be the next one to present device-end status, unless the control unit is addressed in the meantime on the same interface.

If chaining is indicated when 'device end' is presented, the control unit must ensure that the path to the device remains available until the chaining operation is initiated or until chaining is no longer indicated.

If chaining is indicated on an I/O device shared between more than one control unit or channel, the I/O device must remain available until the chaining operation is initiated or until chaining is no longer indicated.

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

Interface Disconnect

The control unit will recognize 'interface disconnect' when 'address out' is up and 'select out' (hold out) is down at least 250 nanoseconds before the completion of any signal sequence. In this case, 'address out' may be up concurrently with another out tag.

When 'operational in' drops, the channel may drop 'address out' to complete the interface-disconnect sequence. 'Address out' must be down for at least 250 nanoseconds before a new channel-initiated-selection sequence may be attempted.

The control unit responds to the 'interface disconnect' by removing all signals (with the possible ex-

ception of 'request in' and 'metering in') from the interface. 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' or 'operational in.' 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.

Any abnormal 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." The control unit will not generate any status as a result of the 'interface disconnect' if the interface-disconnect sequence is performed before initial status is accepted, or after device-end status for an operation that has been accepted by the channel.

The I/O device path will remain busy after it receives an 'interface disconnect,' while performing an operation, until the 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 nor is the I/O device made busy.

Note: If 'address out' is up concurrently with another out tag, the information on 'bus out' must remain valid until the associated in tag drops or until 'operational in' drops.

Selective Reset

'Selective reset' is indicated whenever 'suppress out' is up and 'operational out' drops. This condition causes 'operational in' to fall, and the particular I/O device in operation and its status to be reset. The operation in process will proceed to a normal stopping point, if applicable, with no further data transfer. 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 will be in a busy state throughout this procedure.

'Device end' may be returned after the reset. To be effective, 'suppress out' must rise at least 250 nanoseconds before 'operational out' drops and must remain up until at least 250 nanoseconds after 'operational out' rises. 'Operational out' must stay 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 not changed by a selective reset.

'Selective reset' is issued only as a result of a malfunction detected at the channel or a time-out by the channel.

The interpretation of the 'selective reset' must be defined in the SRI publication for the I/O device.

System Reset

'System reset' is indicated whenever 'operational out' and 'suppress out' are down concurrently, and the I/O device is in the *on-line* mode. This condition causes 'operational in' to fall, and all control units and their attached I/O devices, along with their status, to be reset. The control units will be in a busy state for the duration of their reset procedure. 'System reset' prepares an I/O device for an initial-program-loading sequence.

'System reset' is performed when the system reset key is pressed, when the power for the system is turned on, when the channel is *off-line* to the I/O interface, and as a part of the initial-program-loading procedure. The ready or not-ready state of the control unit is not changed by a system reset.

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

The interruption of the 'system reset' must be defined in the SRL publication for the 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 device or implied control unit whose address appeared on 'bus in' (with 'address in') during the polling or selection portion of the sequence. In the case of the control-unit-busy sequence when no 'address in' occurs, it is assumed that the status pertains to the addressed device or implied control unit.

Note: The information in this manual pertains to control units attached to only one channel interface, unless otherwise stated.

Status Byte

The status byte has the following format:

BIT POSITION	DESIGNATION
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 in six situations:

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 signal 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 then interrogated, is now free.

5. To present any previously stacked status when allowed to do so.

6. To present any externally initiated status to the channel ('attention' and 'device end' because of not-ready-to-ready transition).

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 timing and causes of these conditions for each type of device are specified in the SRL publication for the I/O device.

When the device is accessible from more than one channel, status because of 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' or 'device end' because of transition from the not-ready to the ready state, depends on the type of device and condition, and must be specified in the SRL publication for the I/O device.

Note: Control units must provide interlocks in their design 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

'Attention' is generated when some asynchronous condition occurs in the I/O device. 'Attention' is not associated with the initiation, execution, or termination of any I/O operation.

The attention condition cannot be indicated to the channel while an operation is in progress at the I/O device or control unit. Otherwise, the handling and presentation of the condition to the channel depends on the type of I/O device. An I/O device shared between more than one channel path must present the attention status to all channel paths as defined in the individual SRL device publication. Presentation of 'attention' causes command chaining to be suppressed. Attention can occur with 'device end.' Depending on the I/O device application, 'attention' may or may not be presented until command chaining is no longer indicated.

Status Modifier

'Status modifier' is used by control units in three situations:

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 procedure to differentiate a busy control unit from a busy device.
3. Control units designed to recognize special ending conditions ('search equal' on a disk) present the status-modifier bit with 'device end' when the special condition occurs.

In the first case, provision is made for certain types of control units that are not capable of providing 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. This condition occurs when the control unit is required to perform a function that does not involve the i/o interface (backspace file on magnetic tape). The condition may also occur when the control unit has status pending for an i/o device other than the one addressed.

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 device-end status cycle. Presence of 'status modifier' and 'device end' indicates that the normal sequence of commands must be modified.

Control Unit End

The control-unit-end condition is provided only by channel-shared control units or control units shared by i/o devices, and only when one or both of the following conditions occurs:

1. The control unit was interrogated while it was in the busy state. Interrogated in the busy state means that a previous channel-initiated-selection sequence was issued to a device on the control unit and the control unit responded with 'busy' and 'status modifier' in the unit status byte. See "Status Modifier."
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.

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 its duration is less than approximately 2 milliseconds.

The i/o device address associated with 'control unit end' is determined as follows:

1. The 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 selection sequence, the i/o device address to be used when presenting this status is allowed to be any legitimate address associated with the control unit. (A legitimate address is any address the control unit is capable of recognizing, regardless of whether or not the i/o device is actually attached.)

3. If 'control unit end' is to be presented during a channel-initiated selection sequence, the i/o device address will be the same as the address issued with 'address out.' This channel-initiated-selection sequence may be either a normal full-length selection sequence or the control-unit-busy sequence.

The control-unit-end condition can be signaled with 'channel end,' 'device end,' or between the two. When 'control unit end' is signaled by an i/o interruption in the absence of any other status conditions, the interruption may be identified by any address assigned to the control unit. A pending 'control unit end' causes the control unit to appear not available for initiation of new operations. 'Control unit end' causes command chaining to be suppressed.

Busy

'Busy' can occur only during a channel-initiated-selection sequence and indicates that the i/o device or control unit cannot execute the command because a previously initiated operation is being executed or because status conditions exist (or in the case of switched control units, because the path is unavailable). An 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 'test I/O' only if a previously initiated operation is still being executed and no end status is available.

The busy condition causes command chaining to be suppressed.

Channel End

'Channel end' is caused by the completion of the portion of an I/O operation involving transfer, if any, of data or control information between the I/O device and the channel.

Each I/O operation causes only one channel-end signal to be generated. The channel-end condition is not generated unless the command is accepted (status byte contains either all zeros or 'channel end' and not busy). The exact time during an I/O operation when 'channel end' is generated depends on the operation and the type of device. For operations such as writing, some I/O devices generate the channel-end condition when the block has been written. On other I/O devices that later verify the writing, 'channel end' may or may not be delayed until verification is performed, depending on the I/O device. On devices such as those equipped with buffers, the channel-end condition occurs on completion of data transfer between the channel and the buffer. During control operations, 'channel end' is usually generated after the control information is transferred to the control unit; although for short operations, the 'channel end' may be delayed until completion of the operation. Operations that do not cause data to be transferred can provide the channel-end condition during the initial-selection sequence.

Device End

'Device end' is caused by the completion of an I/O operation at the I/O device or, on some devices, by manually changing the device from the not-ready to the ready state. The device-end condition normally indicates that the I/O device has completed the current operation.

Each I/O operation causes only one device-end condition. 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. In the case of data-transfer operations on some devices, the device terminates the operation at the time 'channel end' is generated, and both 'device end' and 'channel end' occur together. On buffered I/O devices, the device-end condition may occur on completion of the me-

chanical operation. For control operations, 'device end' is generated at the completion of the operation at the device. The operation may be completed at the time 'channel end' is generated or later.

When command chaining occurs, only the 'device end' of the last operation of the chain is normally made available to the program. When the device-end signal is received in the absence of any unusual conditions, it causes the channel to initiate a new I/O operation. If an unusual condition is detected during the initiation of a chained command, the chain is terminated without 'device end.'

An I/O device shared between more than one channel path and which has a 'device end' generated because of the device going from a not-ready to a ready state must present a 'device end' to all attached channels as defined in the individual SRL publication for the I/O device.

If a device is addressed while in a busy state, a 'device end' must be signaled to the path that initiates the command when the I/O device becomes not busy.

Note: "Not ready" means that an I/O device requires operator intervention to become operational. The not-ready condition can occur, for example, because of any of the following actions:

1. An unloaded condition on magnetic tape.
2. Card equipment out of cards or stacker full.
3. Printer out of paper.
4. Error conditions which need operator intervention.

Unit Check

'Unit check' indicates that the I/O device or control unit has detected an unusual condition that is detailed by the information available to a sense command. 'Unit check' may indicate that a programming or an equipment error has been detected, that the not-ready state of the device has affected the execution of the command or instruction, 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.

An error condition causes the unit-check indication only when it 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 and is of significance to the program, the condition does not cause the program to be alerted after 'device end' has been cleared; a malfunction 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 device. When no interruption condition is pending for the addressed 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 device. In the case of no operation, the command is rejected and 'channel end' and 'device end' do not accompany 'unit check.'

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

If, during the initiation sequence, the device detects that the command cannot be executed, '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 execution has been started, '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' or 'unit check' with 'control unit end.'

When 'unit check' appears with 'channel end' and without 'device end,' a shared control unit must preserve the sense data and an available device path until after the 'device end' is accepted.

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 device responds by providing the busy bit and indicating the pending interruption condition, if any. The command code validity is not indicated.

Termination of an operation with the unit-check indication causes command chaining to be suppressed.

Programming Note

If a device becomes not ready on completion of a command, the ending interruption condition 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 device causes a unit-check indication.

Unit Exception

'Unit exception' means that the I/O device detected an unusual condition such as end of file. '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 can be generated only 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 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 normal execution of the operation occurs after the execution is started, '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' causes command chaining to be suppressed.

Sense Information

Data transfer during a sense operation provides information concerning unusual conditions detected in the last 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.

All sense information significant to the use of the I/O device normally is provided in the first two bytes. Any bit positions that follow those used for programming information contain diagnostic information, which may extend to as many bytes as needed. The amount and the meaning of the status information are peculiar to the type of device, and must be specified in the SRL publication for the I/O device.

A device which normally operates in multiplex mode, and transfers a single byte during data transfer, should transmit the maximum number of bytes of

sense data possible during the 32-microsecond multiplex time-out limitation.

The sense information that pertains to the last I/O operation or other unit action may be reset by the next command addressed to the control unit, provided the busy bit is not included in the initial-selection status byte, except where the command is a basic 'sense,' 'test I/O' or 'no-op' and it is addressed to the device that causes the sense. The sense information may also be changed as a result of asynchronous actions such as when 'attention' or not-ready-to-ready device-end status is generated.

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

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

Sense Byte

For most I/O devices, the first six bits of the first sense data byte (sense byte 0) are common to all I/O devices that have this type of information. The six bits are independent of each other and are designated as follows:

BIT	DESCRIPTION
0	Command Reject
1	Intervention Required
2	Bus-Out Check
3	Equipment Check
4	Data Check
5	Overrun

Sense Conditions

Command Reject

The device detected a programming error. A command was received which the device is not designed to execute ('read backward' issued to a direct-access storage device), or which the device cannot execute because of its present state (a 'write' issued to a file-protected tape unit). In the former case, 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 ('write' to a direct-access storage device without previously designating the data block).

Intervention Required

'Intervention required' is generated when the last operation could not be executed because of a condition that requires intervention at the I/O device. This bit indicates conditions such as an empty hopper in a card punch or the printer being out of paper. 'Intervention required' is also turned on when the addressed device is in the not-ready state, is in test mode, or is not provided on the control unit.

Bus-Out Check

'Bus-out check' results 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 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 sense bit indicates detection of an equipment malfunction logically between the I/O interface and the input/output medium. On output operations, this malfunction may have caused invalid data to be recorded. Detection of 'equipment check' stops data transmission and terminates the operation prematurely when the error prevents any meaningful continuation of the operation.

Data Check

'Data check' indicates that invalid data has been detected or is probable on the recording medium. This invalidity can be accurately detected only by reading the recording medium.

The control unit forces correct parity on data sent to the channel. On writing, the data-check condition indicates that incorrect data may have been recorded at the I/O device. 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

'Overrun' results when the channel fails to respond on time to a request for service from the I/O device. 'Overrun' can occur when data is transferred to or from a nonbuffered control unit that operates with a moving medium, and the total activity initiated by the program exceeds the capability of the channel.

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 in 'stop.' The overrun bit is also turned on when the device receives the new command too late during command chaining.

General System Considerations

Interface Time-Out Considerations

Signaling over the I/O interface is specified to be direct-current interlocked and thus is not time-dependent. Since the signaling is not time-dependent, a certain category of machine malfunctions may cause "hangup" of the channel and, unless detected and canceled, results in "hangup" of the system. The timing considerations involved in determining malfunction cases are described in the following paragraphs.

All references to particular time considerations, in this section, represent the maximum permissible time in worst-case situations. All control units must be 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. A maximum worst-case interface signal sequence of 32 microseconds, because of the control unit, is specified. This means that from the time 'select out' rises at a particular control unit for any signal sequence until the sequence is completed ('operational in' drops at the control unit), no more than 32 microseconds, because of the control unit circuitry or the sequencing method used, will be required by the control unit.

This applies to all multiplex sequences and to all selection sequences. An initial-selection sequence is defined as beginning with the rise of 'select out' at the particular control unit to the fall of 'status in' for initial status. However, a particular interface signal sequence may take longer because of other factors:

1. Delays introduced by the channel.
2. Delay because of the need for a burst-mode I/O device to capture the interface prior to reaching the specified record area of the recording medium (not greater than 500 milliseconds).

In addition to excessive delays which may occur within a particular signal sequence, excessive delays may exist between data cycles during burst-mode operation. The maximum allowable delay between data cycles in burst mode is 500 milliseconds. The channel should not indicate a malfunction unless a delay of interface activity exceeds approximately 30 seconds. Absence of activity on the interface may be because of other conditions (reading a long gap on magnetic tape created by successive erase commands).

Propagation of Select Out

During an initial-selection or scan sequence, if the control unit does not require service or selection, 'select out' must be propagated by the unit within 600 nanoseconds to meet performance requirements. In no case may propagation time exceed 1.8 microseconds per control unit. This time is measured 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 channel I/O interface.

Internal Cabling: The maximum allowable internal resistance, including all contact resistance, contributed by a channel or control unit is specified as 2.0 ohms for every signal line, except 'select out' and 'select in.' (Extended or high-performance configurations require that internal resistance be reduced to a minimum.) A total of 3.0-ohm maximum resistance is specified for the control unit for the combined 'select out' and 'select in.' The maximum resistance case for 'select out' '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 specified at 15 nanoseconds. The maximum skew between any two signal lines is specified at 1 nanosecond.

The SRL publication for each control unit and channel must define, for each possible interface implementation, the value of its internal resistance for the signal lines and for the 'select out/select in' line-pair.

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 will 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 must not be 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 will be determined by the 2.0-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.

Off-Line/On-Line

A control unit in an off-line mode must not interfere in the operation between the channel and other control units on the same interface. The control unit in the off-line mode must provide the following:

1. Logical bypass for 'select out.'
2. Gate off all other line drivers from interfering with activity on the interface.

The control unit transition to or from an off-line/on-line condition must not cause machine malfunction. The minimum following conditions must exist concurrently before the on-line-to-off-line transition can occur:

1. The on-line/off-line switch is set to OFF LINE.
2. No active communication on the interface because of this control unit, such as 'operational in,' or 'status in,' etc.
3. No stacked or pending status is on this unit. (Stacked status is that which has been presented to the channel but which has not been accepted. Pending status is that which is forthcoming or which has been generated but not presented.)
4. No command chaining is indicated for this unit.

The preceding conditions ensure that no machine malfunctions can occur because of an operator throwing the switch from ON LINE TO OFF LINE.

Power Effects

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

Transient and Spurious Signals: Each control unit must be designed so that, if proper procedures are followed, the process of individually powering up or down does not cause its interface driver or receiver circuits to generate noise on the interface signal lines.

Transient and Select-Out Pulse Splitting: Each control unit must ensure that the process of electrically bypassing 'select out' prior to 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 must ensure that 'select out' discontinuities, which may occur when another control unit on the interface is powering up or down, do not affect the propagation of 'select out.' This 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 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 (Figure 3).

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

1. Logically disconnect the unit from the interface (a panel indicator comes on). (The control unit can become logically disconnected when it and all connected devices have completed all operations, no status is pending, or stacked, and 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 interface, all its drivers except 'select out' are logically gated off. A logical disconnect may be accomplished as a result of going off-line by use of the on-line/off-line or the metering disable/enable switch.

2. Close 'select out' bypass circuit (mechanical contact, K1, Figure 3). The normal logical electronic bypass of 'select out' is still active when the control unit is logically disconnected.

3. Open the connection from the line to the 'select out' receiver terminator (mechanical contact, S1, Figure 3).

4. Clamp interface driver gates to ground by means of a mechanical contact, S2, Figure 3), if gated drivers are used to avoid transient signals on the interface lines.

5. Turn off power (remote/local power-control switch set to LOCAL). Refer to Systems Reference Library, *IBM Power Control Interface*, Form A22-6906.

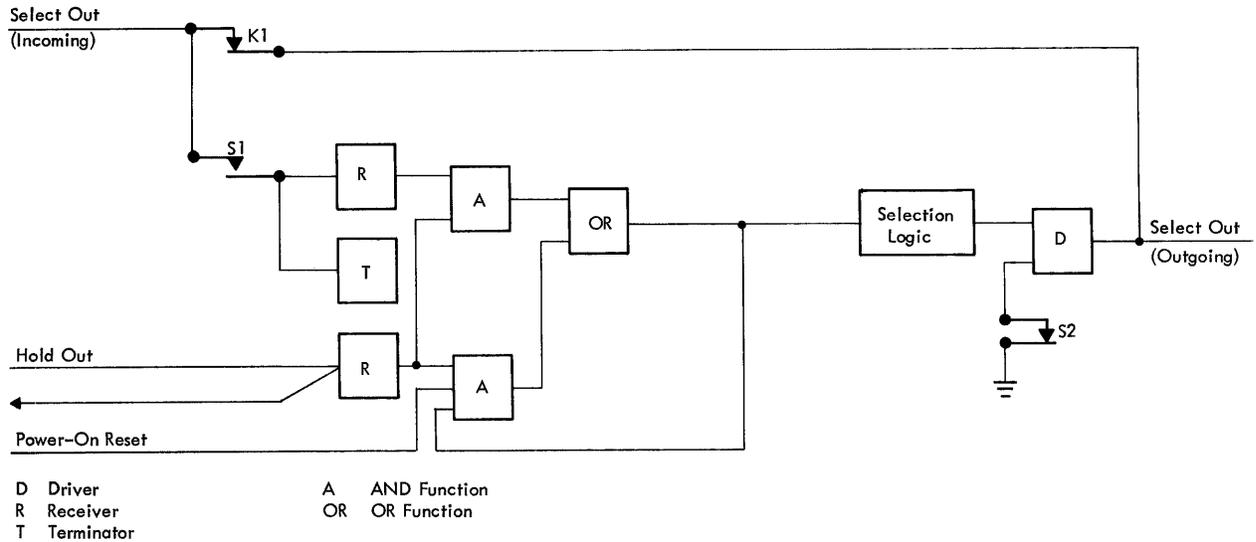


Figure 3. Representative Select-Out/Hold-Out Special Latch

For power on, the sequence is reversed:

1. Turn power on (remote/local power-control switch set to LOCAL). During the power-on sequence, a power-on reset pulse must automatically reset the control-unit circuitry, including reset of the special 'select out' latch, regardless of 'hold out.'
2. Unclamp driver gates. (Open mechanical contact, S2.)
3. Connect 'select out' receiver terminator (S1).
4. Open 'select out' bypass circuit (K1).
5. Logically connect the unit to the interface (panel indicator turns off).

If some method, such as automatic power sequenc-

ing rather than gated 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 any 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. In any case, the 'select out' bypass function must not be interrupted, but must be completed once it is started.

Appendix A. Electrical Specifications

Physical Requirements

Multiple Drivers and Receivers

Up to ten receivers must be able to be driven by one driver. The driver must be located at one of the extreme ends.

Up to ten drivers must be able to be connected to drive one receiver. The receiver must be located at one of the extreme ends.

Note: An end-of-line driver or receiver may be placed beyond the terminator. In this case, the distance between the end-of-line driver or receiver and the terminator must be less than six inches.

Receivers must be spaced at least three feet apart. No minimum requirement is set as regards the spacing between drivers. No minimum requirement is set as regards 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 six inches.

General Electrical Requirements

Voltage Levels

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

Cable

All lines must have a characteristic impedance of 92 ± 10 ohms and, with the exception of 'select out,' must be 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 is never to exceed a maximum line resistance of 33 ohms. The 33-ohm line resistance includes all contact resistance, internal cable resistance, and inter-unit cable resistance.

Terminating Networks

The terminating network must present an impedance of 95 ohms ± 2.5 percent between the signal line and ground, and must be capable of dissipating 390 milliwatts.

Ground Shift and Noise

The maximum noise (measured at the receiver input) coupled onto any signal line must 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" under "General Electrical Requirements" 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 1.7 volts or more. See Diagram A.

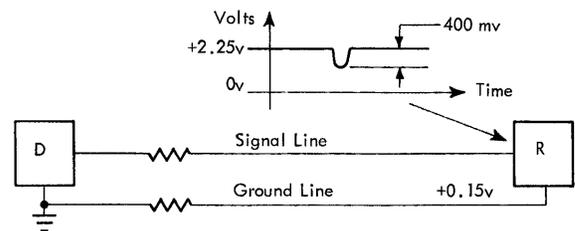


Diagram A. Negative Noise

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 0.7 volt or less. See Diagram B.

Note: The noise measurements are made at the input to the receiver. A combination of the dc level and ac noise must not exceed 0.7 volt for the down level and must not be less than 1.7 volts for the up level.

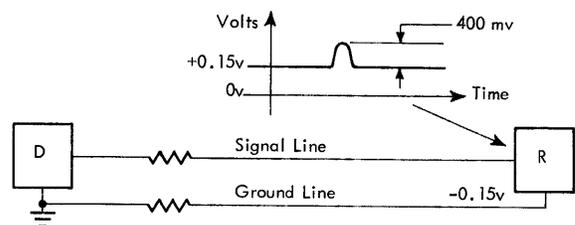


Diagram B. Positive Noise

Note: Noise may be generated by circulating currents in the grounding network if the proper grounding rules are not adhered to. In IBM System/360, the ground lead (green with yellow tracer) in the power source cable to the CPU is connected to the machine (frame) ground. All signal lead shields are connected to circuit (electronic) ground. In the CPU these two grounds, circuit and machine, are connected at one point only.

Devices attached to the system should follow the same convention – signal shields connected to circuit ground, and power ground connected to machine frame. These two grounds should not be connected in the device. Excessive noise may be generated by circulating currents in the grounding network when circuit and machine grounds are commoned in both CPU and attached units.

Interface Circuit Requirements

Receivers

An input voltage (relative to receiver circuit ground) of 1.7 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 should not be 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.

The receiver input must not require a positive current (Diagram C) larger than $+0.42$ milliamperes at an input voltage of $+3.11$ volts.

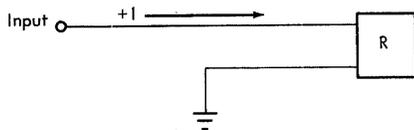


Diagram C. Receiver

Negative receiver input current at $+0.15$ volt must not exceed -0.24 milliamperes. In addition, receiver input impedance must be larger than 4.0 kilohms and less than 20 kilohms.

Receivers must be designed to ensure that no spurious noise is generated on the line during a normal power-up or power-down sequence.

Drivers

In the logical zero state:

1. The output voltage must not exceed 0.15 volt at

a load of $+240$ microamperes. (See Diagram D for current polarity definition.)

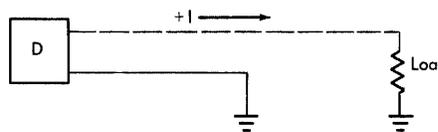


Diagram D. Current Polarity

In the logical one state:

1. The output voltage must be 3.11 volts or more at a load of $+59.3$ milliamperes (two terminators, ten receivers).

2. The output voltage must not exceed 5.85 volts at a load of $+30$ microamperes (one receiver, no terminator).

3. The output voltage must not exceed 7.0 volts at a load of $+123.0$ milliamperes during an over-voltage internal to the drivers.

Drivers must be designed to ensure that no spurious noise is generated on the line during a normal power-up or power-down 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 Diagram E.)

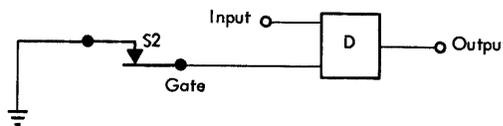


Diagram E. Driver Gate

For a normal power-down sequence:

1. Logically ensure that the driver is in the zero state.
2. Close contact S2. (See Diagram E.)
3. Turn power off.

For a normal power-up sequence:

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

Fault Conditions

A grounded signal line must not damage drivers, receivers, or terminators.

With one driver transmitting a logical one, loss of power in any single circuit driver, receiver, or termi-

nator on the line must not cause damage to other components.

With both terminators connected, line operation must not be affected by power off in any drivers or receivers on the line.

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.

Note: Because of the nature of the 'select out/select in' line, negative noise tolerance has been neglected.

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

Receiver

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

Driver

The select out driver must be capable of withstanding 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 must not exceed 0.15 volt at a load of 1 milliamper.

For the logical one state:

1. Output voltage of a channel driver or the driver of a control unit contained within a channel frame must exceed 3.9 volts at a load of 41 milliamperes.

2. The output voltage of a control unit driver not contained within a channel frame must exceed 3.7 volts at a load of 41 milliamperes.

The output voltage of a select out driver should not exceed:

1. 5.8 volts at a load of 0.3 milliamper.

2. 7.0 volts at a load of 72 milliamperes during an over-voltage internal to the driver.

Terminator

A 95-ohm ± 2.5 percent, 390-milliwatt terminator to ground must be 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' must not be terminated, including the driver end of 'select out' located in the channel. Also, the jumped 'select out' or 'select in' and the bypassed 'select out' or 'select in' path is not terminated.

Circuits

Figures 4 through 6 show representative circuits used to drive, receive, and terminate the lines between the channel and attached control units.

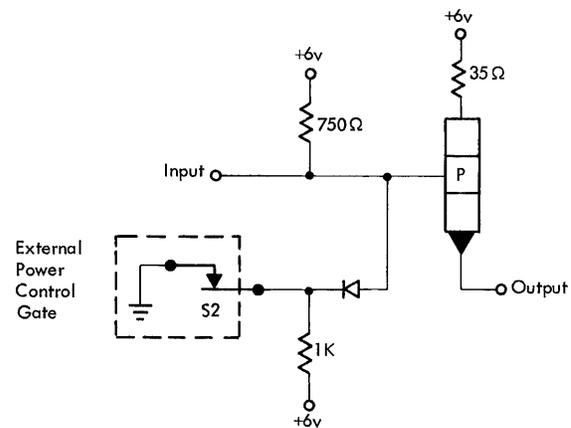


Figure 4. Line Driver

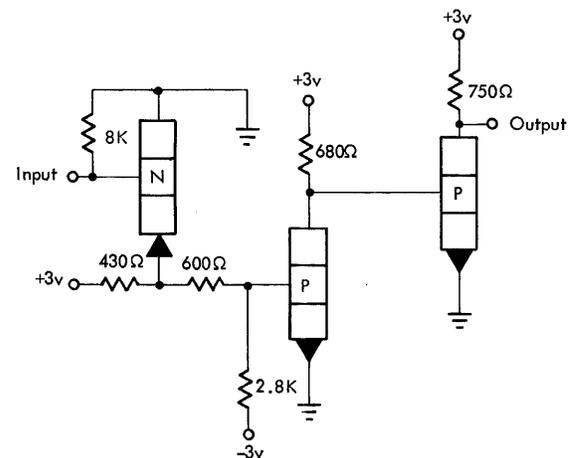


Figure 5. Line Receiver

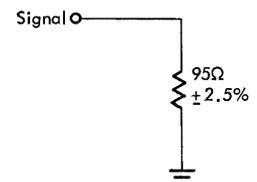


Figure 6. Line Terminator

Pin Assignments

Pin assignments for the I/O interface are shown in Figure 7. They are shown as viewed from the connector side of the channel and control unit tailgates. Two

interface cables 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).

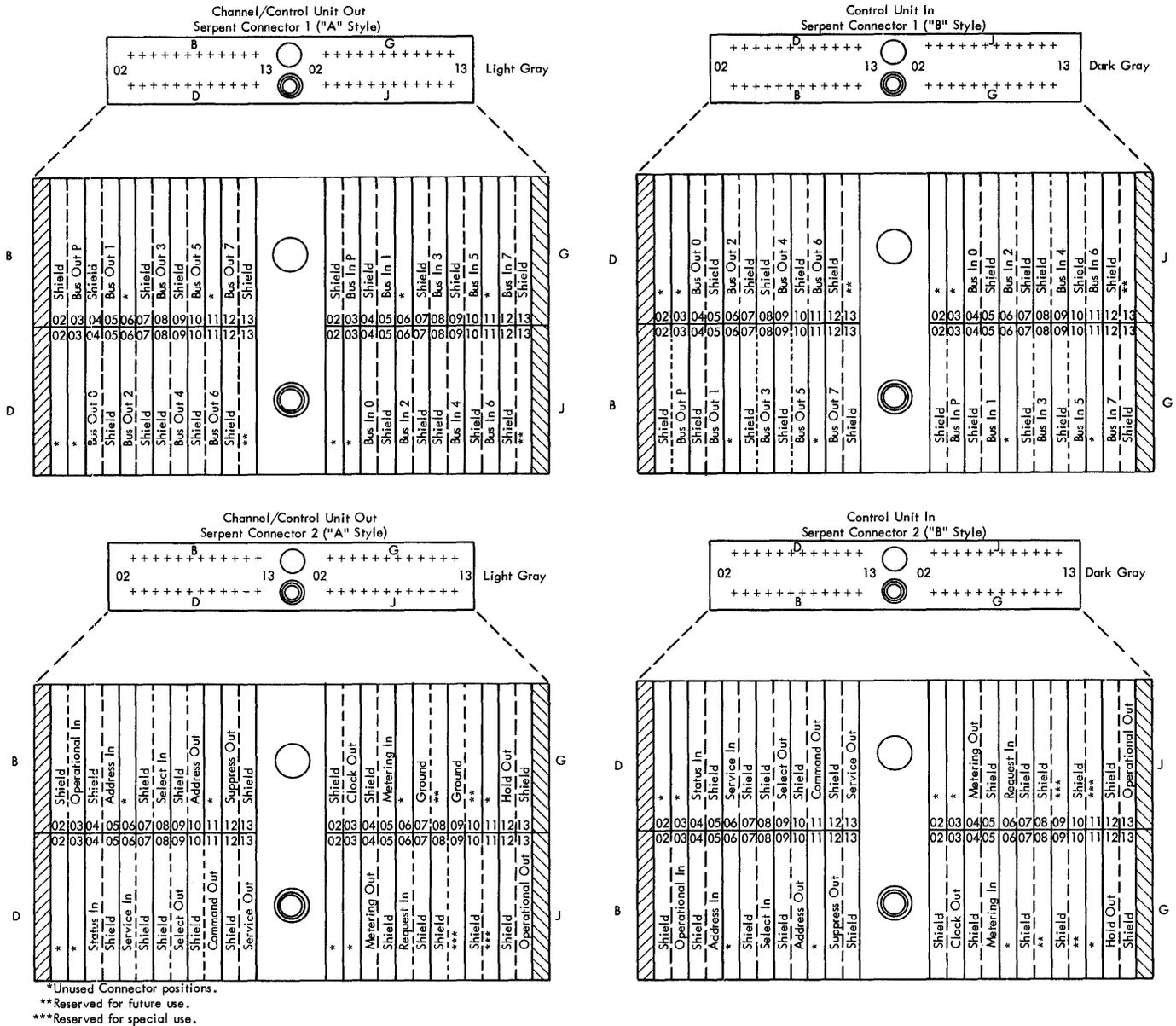


Figure 7. I/O Interface Connector Pin Assignments

Appendix B. Signal Cabling and Connectors

Cabling

Cable Halves

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

Mounting

In normal applications, the blocks are mounted in the horizontal plane to provide a smoother bend into the coaxial cables. (See Figure 8.) Vertical mounting produces unusual bending configurations, requiring careful routing and strain relieving of the external cables.

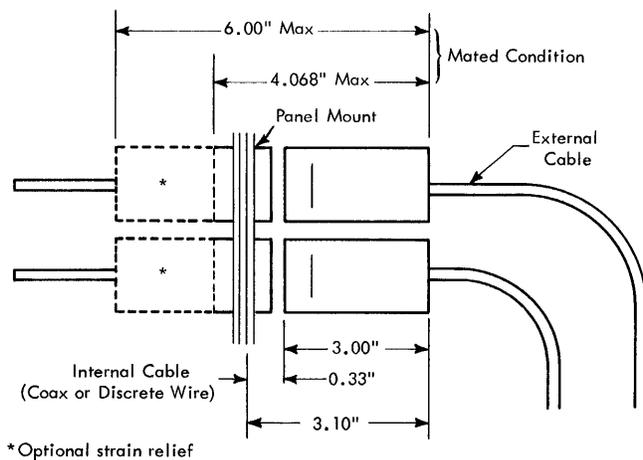


Figure 8. Mounted Cable Halves

Figure 9 shows an exploded view of the panel mounting.

Off-Line Utilization

This cable connector has the unique feature of being able to mate "cable half" to "cable half" for off-line utilization or for the physical by-passing of machine units. (See Figure 10.)

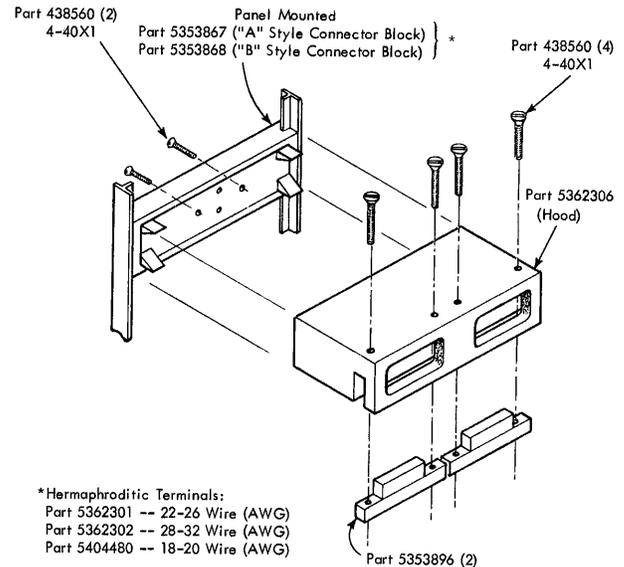


Figure 9. Panel Mount with Optional Strain Relief

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

Typical connections are shown in Figures 10 and 11. (Information concerning the EPO cable shown in Figure 11 is contained in SRL publication, *IBM System/360 Power Control Interface, Original Equipment Manufacturers' Information*, Form A22-6906.)

Connectors

Connector Blocks

Three styles of connector blocks are available: "A," "B," and "B-." The "A" and "B" designations are used to identify proper mating arrangements since the physical hardware is identical. The two styles are differentiated by the color coding of the blocks; the "A" style is light gray and the "B" style is dark gray. (See Figure 12.)

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

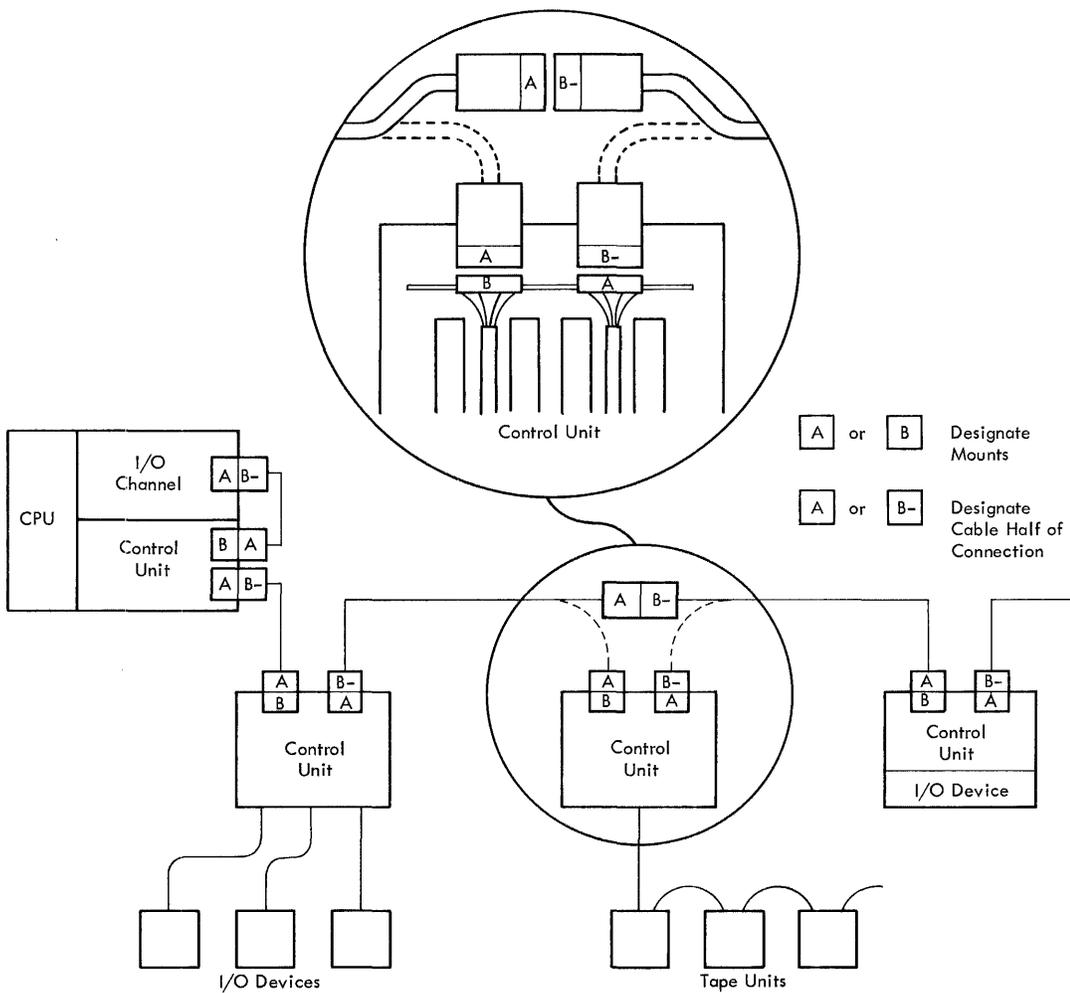


Figure 10. Off-Line Utilization

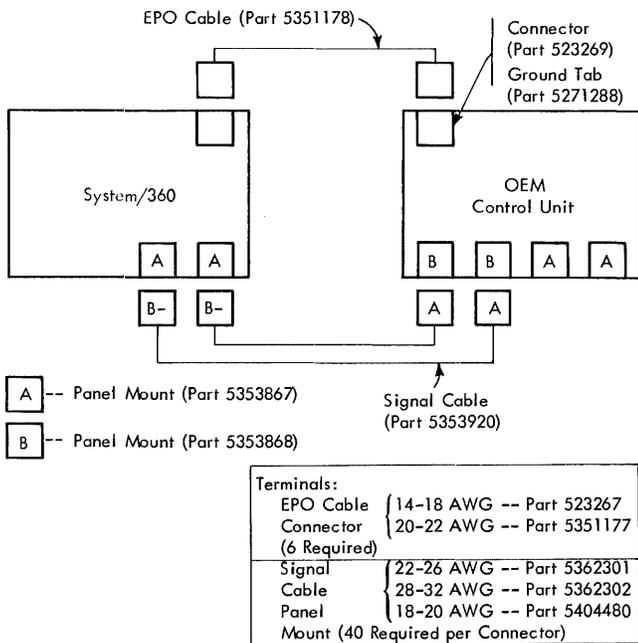


Figure 11. Typical Connection

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

Both the "A" and the "B" style blocks are used for panel mounts. The "A" style may also be used for a cable end. The "B-" style is used only as a cable end. (Refer to "Off-Line Utilization.")

Capacity

Forty-eight (48) individual positions (serpent contacts) are provided for in 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, 40 serpent contacts are required on each connector to individually terminate the shield and signal wires of the 20 coaxial wires.

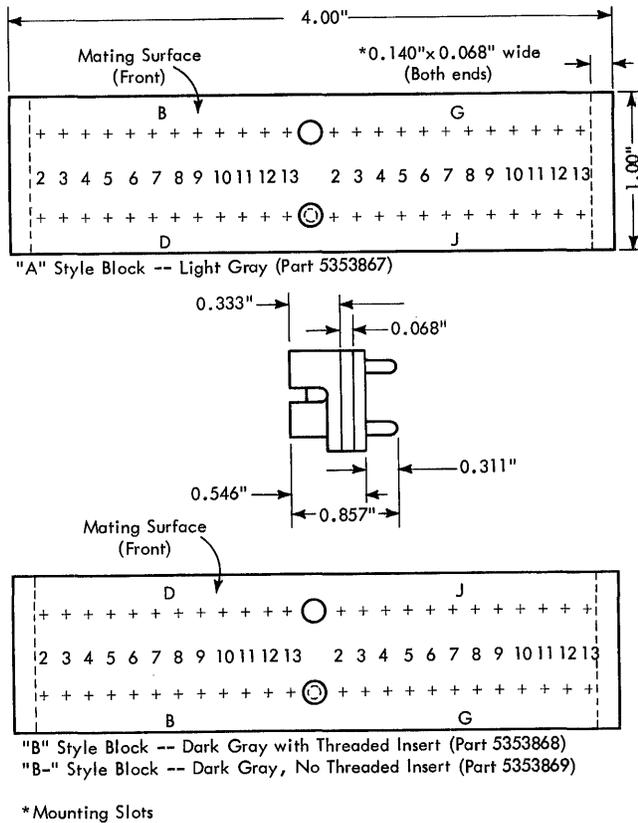


Figure 12. Connector Blocks and Contact Location

Serpent Contacts

Description

The serpent contact is a hermaphroditic, gold-plated phosphor bronze, dual-mating surface contact. (See Figure 13.)

Wire Termination

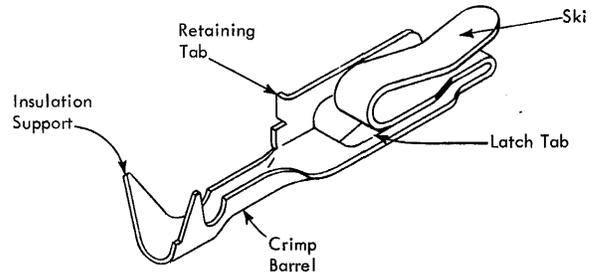
Termination is accomplished by the bare-wire crimp method. Three contacts are available to cover the range of solid or stranded wire sizes required:

IBM PART	WIRE SIZE (AWG)	INSULATION RANGE
5404480	18-20	0.042 to 0.103
5362301	22-26	0.028 to 0.103
5362302	28-32	0.026 to 0.090

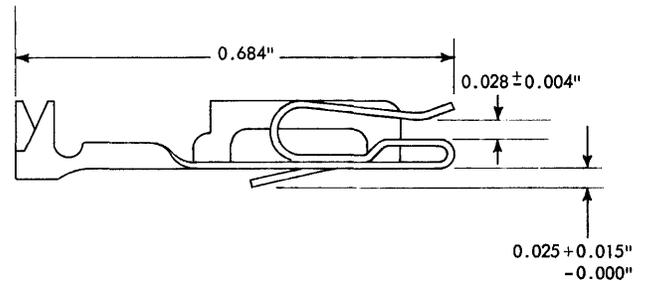
Terminators

The System/360 standard I/O interface line termination is provided by two assemblies (Part 5440649 and 5440650) shown in Figure 14.

A. Part Identification



B. Dimensions



C. Mating Configuration

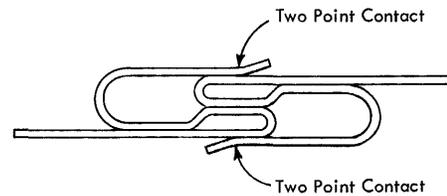


Figure 13. Serpent Contact

Electrical Specifications

Voltage Ratings: 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) will 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 1 hour at a temperature of 38° C and 85-90 percent relative humidity.

Grounding: All surfaces of the connectors are non-conductive plastic; therefore, no grounding is necessary.

Note: Direct all questions concerning cable and

connector prices and availability to IBM Corporation, Industrial Products Marketing, Systems Development Division HQ, 1000 Westchester Avenue, White Plains, New York.

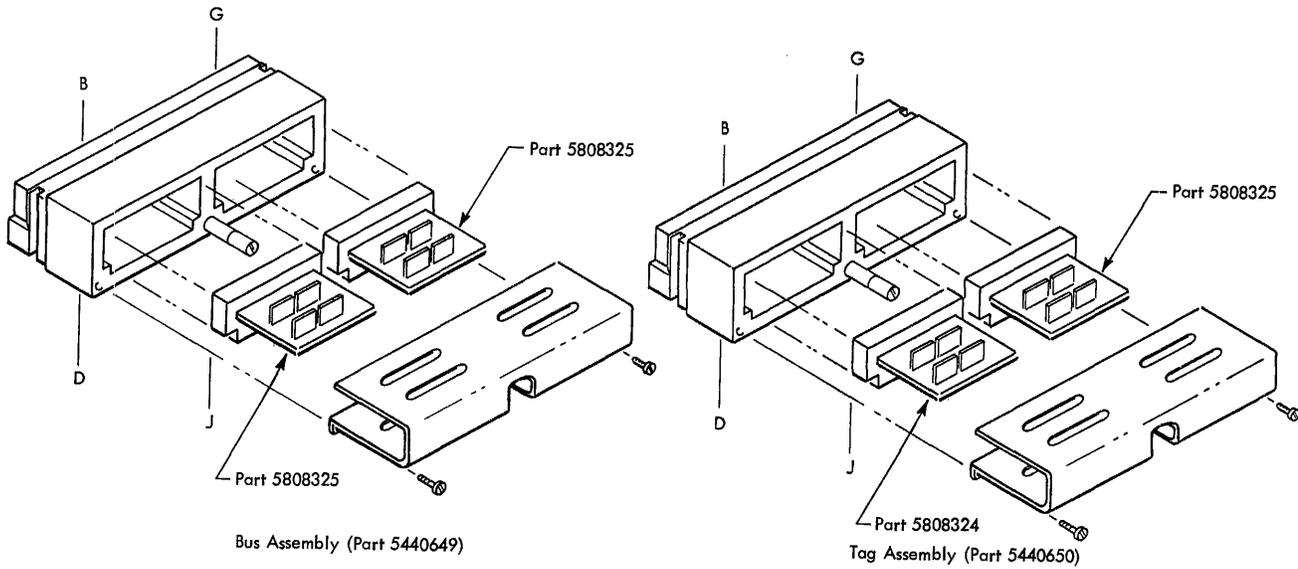
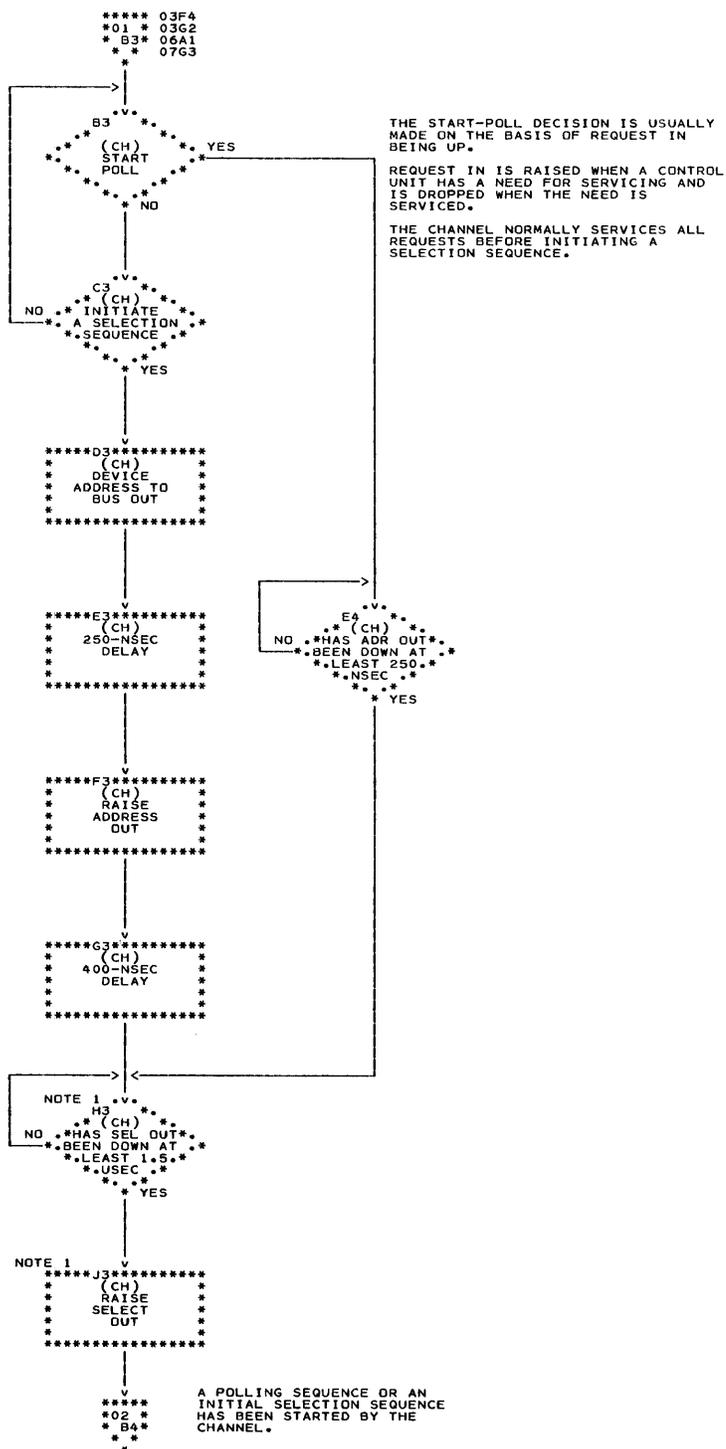


Figure 14. Terminator Assemblies

Appendix C. Supplementary Application Information

AT DECISION BLOCKS INVOLVING SIGNAL LINES, THE QUESTION TO BE RESOLVED IS — IS THE LINE LOGICALLY UP.

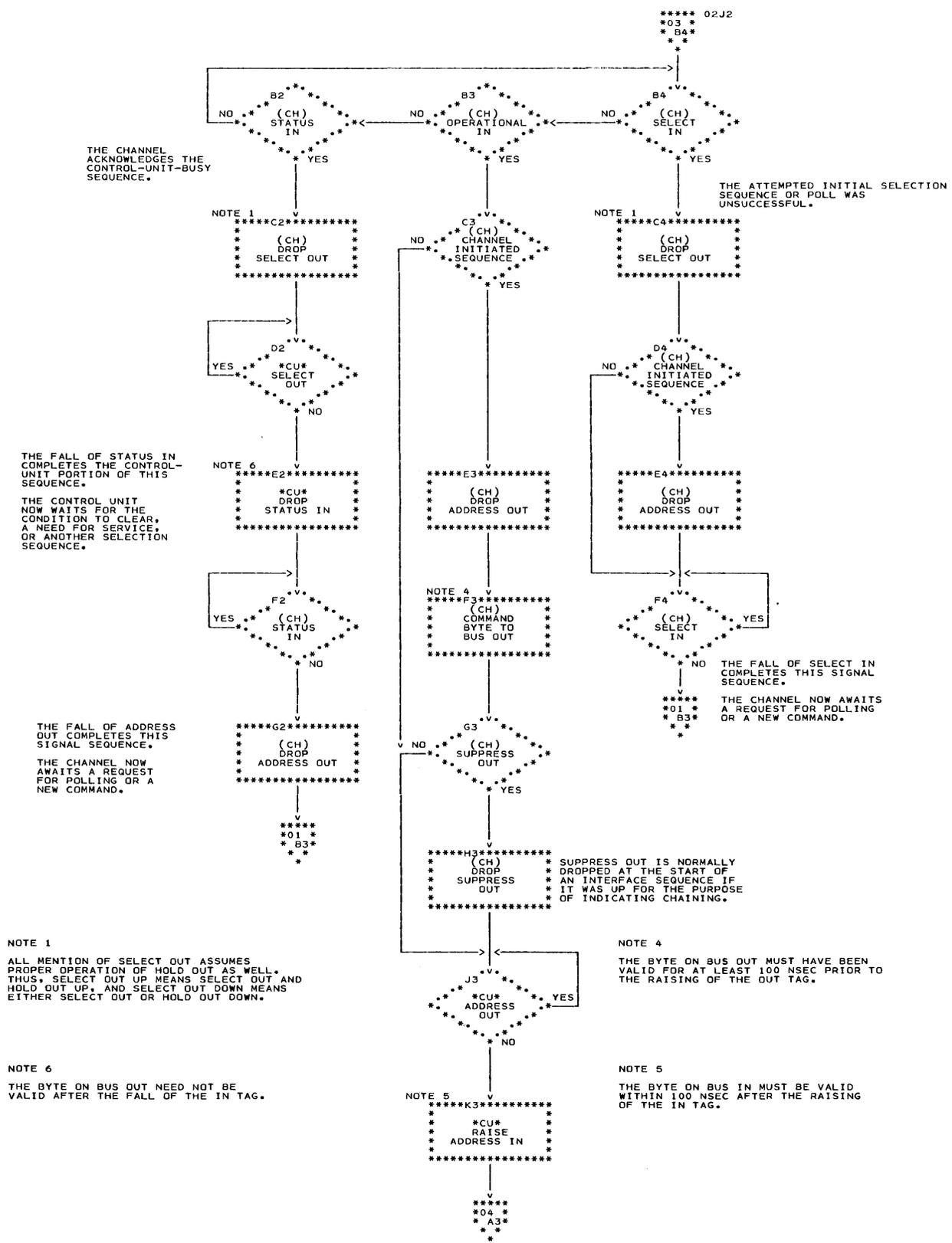
THE PRESENCE OF CH (CHANNEL) OR CU (CONTROL UNIT) IN A BLOCK, SPECIFIES RESPONSIBILITY FOR THE ACTION TAKEN OR DECISION MADE.



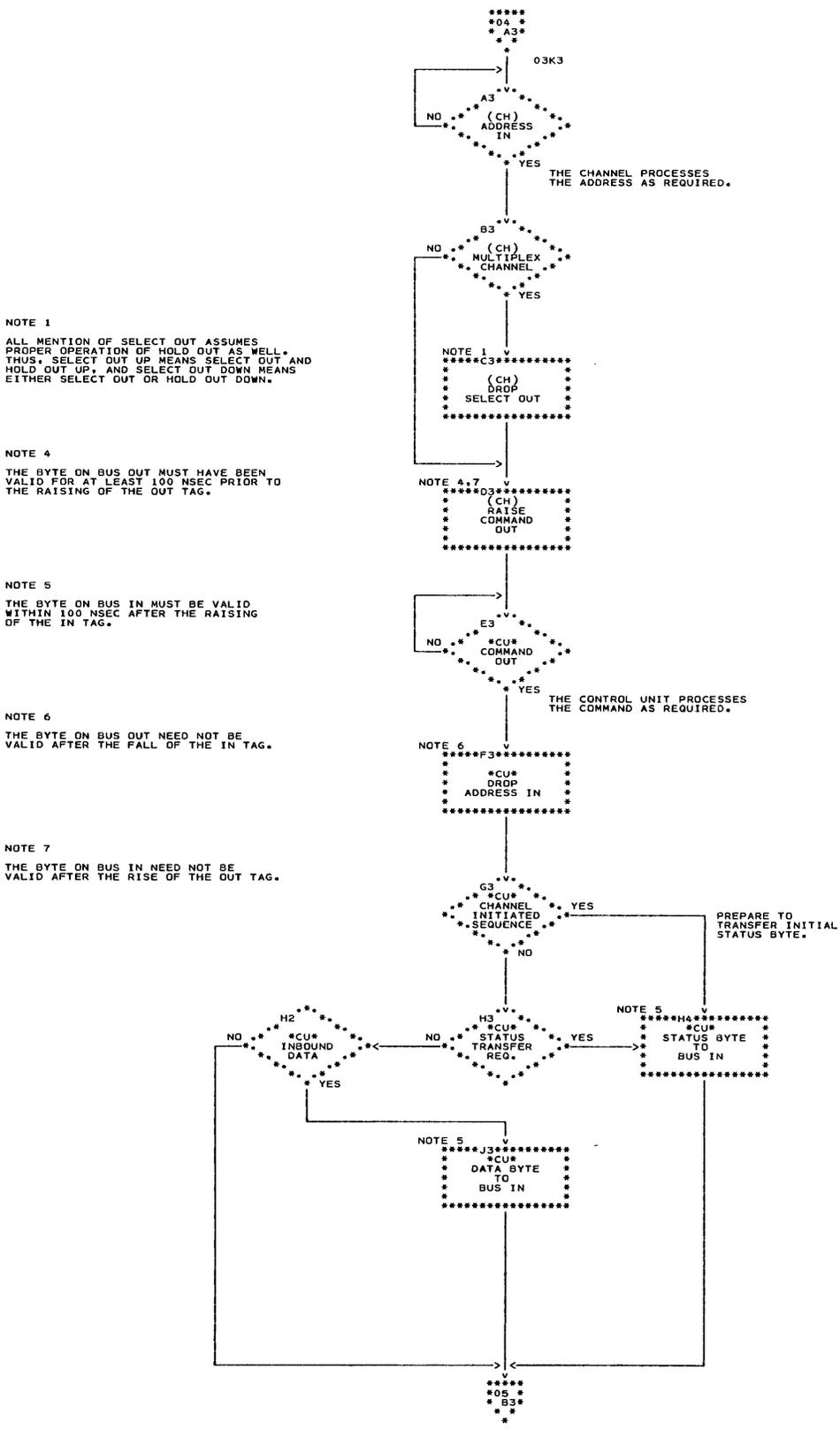
NOTE 1
ALL MENTION OF SELECT OUT ASSUMES PROPER OPERATION OF HOLD OUT AS WELL. THUS, SELECT OUT UP MEANS SELECT OUT AND HOLD OUT UP; AND SELECT OUT DOWN MEANS EITHER SELECT OUT OR HOLD OUT DOWN.

THE START-POLL DECISION IS USUALLY MADE ON THE BASIS OF REQUEST IN BEING UP.
REQUEST IN IS RAISED WHEN A CONTROL UNIT HAS A NEED FOR SERVICING AND IS DROPPED WHEN THE NEED IS SERVICED.
THE CHANNEL NORMALLY SERVICES ALL REQUESTS BEFORE INITIATING A SELECTION SEQUENCE.

Flow Diagram 1. Initiation of Polling or Selection



Flow Diagram 2. Control Unit Response to Select Out (Part 2 of 2)



NOTE 1
ALL MENTION OF SELECT OUT ASSUMES
PROPER OPERATION OF HOLD OUT AS WELL.
THUS, SELECT OUT UP MEANS SELECT OUT AND
HOLD OUT UP, AND SELECT OUT DOWN MEANS
EITHER SELECT OUT OR HOLD OUT DOWN.

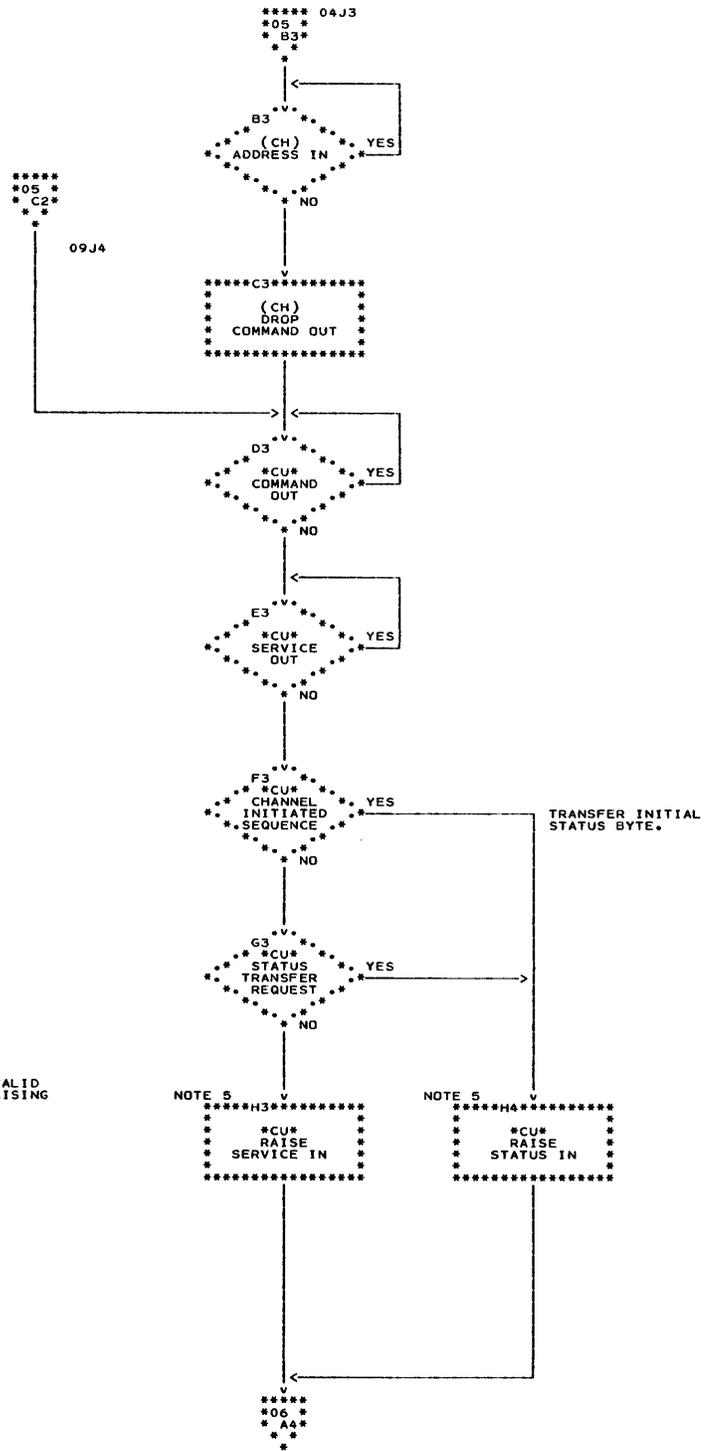
NOTE 4
THE BYTE ON BUS OUT MUST HAVE BEEN
VALID FOR AT LEAST 100 NSEC PRIOR TO
THE RAISING OF THE OUT TAG.

NOTE 5
THE BYTE ON BUS IN MUST BE VALID
WITHIN 100 NSEC AFTER THE RAISING
OF THE IN TAG.

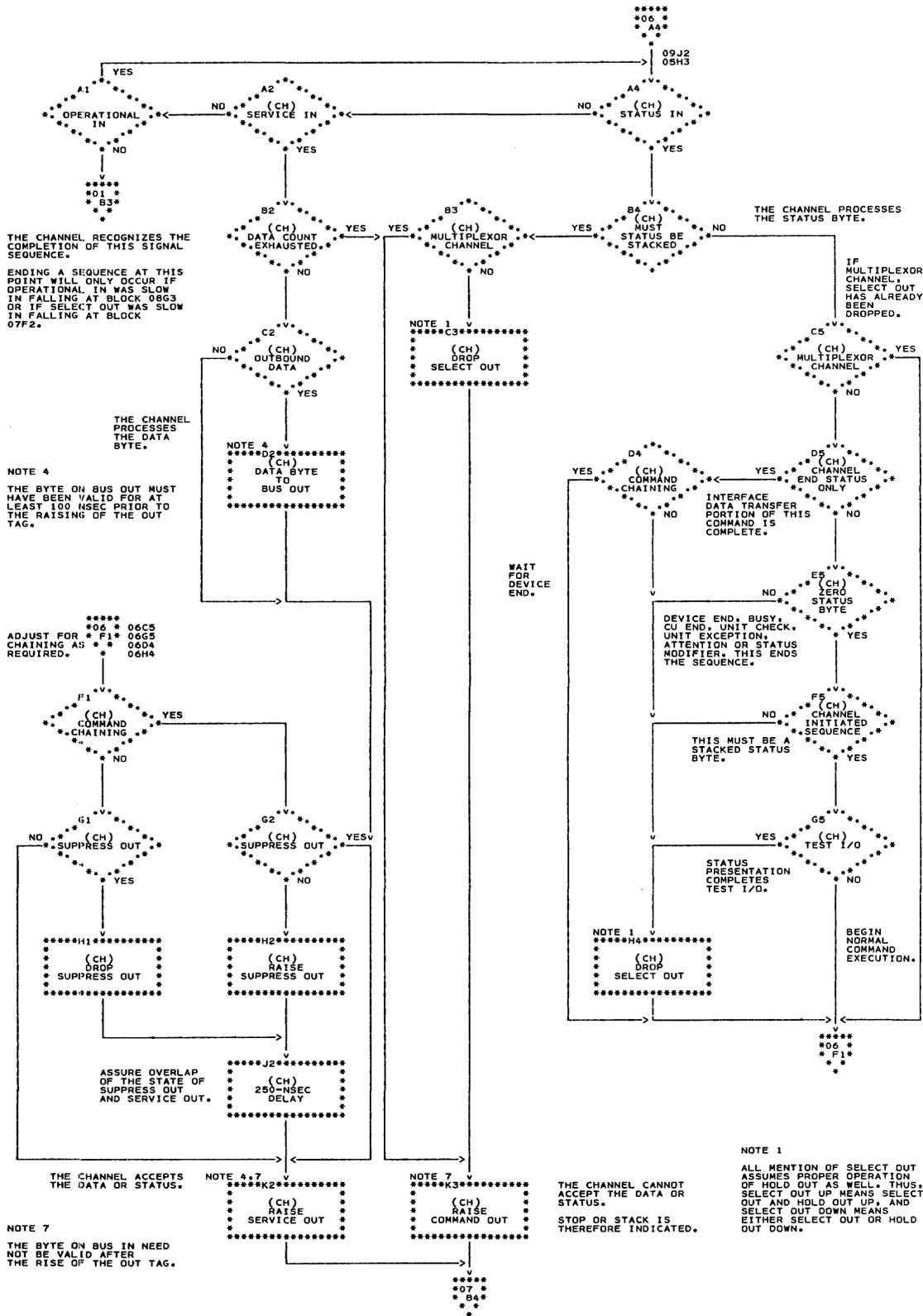
NOTE 6
THE BYTE ON BUS OUT NEED NOT BE
VALID AFTER THE FALL OF THE IN TAG.

NOTE 7
THE BYTE ON BUS IN NEED NOT BE
VALID AFTER THE RISE OF THE OUT TAG.

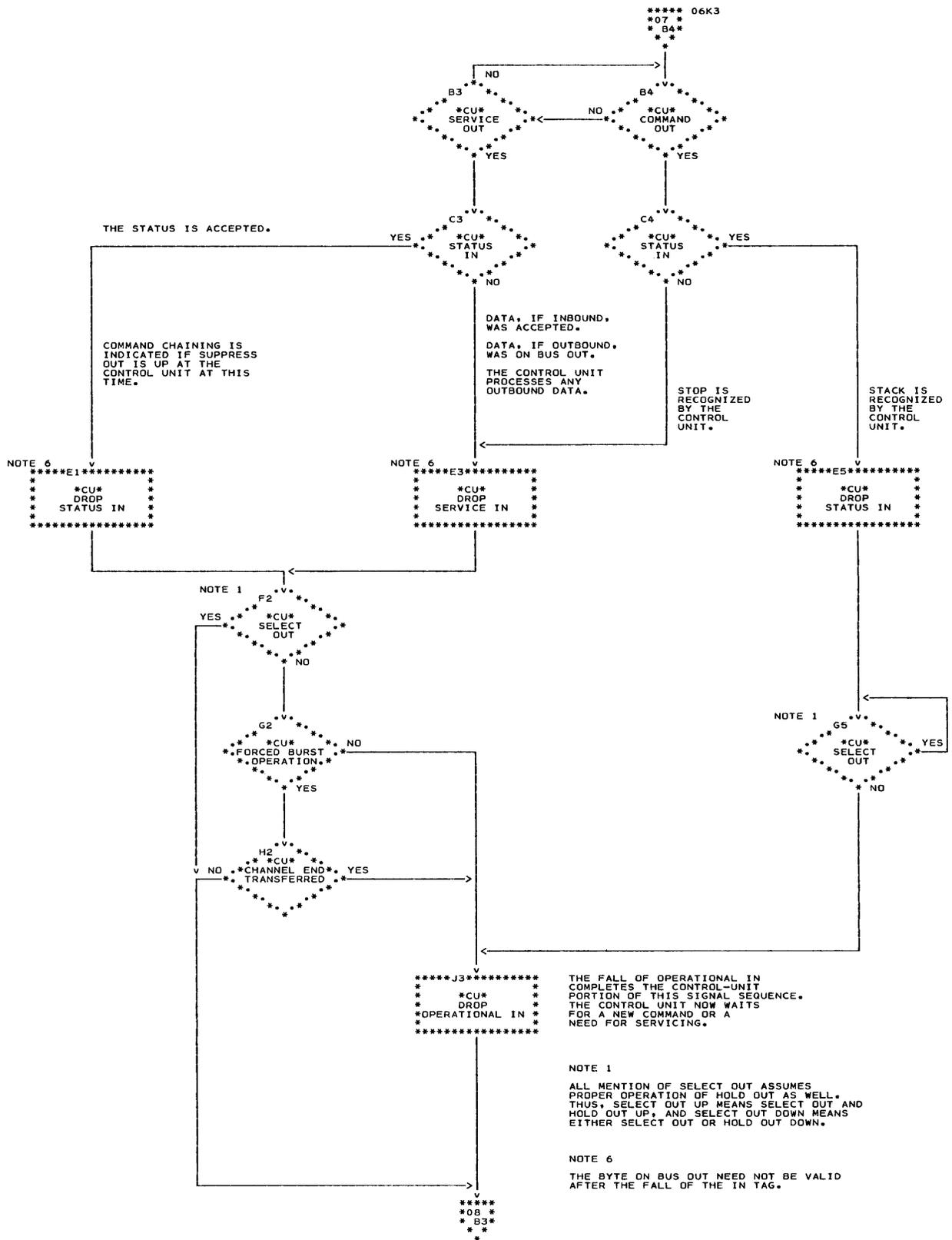
Flow Diagram 3. Command Transfer



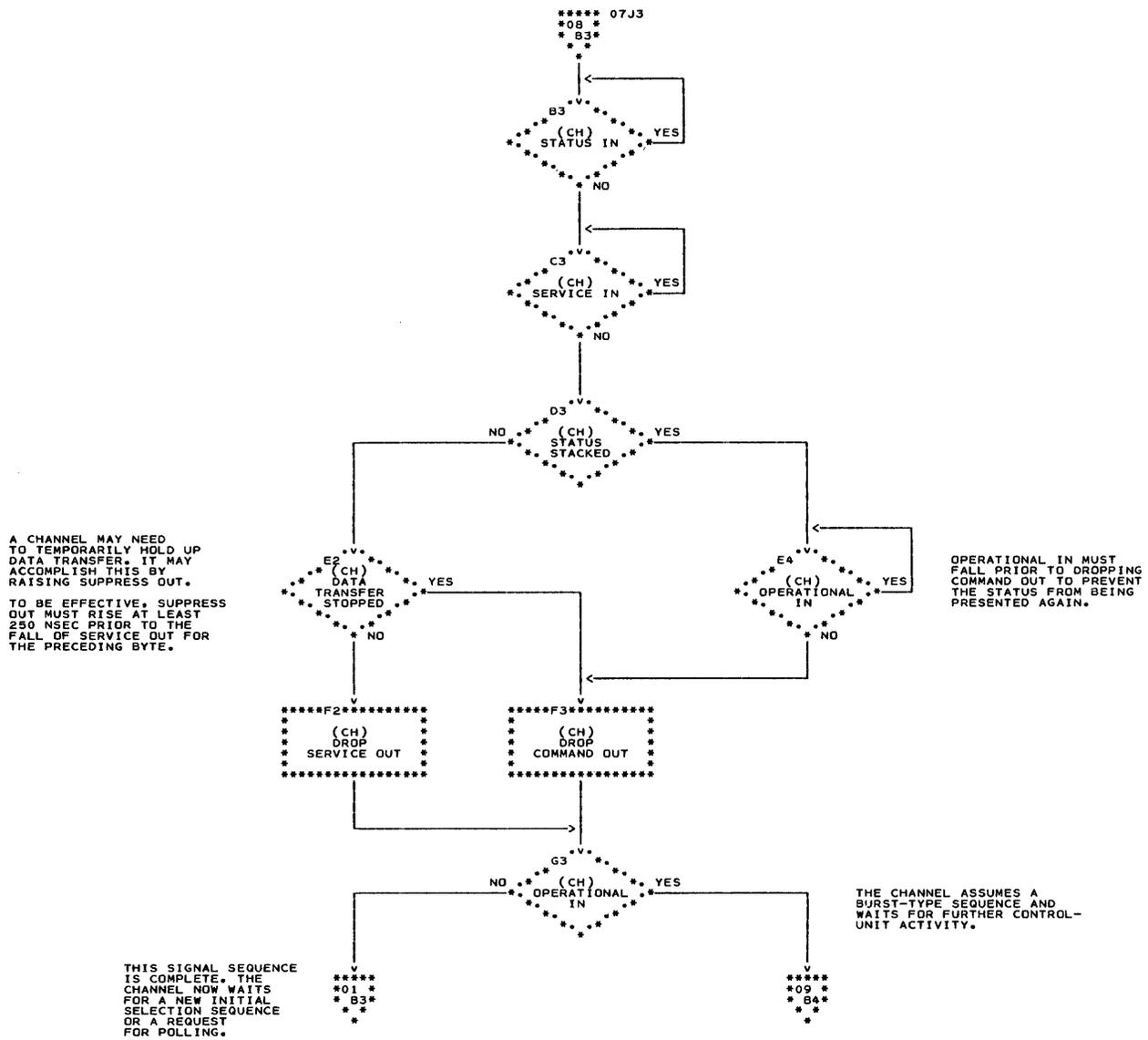
Flow Diagram 4. Status/Data Presentation



Flow Diagram 5. Response to Status/Data Presentation

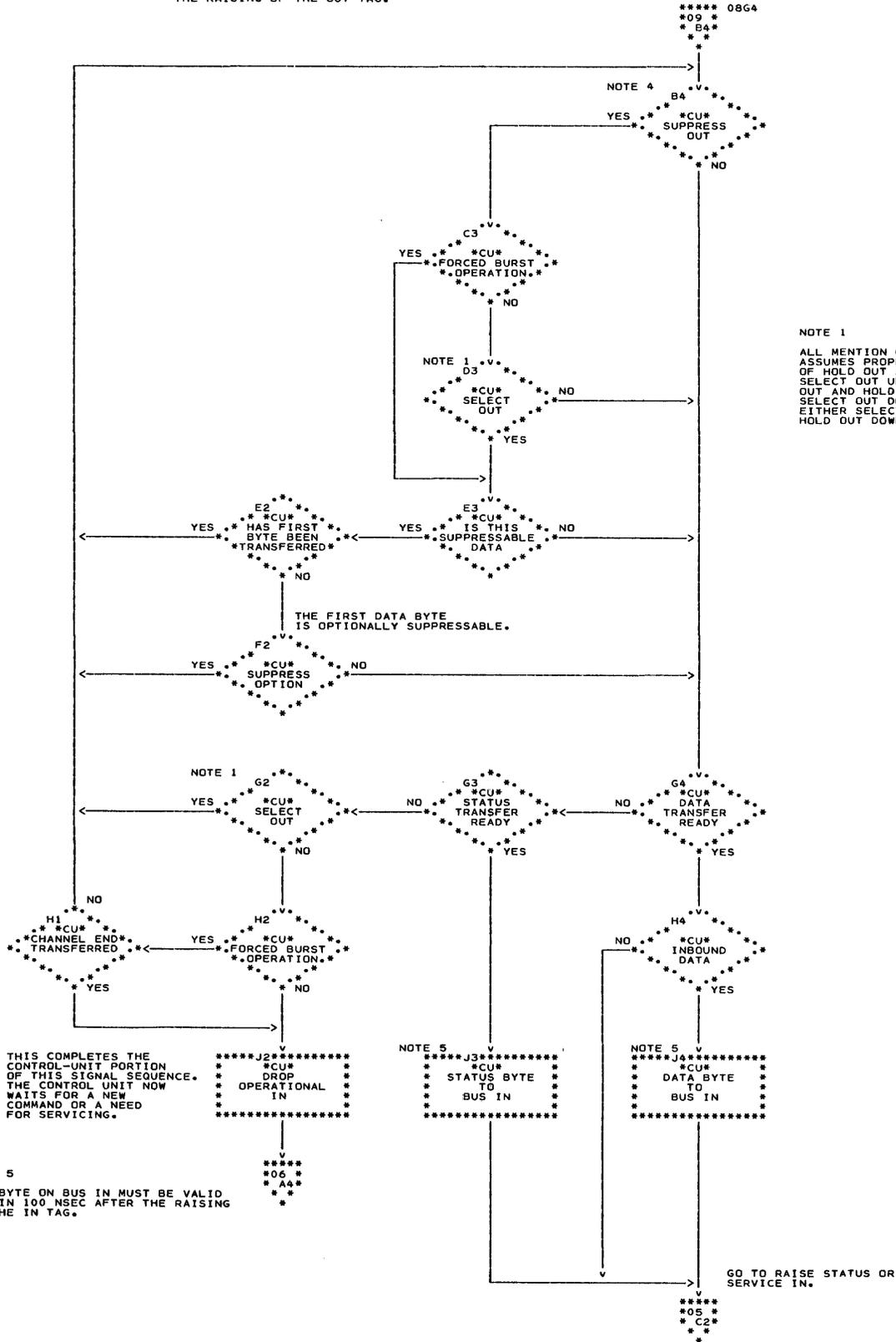


Flow Diagram 6. Response to Stack/Stop/Accept

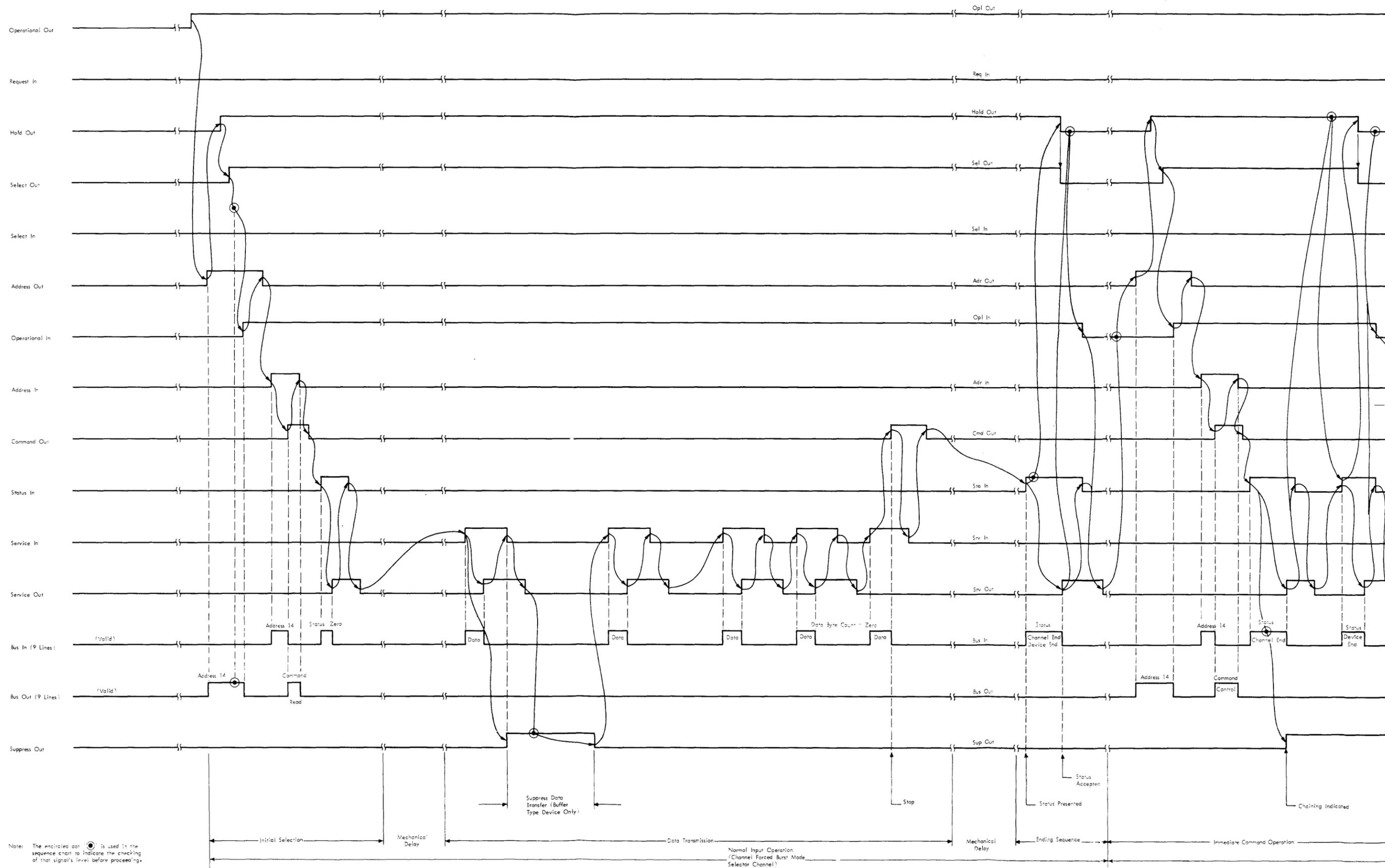


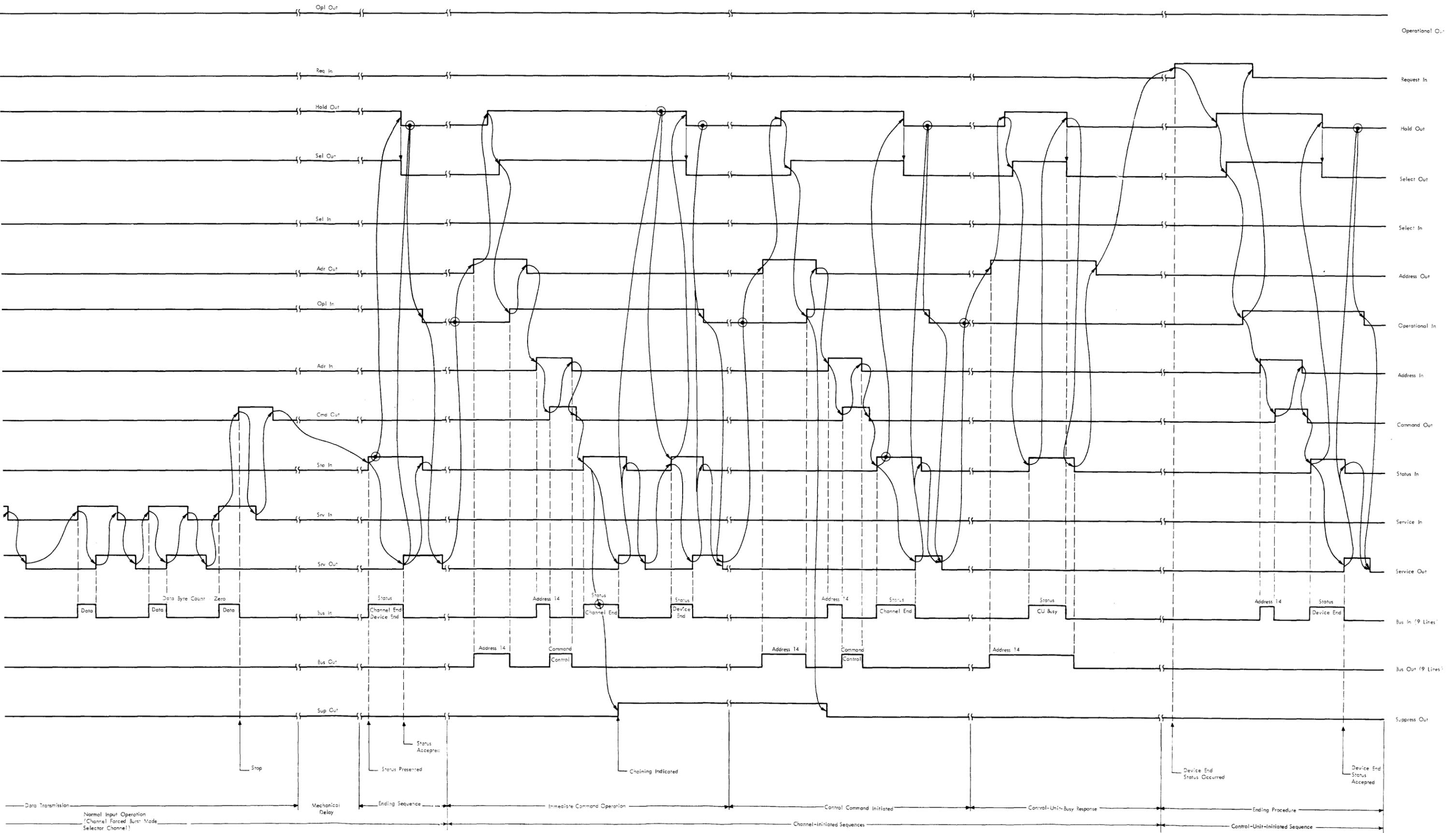
Flow Diagram 7. Response to Fall of Status/Service In

NOTE 4
 THE BYTE ON BUS OUT MUST HAVE BEEN VALID FOR AT LEAST 100 NSEC PRIOR TO THE RAISING OF THE OUT TAG.

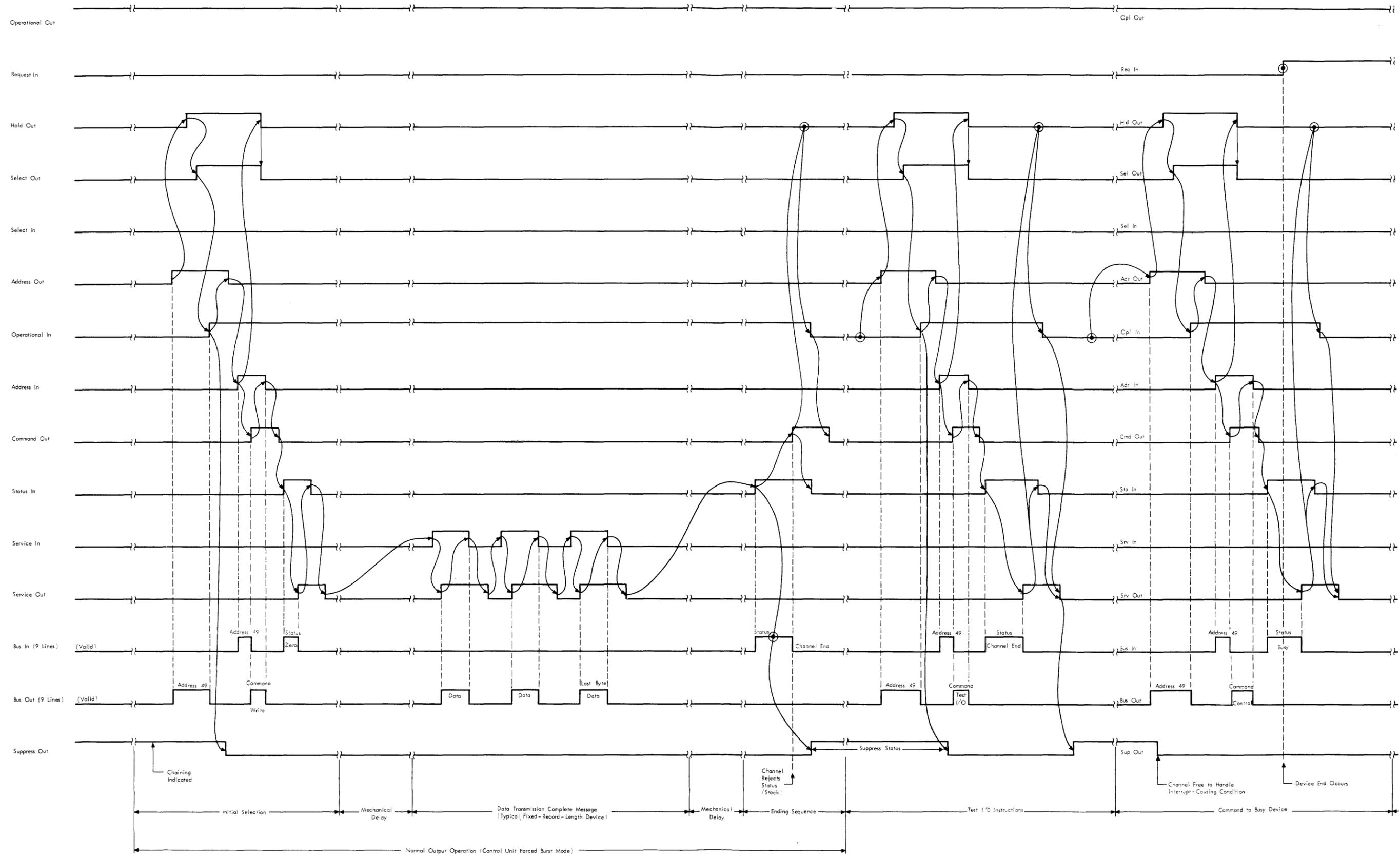


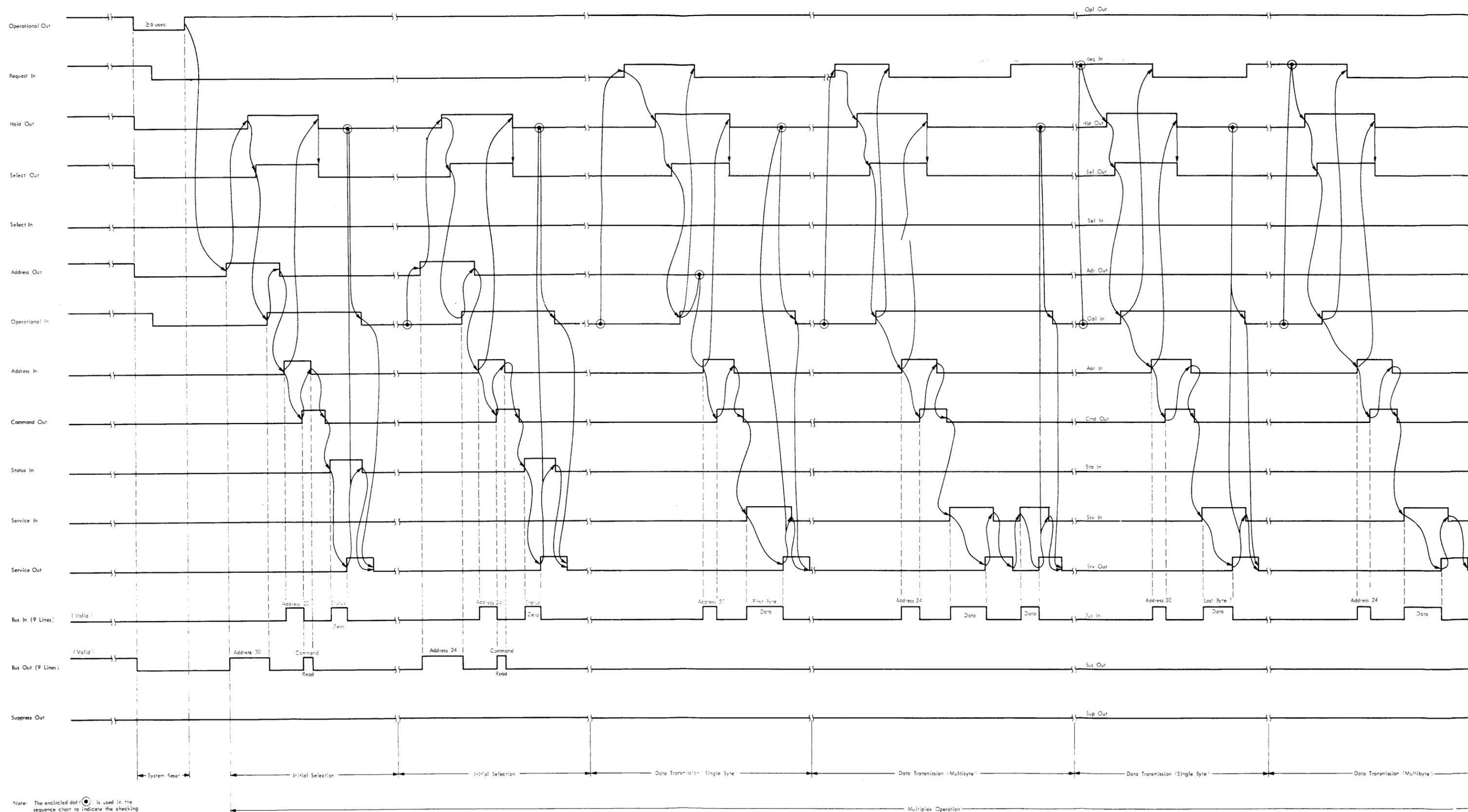
Flow Diagram 8. Burst Mode Waiting Loop





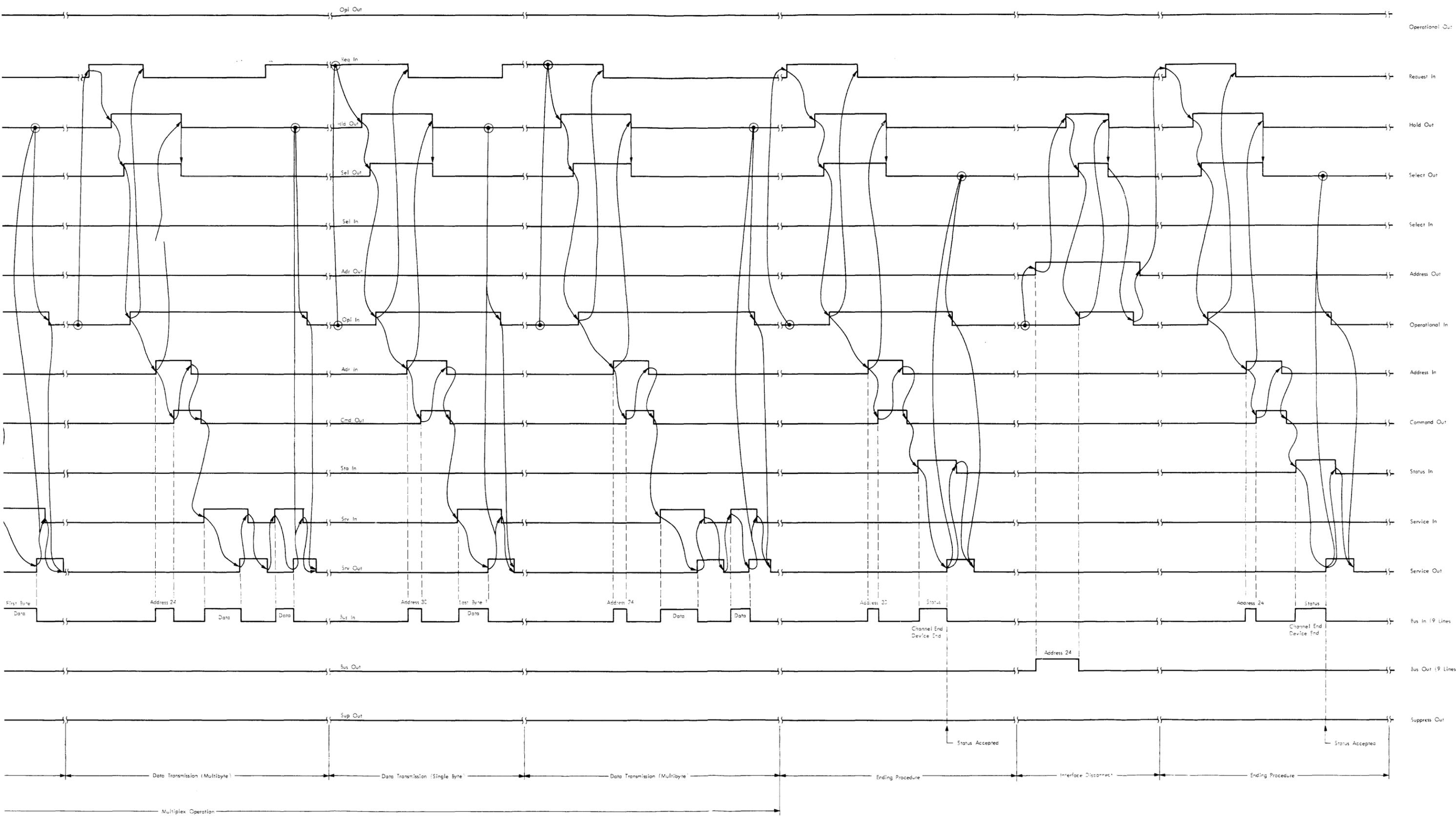
Sequence Chart 1. Selector Channel Operation
 Appendix C. Supplementary Application Information C-11





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

Multiplex Operation



Sequence Chart 3. Multiplexing Operation on Multiplexer Channel
 Appendix C. Supplementary Application Information C-15



International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]