Contents

Work Station Attachment

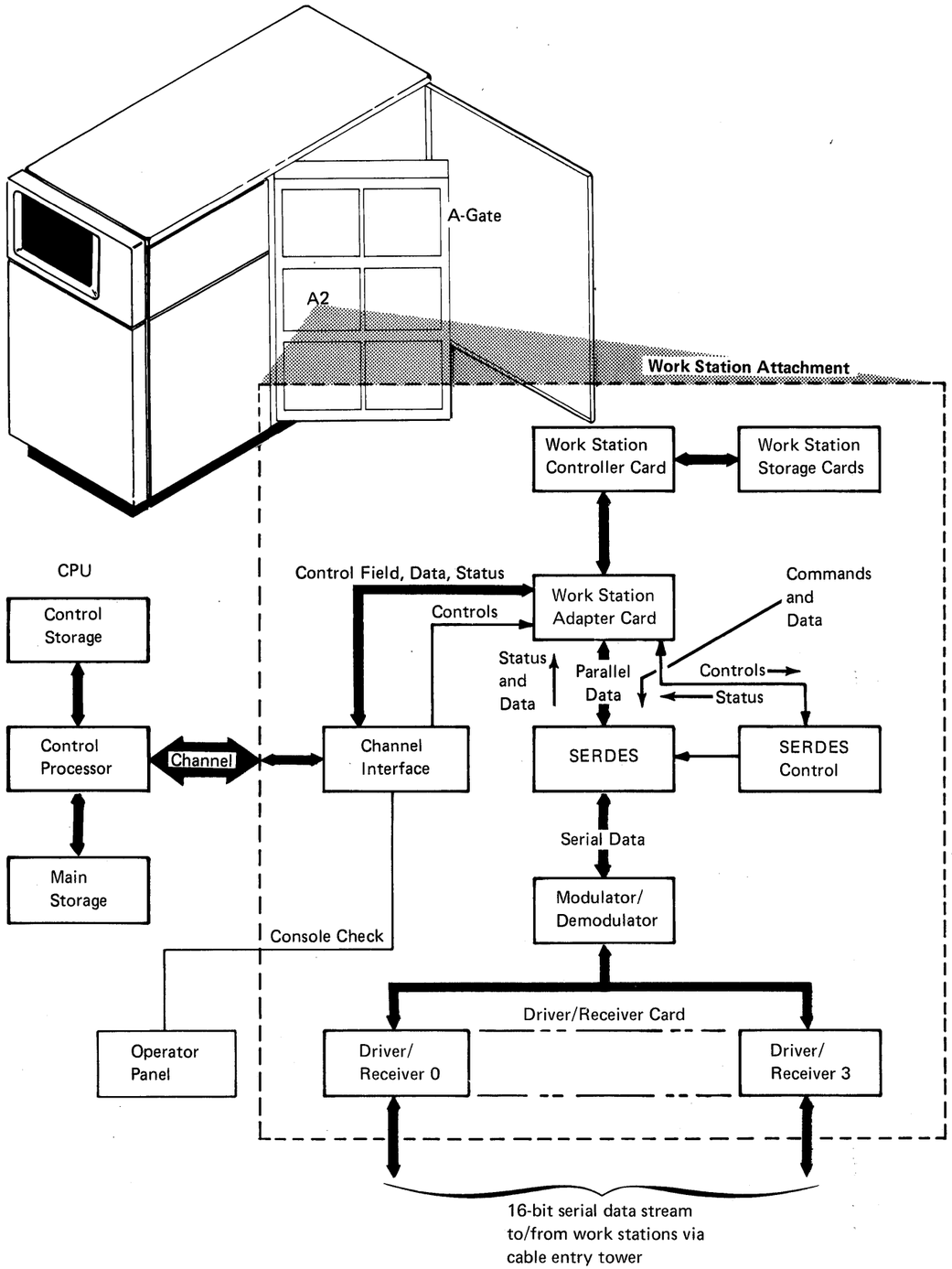
INTRODUCTION

In System/34, the work station attachment controls from one to a maximum of eight work stations. One of the work stations, a 5251 Display Station or a 5252 Dual Display Station, is the system console and must be attached to System/34 on port 0. Then, at the customer's option, up to seven additional work stations can be attached to ports 1 through 3. These additional work stations can be either 5251 or 5252 Display Stations, 5256 Printers, or a combination of them.

All work stations communicate with the attachment through a 16-bit serial data stream via twinaxial cables. For a description of the data stream, see *Serial Interface to Work Stations* later in this section.

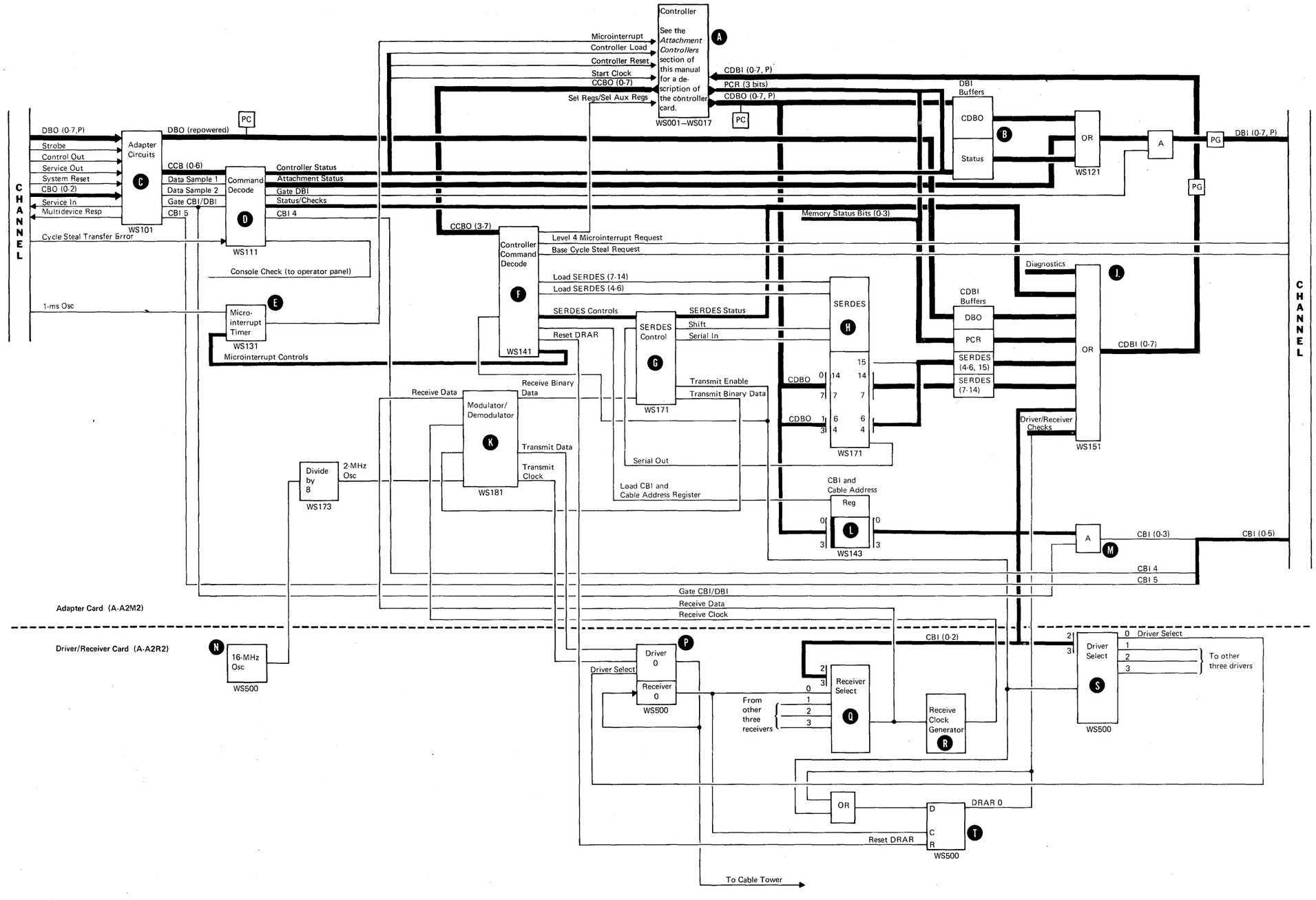
The work station attachment circuits are located on the A-gate, A2 board, and include the following:

- Controller card in A-A2N2
- Adapter card in A-A2M2
- Driver/receiver card in A-A2R2
- Storage cards in A-A2P4 and A-A2Q4
- Feature storage card in A-A2R4 (optional)
- Four top card connectors
- One cable from A-A2V2 to the twinaxial cables attached to the cable entry tower



DATA FLOW

The functional units identified with an alphabetic character on this data flow figure are described on the following three pages. For a logic description of the functional units, see the appropriate FSL page. For example, see FSL page WS500 for a logic description of the driver/receiver circuits.



FUNCTIONAL UNITS

The functional units shown in the data flow figure on the preceding page are described in the following paragraphs. The controller signals (A), the controller, and the storage signals are described in the *Attachment Controllers* section of this manual.

Two cards are shown in the data flow figure: the driver/receiver card (below the dashed line) and the adapter card (above the dashed line). The adapter card communicates with (1) the system through the channel, (2) the controller through the controller interface, and (3) the work stations through the serial interface.

Channel Interface

The channel communicates with the adapter card through an interface that includes the adapter circuits (C), the command decode (D), the DBI buffers and the DBI bus (B), and the CBI bus (M)

Adapter Circuits (C)

The adapter circuits:

- Synchronize the channel with the adapter card
- Repower DBO
- Generate:
  - A 7-bit channel command bus (CCB)
  - Control lines used by the command decode to gate the CCB
  - CBI bit 5 (bad DBO parity)

The adapter circuits let the channel communicate with the attachment through control lines and data buses. The 'control out', 'service out', and 'strobe' lines signal the adapter circuits when data is available on DBO or should be on DBI. The 'service in' and 'multidevice resp' lines signal the channel that the data on DBO has been received or that data is available on DBI.

Data bus out (DBO) is a 9-bit bus (0 through 7 and parity) that performs one of two functions; (1) it supplies an I/O output instruction, or (2) it supplies data to the attachment during service out of a cycle steal operation. On any I/O output instruction, DBO bits 4-7 contain the command modifier during control out. DBO bits 0-3 rely on the instruction type. For example, during control out of the sense interrupt level status byte (SILSB) command, DBO bits 0-3 contain the attachment interrupt level address (0100). During control out of any I/O instruction other than SILSB, DBO bits 0-3 contain the work station attachment device address (1100). During control out of a cycle steal operation, DBO bits 0-3 contain an address (0001) forced by the channel.

Command bus out (CBO) is a 3-bit bus that specifies the type of I/O command to be executed. For the work station attachment, these commands are:

```
000 =  Not used
001 =  SILSB
010 =  Base cycle steal
011 =  Jump on I/O condition
100 =  I/O load
101 =  I/O sense
110 =  I/O control load
111 =  Not used
```

Command Decode (D)

The command decode logic decodes the commands on the channel command bus (CCB), which is a 7-bit bus generated by the adapter circuits. Bits 0-2 contain CBO bits 0-2 and bits 3-6 contain the command modifier bits from DBO bits 4-7 during control out. See *Commands* later in this section for a description of the commands decoded by this logic.

Data Bus In (B)

Data bus in (DBI) is a 9-bit bus (0 through 7 and parity) that supplies data and status information to the channel. It contains:

- Data or sense information during I/O sense instructions
- Data during base cycle steal operations
- Sense information during SILSB instructions

The 'gate CBI/DBI' line from the adapter circuits activates the DBI bus, and CCB bits 5 and 6 determine what information is gated on the DBI bus. That information is:

- Controller DBO during a sense controller DBO instruction or a base cycle steal operation
- Controller status latches during a sense controller status instruction
- Attachment status and check conditions during sense check or sense attachment status instructions

Command Bus In (M)

Command bus in (CBI) is a 6-bit bus that supplies control information to the channel; it is activated during I/O instructions or during base cycle steal operations by the 'gate CBI/DBI' line from the adapter circuits.

CBI bits 0-3 supply control information to the channel during a base cycle steal operation through the following sequence of events:

1. A controller instruction, decoded in the controller command decode logic (F), generates the 'base cycle steal request' signal to the channel.
2. The channel responds on command bus out (CBO) with a base cycle steal (010).
3. A cycle steal operation is then executed as instructed by CBI bits 0-4.

CBI Bits 0 1 2 3 4	Function
0 0	Base cycle steal sense (WR not incremented)
0 1	Base cycle steal load (WR not incremented)
1 0	Base cycle steal sense (WR incremented)
1 1	Base cycle steal load (WR incremented)
0 0	Base cycle steal WR4 on interrupt level 4
0 1	Base cycle steal WR5 on interrupt level 4
1 0	Base cycle steal WR6 on interrupt level 4
1 1	Base cycle steal WR7 on interrupt level 4
1	Control storage
0	Main storage

CBI bit 4, when active, has three functions:

- It specifies main storage or control storage during base cycle steal operations.
- It signals the channel that DBI parity may not be valid.
- It signals the channel that a condition tested for by a jump on I/O condition instruction is present.

CBI bit 5 is active when DBO even parity is sensed.

Controller Interface

In addition to the controller signals (A), the controller interface includes the 16-MHz oscillator (N), the controller command decode (F), the microinterrupt timer (E), and the controller DBI buffers and controller DBI bus (J).

16-MHz Oscillator (N)

The 16-MHz oscillator is located on the work station driver/receiver card. It supplies a free-running clock to the work station controller card and to the 5211 Printer controller card (if the system has a 5211 Printer).

Controller Command Decode (F)

The controller command decode logic decodes the controller instructions from bits 3-7 of the controller CBO (CCBO) bus. It also generates the 'level 4 microinterrupt request', 'base cycle steal request', and the SERDES control lines.

On input instructions, controller CBO bits 4-7 gate information on the controller DBI (CDBI) bus. On output instructions, the decode of the controller CBO bus causes the data byte on controller DBO (CDBO) to be gated into the DBI buffers (B), the SERDES (H), or the CBI and cable address register (L). The decode of the controller CBO bus is as follows.

Controller CBO Bits 3 4 5 6 7	Function	Comments
0 0 0 0 0	Diagnostic load	
0 0 0 0 1	Set level 4 microinterrupt request	
0 0 0 1 0	Load SERDES	Bits 7-14
0 0 0 1 1	Load SERDES	Forces bits 0-2 to 000, initializes the parity bit (bit 3), loads bits 4-6, and forces the sync bit (bit 15) to a 1.
0 0 1 0 0	Diagnostic SERDES clock and set controller DBO parity check	
0 0 1 0 1	Reset base cycle steal transfer error	
0 0 1 1 0	Set base cycle steal request	
0 0 1 1 1	Reset command pending	
0 1 0 0 0	Load CBI and cable address register	Loaded with DBO bits 0-3
0 1 0 0 1	Reset op check	
0 1 0 1 0	Reset base cycle steal control storage	Sets base cycle steal main storage
0 1 0 1 1	Reset service required	
0 1 1 0 0	Set controller busy	
0 1 1 0 1	Reset controller DBO/DBI parity check	
0 1 1 1 0	Reset serial transfer check	
0 1 1 1 1	Reset transmit enable	
1 0 0 0 0	Set transmit enable	
1 0 0 0 1	Select aux regs	
1 0 0 1 0	Select regs	Default case
1 0 0 1 1	Reset controller busy	
1 0 1 0 0	Reset receive enable	
1 0 1 0 1	Set base cycle steal control storage	Resets base cycle steal main storage
1 0 1 1 0	Load DBI buffer	Loaded with controller DBO bits 0-7
1 0 1 1 1	Set op check	
1 1 0 0 0	Enable 7-second time-out	
1 1 0 0 1	Set receive enable	
1 1 0 1 0	Reset driver/receiver activity reg	
1 1 0 1 1	Disable 7-second time-out	Default case
1 1 1 0 0	Enable controller microinterrupt	
1 1 1 0 1	Load microinterrupt timer	Loaded with controller DBO bits 0-4
1 1 1 1 0	Reset 7-second time-out	
1 1 1 1 1	Disable controller microinterrupt	Default case
X 0 0 0 0	Sense SERDES	Bits 7-14
X 0 0 0 1	Sense SERDES	Bits 0-6 and 15
X 0 0 1 0	Sense checks	
X 0 0 1 1	Sense driver/receiver activity reg	Bits 0-3
X 1 1 0 0	Sense diagnostic adapter status	
X 1 1 0 1	Sense 'not go' latch	
X 1 1 1 0	Sense PCR	From controller DBI buffer
X 1 1 1 1	Sense channel DBO	In controller DBI buffer

See *Controller DBI* for descriptions of the controller DBI bits on a sense command.

Microinterrupt Timer (E)

The microinterrupt timer is a 5-bit counter (counts down) that is loaded with a value from the controller DBO (CDBO) bits (0-4). It is decreased by a value of 1 every millisecond by a free-running, one-millisecond oscillator from the channel. When the timer has counted down to a value of 0 and if the 'enable microinterrupt' latch in the command decode logic is set, the 'microinterrupt' signal to the controller goes active.

It is typical to have the timer set to some value such as 30, which causes the 'microinterrupt' signal to go active every 30 milliseconds. This signal forces the controller program to some routine, such as a poll, which in turn checks each keyboard for operator keystrokes.

Controller DBI (J)

Controller DBI (CDBI) is a 9-bit bus (0 through 7 and parity) that supplies data and status information to the controller during a controller input instruction. The following table lists all of the data and status sent to the controller on controller DBI.

I/O Sense Command	Controller DBI Bits							
	0	1	2	3	4	5	6	7
Sense shift register (hex 00)	7	8	9	10	11	12	13	14
	SERDES Bits							
Sense shift register (hex 01)	Set to 1	4	5	6	$\overline{3}$	15	Transmit active	Not end of message
		SERDES Bits						
Sense checks (hex 02)	Cycle steal request	Controller DBO/DBI parity check	Cycle steal transfer error	Interrupt level 4 request	Serial parity check	Service required	Command pending	Set to 1
Sense driver/ receiver activity register (hex 03)	0	1	2	3	4	5	6	7
	Driver/Receiver Activity Register							
Sense diagnostics (hex 0C)	Select primary registers	Sample receive disable	Cycle steal main storage	Delayed transmit disable	Cable address bit 0	Cable address bit 1	Cable address bit 2	Cable address bit 3
Sense 'go' latch (hex 0D)	These 3 bits are set to 0.			Not go	These 3 bits are set to 0.			Not go
Sense PCR and memory (hex 0E)	0	1	2	3	Nonzero	Carry	Zero	Controller DBO/DBI parity check
	Memory Status Bits				PCR Bits			
Sense DBO (hex 0F)	0	1	2	3	4	5	6	7
	DBO Bits 0-7							

Serial Interface

System/34 communicates with the work stations through the work station serial interface. This interface includes the SERDES control (G), the SERDES (H), the modulator/demodulator (K), the CBI and cable address register (L), and the driver/receiver circuits (P), (Q), (R), (S), and (T).

SERDES Control (G)

The SERDES control signals from the controller command decode logic control the serial interface latches (located in SERDES control) by setting or resetting the 'transmit enable' and 'receive enable' latches. The SERDES control also contains a 'go' latch that synchronizes the controller with the serial interface. The 'go' latch is one of the bits of SERDES status gated to the controller on controller DBI (J).

The SERDES control generates even parity on transmit operations and checks for even parity on receive operations. It also controls the SERDES shifting, reset, and gating of the SERDES parallel outputs to controller DBI on receive operations.

SERDES (H)

The serializer/deserializer is a 16-bit shift register that is loaded with parallel data. It also has parallel outputs that are put into the controller DBI buffers when 16 bits have been shifted into the SERDES on a receive operation. Also on a receive operation, the 'serial in' data to SERDES is generated from the 'receive binary data' line by the SERDES control.

On transmit operations, the SERDES is loaded in parallel with controller DBO by a controller output instruction and the 'serial out' data to SERDES control becomes 'transmit binary data'.

The unit of information transmitted or received between the work stations and the attachment is a 16-bit serial frame. This frame is described under *Serial Interface to Work Stations* later in this section.

CBI and Cable Address Register (L)

In addition to being used for base cycle steal operations, this 4-bit register uses bits 2 and 3 to select one of four driver/receivers for transmit or receive operations.

Driver/Receiver Circuits (P), (Q), (R), (S), *and* (T)

There are four driver/receivers (P) on the driver/receiver card: one for each twinaxial cable attached to the cable entry tower.

The outputs of the receiver select logic (Q) are always active, and they continuously represent the data on the cable selected by the cable address register. On receive operations, the following sequence occurs:

1. The 'receive data' output from the receiver select logic feeds the modulator/demodulator (K) and the receive clock generator (R).
2. The 'receive clock' signal from the receive clock generator (R) is sent to the modulator/demodulator.
3. The modulator/demodulator generates 'receive binary data' and sends it to the SERDES control logic (G).

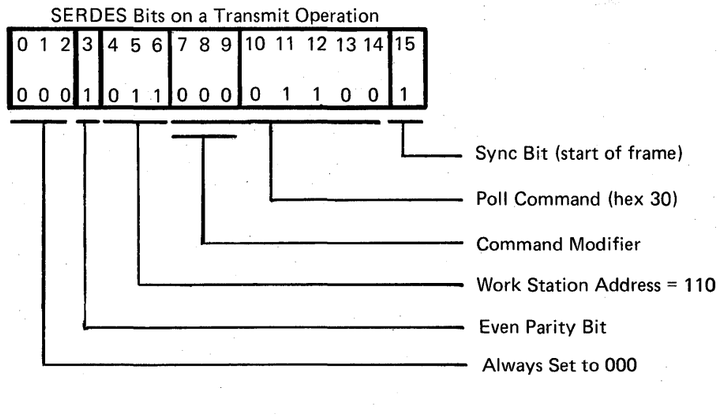
The driver select logic (S) outputs, used on transmit operations, are gated by the 'transmit enable' line from the SERDES control. Then, the driver selected by the cable address register uses the 'transmit clock' and 'transmit data' lines from the modulator/demodulator to transmit serial data to the work station.

There is a clocked latch, called a driver/receiver activity register (DRAR) (T), for each driver/receiver (P). The latch is clocked by a transition on the receiver output. The data input to each latch comes from one of two sources: either the output of the latch itself, or the 'transmit enable' line from the SERDES control (G). All driver/receiver activity registers are reset at the start of transmit operations (polls) to the work stations. The outputs of these registers are gated to CDBI for sensing.

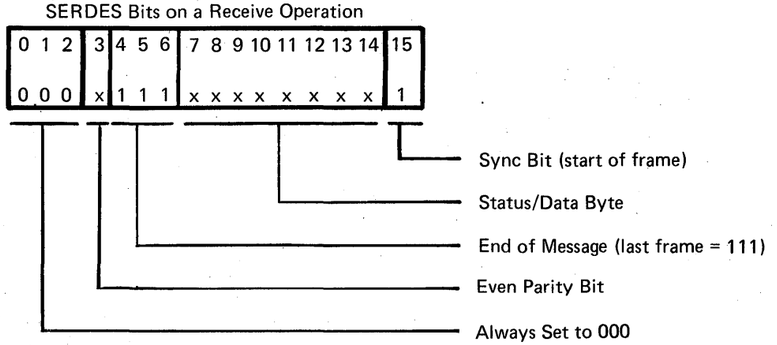
SERIAL INTERFACE TO WORK STATIONS

System/34 communicates with the work stations through a data stream frame. A frame, as shown in the following figure (for the last frame of typical transmit and receive operations) and described below, is 16 bits long. A frame is serially transmitted to, or received from, the work stations.

SERDES Bits on a Transmit Operation



SERDES Bits on a Receive Operation



Note: Bits 4, 5, and 6 contain the work station address if the frame is not the last frame received from a work station.

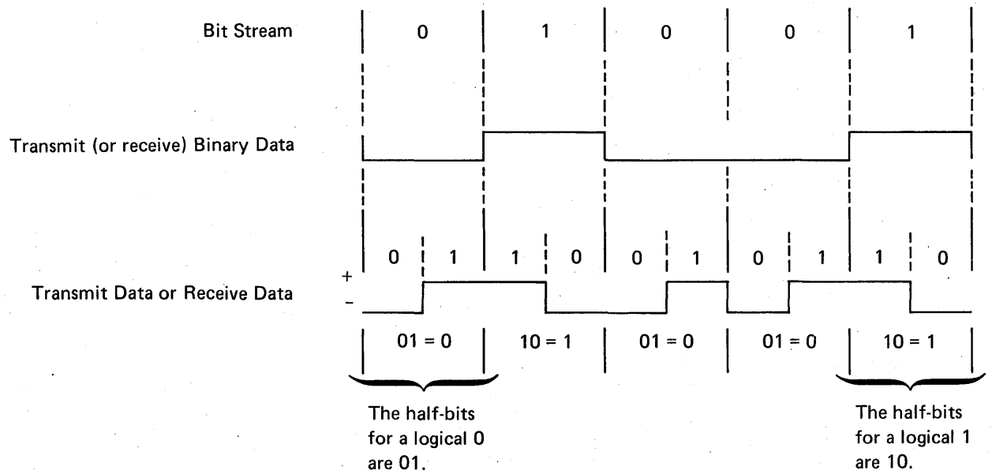
SERDES Bits	Description
15	Bit 15 is the sync bit and the first bit of the frame. It is always active on transmit and receive operations. After the SERDES control logic senses the sync bit during a receive operation, it counts the next 15 bits going into SERDES. Then, the pulse to load the SERDES into the controller DBI buffers is generated.
14-7	These bits represent a data byte or a command byte on transmit operations and a data byte or a status byte on receive operations.
6-4	These bits represent the work station address (000 through 110) on transmit operations or receive operations. They also represent an end of message (111) for the last frame on transmit and receive operations.
3	This is an even parity bit that is generated on bits 3 through 15 on transmit and receive operations.
2-0	These bits are always set to 000 on transmit and receive operations.

Data Transmissions

The data transmitted on the serial interface is described in the following paragraphs. *Phase Encoded Data, Predistorted and Distorted Data,* and *Differential Data* describe how the signal is generated. *Bit Synchronization* and *Frame Synchronization* describe the information transmitted before the first transmitted frame, and *Line Turnaround* describes the time between receive mode and transmit mode. Read all of this information before reading *Operations* later in this section.

Phase Encoded Data

The 'receive binary data' signal into the SERDES control (G in the *Data Flow* figure) and the 'transmit binary data' signal out of the SERDES control both have a frequency of 1 megahertz (1 microsecond per bit time). Each bit time contains phase encoded data on the 'transmit data' or 'receive data' signal, as shown in the following figure. These two signals have each bit time divided into 2 half-bits to supply data integrity and noise detection on the serially transmitted data between System/34 and the work stations.



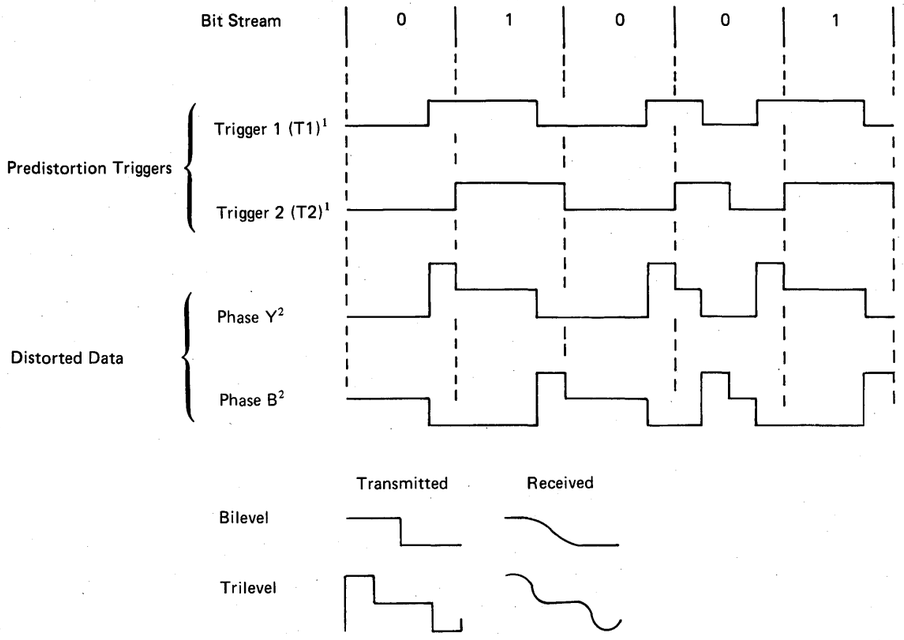
The 2 half-bits have a binary configuration of 10 for a logical 1, and a binary configuration of 01 for a logical 0. Binary configurations of 00 and 11 specify code violations. These code violations are used for the frame synchronization bits. When transmitting a logical 0 followed by a logical 1 (or a logical 1 followed by a logical 0), 2 consecutive half-bits must be at the same voltage level; that is, they must be plus or minus.

Predistorted and Distorted Data

Each driver (P in the *Data Flow* figure) contains two triggers that apply a predistortion method to the 'transmit data' signal. This method supplies increased integrity on the serially transmitted data by generating trilevel voltage signals for phase Y and phase B (the two signal conductors of the twinaxial cable).

The generation of the trilevel signals is shown in the following figure. The bottom part of the figure compares transmitted and received bilevel signals to transmitted and received trilevel signals, respectively. Notice that by adding small pulses (predistortion/distortion) to the trilevel signal, the rise and fall times of the signal are greatly improved.

The transmission of the trilevel signals produces the four-level differential data because the work station hardware responds to the differential current flow between phase Y and phase B. The four-level differential data is described in the next paragraph.



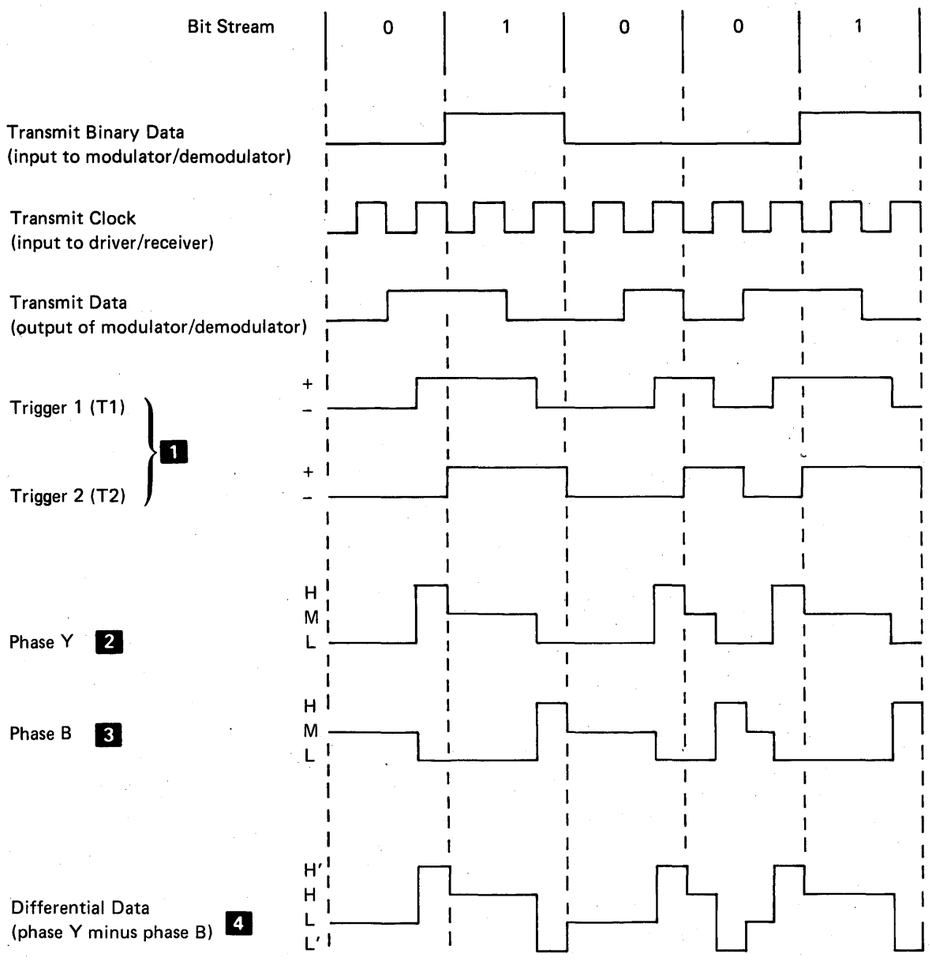
[1] The two triggers are in the driver/receiver.
[2] Phase Y and Phase B are the two signals transmitted on the twinaxial cable.

Differential Data

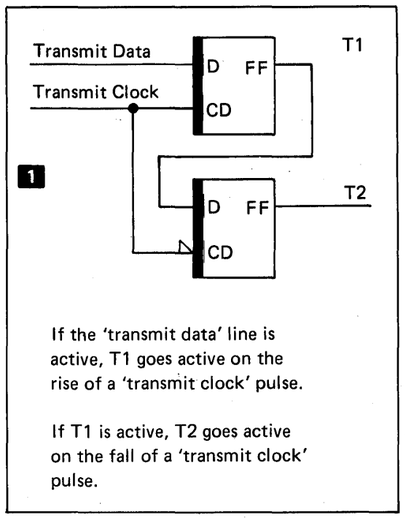
The differential data includes phase Y and phase B data as transmitted on the twinaxial cable from System/34 to a work station, or from a work station to System/34.

When data is received from a work station, the receiver (P in the *Data Flow* figure) subtracts the 'phase Y' signal from the 'phase B' signal to generate the four-level differential data signal. The receiver select logic (Q in the *Data Flow* figure) then supplies the 'receive data' signal to the modulator/demodulator (K in the *Data Flow* figure).

The following figure shows the relationship of the differential data signal to the signals that generate it. 2, 3, and 4 in the figure show how phase Y, phase B, and differential data are generated, respectively.



Note: H means high, M means medium, and L means low.



If the 'transmit data' line is active, T1 goes active on the rise of a 'transmit clock' pulse.

If T1 is active, T2 goes active on the fall of a 'transmit clock' pulse.

2 Phase Y is:
H if T1 is + and T2 is –
M if T1 and T2 are both +
L if T1 and T2 are both –, or
if T1 is – and T2 is +

3 Phase B is:
H if T1 is – and T2 is +
M if T1 and T2 are both –
L if T1 is +

4 Differential data is:
H' if phase Y is H and phase B is L
H if phase Y is M and phase B is L
L if phase Y is L and phase B is M
L' if phase Y is L and phase B is H

Bit Synchronization

When commands or data are transmitted to a work station, the first frame in the bit stream is preceded by the bit synchronization bits and the frame synchronization bits. The bit synchronization bits contain five logical 1-bits (binary 1010101010, as described under *Phase Encoded Data*) that inform the work station to check for the frame synchronization bits.

Frame Synchronization

The frame synchronization bits follow the bit synchronization bits and precede the sync bit of the first transmitted frame. The frame synchronization bits include 3 half-bit 1's and 3 half-bit 0's. Therefore, in binary, the frame synchronization bits are 111000.

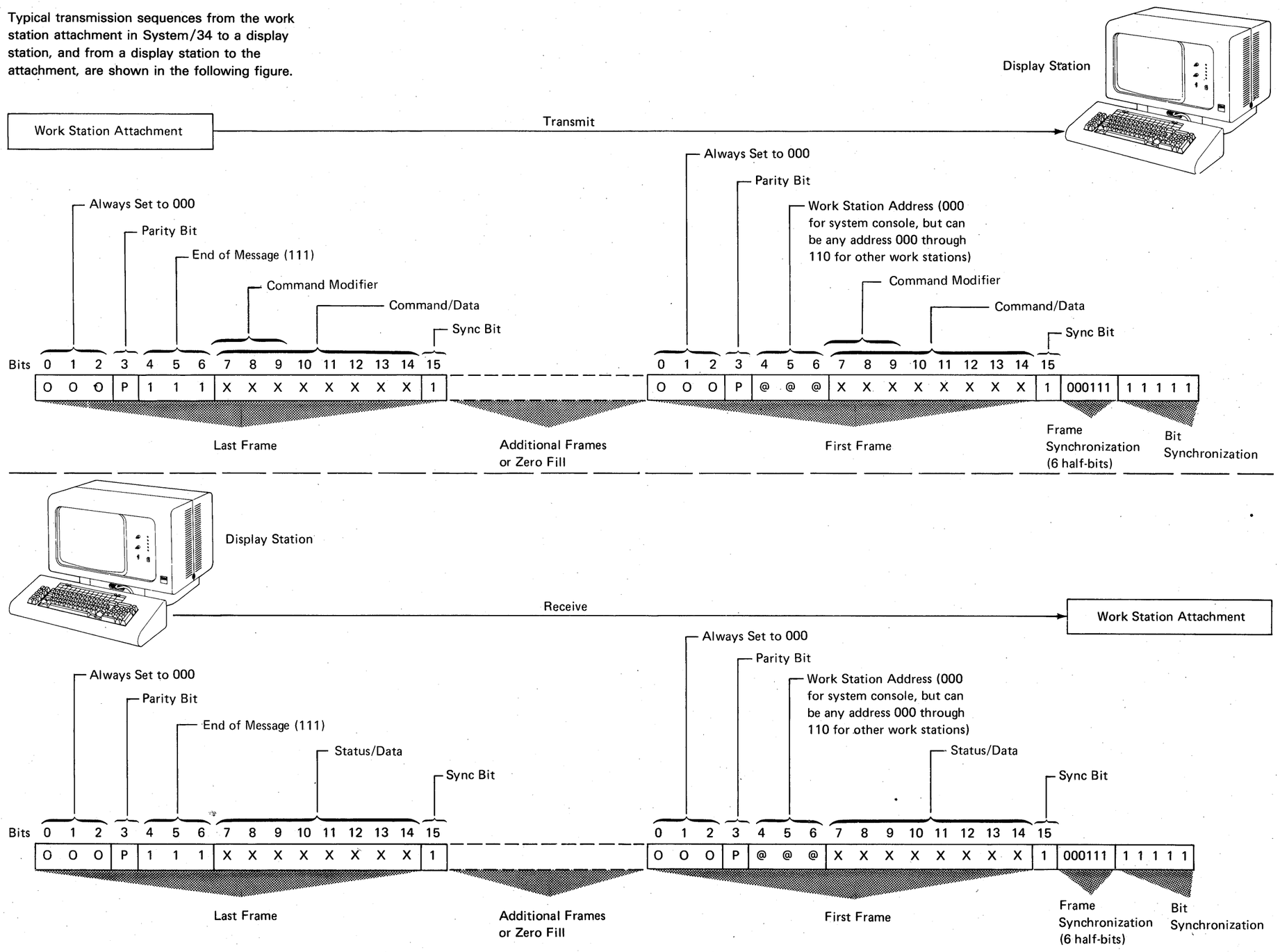
Line Turnaround

Work station line turnaround lasts from the time a work station (in receive mode) receives the last bit of a frame (bit 0) to the time the same work station goes into transmit mode and starts sending the bit synchronization bits. In the work stations, line turnaround occurs in a minimum of 15 microseconds to a maximum of 30 microseconds.

Attachment line turnaround lasts from the time the last bit of a frame (bit 0) is transmitted by the attachment (in transmit mode) to the time the attachment goes into receive mode and starts receiving the bit synchronization bits. Attachment line turnaround is less than 15 microseconds because the attachment must be ready to receive a frame from the work station in 15 microseconds (minimum time for work station line turnaround).

Typical Transmission Sequences

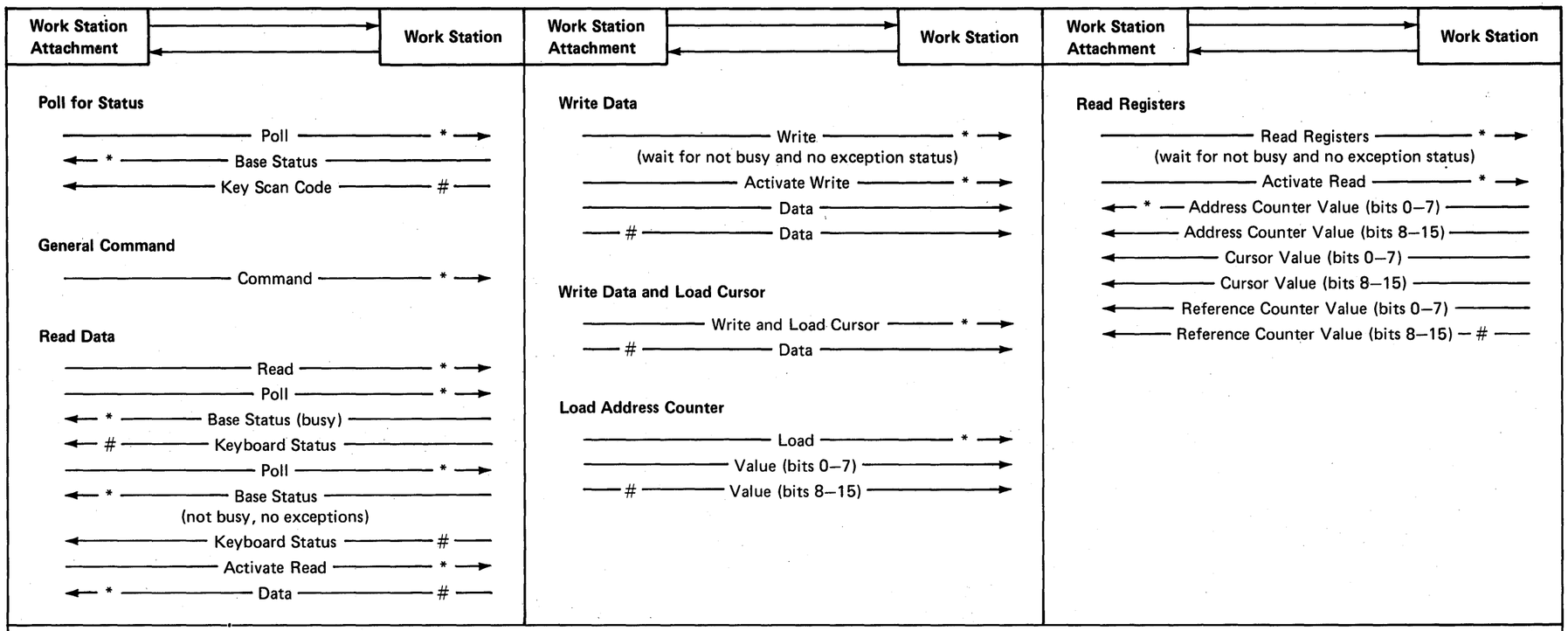
Typical transmission sequences from the work station attachment in System/34 to a display station, and from a display station to the attachment, are shown in the following figure.



Examples of Transmission Sequences

The work stations never start a transaction to or from the work station attachment. Instead, the work station attachment starts all transactions via a command, and the work stations respond to the commands. For example, if the work station attachment issues an activate read command, the work station responds by returning data to the work station attachment.

Information is passed between the work station attachment and the work stations in a half duplex mode; that is, information is sent in only one direction at a time. The following figure shows some of the typical commands sent to the work stations, and the information that is returned to the work station attachment.



Legend:

* Bit synchronization and frame synchronization bits are generated by the work station attachment. These bits condition the work station to start receiving a command.

\# Data stream terminating delimiter that ends the message and causes the work station to expect frame synchronization bits next. For example, under the Poll for Status command, the message ends as soon as the key scan code is sent to the work station attachment. Then, the work station can expect to receive the frame synchronization bits before receiving any other information.

→ A 16-bit frame from the work station attachment to the work station.

← A 16-bit frame from the work station to the work station attachment.

This page intentionally left blank.

Commands

The following table gives all the command names and the decode of bits 7 through 14 of the 16-bit frame. Bits 7, 8, and 9 are either the command modifier, or they are the device address. Bits 10 through 14 are always the command bits when the system sends a command to the work station. Only the valid command bit configurations are shown; all other bit configurations are reserved.

Command Name	Modifier/ Device Address	Command	Addressed To	Notes
	7 8 9	10 11 12 13 14		
Activate read	0 0 0	0 0 0 0 0	Base	Device address is specified by preceding command.
Activate write	0 0 0	0 0 0 0 1	Base	
Clear	0 0 0	1 0 0 1 0	Base	
EOQ delimiter	0 1 1	0 0 0 1 0	Base	
Insert character	0 0 0	0 0 0 1 1	Base	
Load address counter	0 0 0	1 0 1 0 1	Base	
Load cursor	0 0 0	1 0 1 1 1	Base	
Load reference counter	0 0 0	0 0 1 1 1	Base	
Move data	0 0 0	0 0 1 1 0	Base	
Poll	X X X	1 0 0 0 0	Base	See note below for a description of bits 7, 8, and 9.
Read immediate field	0 0 0	1 1 0 0 1	Base	Activate required
Read between limits	0 0 0	1 1 0 1 0	Base	Activate required
Read registers	0 0 0	1 1 1 0 0	Base	Activate required
Reset	0 0 0	0 0 0 1 0	Base	
Search attribute	0 0 0	0 0 1 0 0	Base	
Search null	0 0 0	1 0 1 0 0	Base	
Set mode	0 0 0	1 0 0 1 1	Base	
Write control data	0 0 0	0 0 1 0 1	Base	
Write data and load cursor	0 0 0	1 0 0 0 1	Base or indicators	
Write immediate data	0 0 0	1 1 1 0 1	Base	Activate required

Note: Bits 7, 8, and 9 of the poll command have the following meaning:

- Bit 7 = Should be 0.
- Bit 8 = Resets the line parity check bit (bit 8 of a base response).
- Bit 9 = Acknowledge bit. Acknowledges that status has been received from the work station, and that status can be updated.

Frame Formats

The following figure shows the different formats for the 16-bit frame. In every format, bit 3 is shown as the parity (P) bit, and bit 15 is shown as the sync (S) bit. In the *Keyboard Response Format*, bit 7 is shown as the make/break (M/B) bit.

In response to a poll command, the work station sends two frames of information. The first frame in the response informs the system of the work station's status. This first frame is shown below as the *Base Response Format*. The second frame in the response gives the key scan code, and is shown below as the *Keyboard Response Format*.

Basic Format

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fill			P	Station Address			Base Device ID			Command					S

Data Word Format

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fill			P	Station Address			Data Character								S

Base Response Format

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fill			P	Station Address			Busy	Line Parity	Not Used	Not Used	Exception Status			Level	S

Determines new or old status from the work station

Keyboard Response Format

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fill			P	Station Address			M/B	Key Scan Code							S

OPERATIONS

As the following figure shows, the work station attachment contains a controller card, an adapter card, storage cards, and a driver/receiver card. All communications between the System/34 control processor, the work station controller, the work station adapter, and the work stations occur through the work station adapter card. For a description of the adapter card, see *Functional Units* earlier in this section.

The attachment controls:

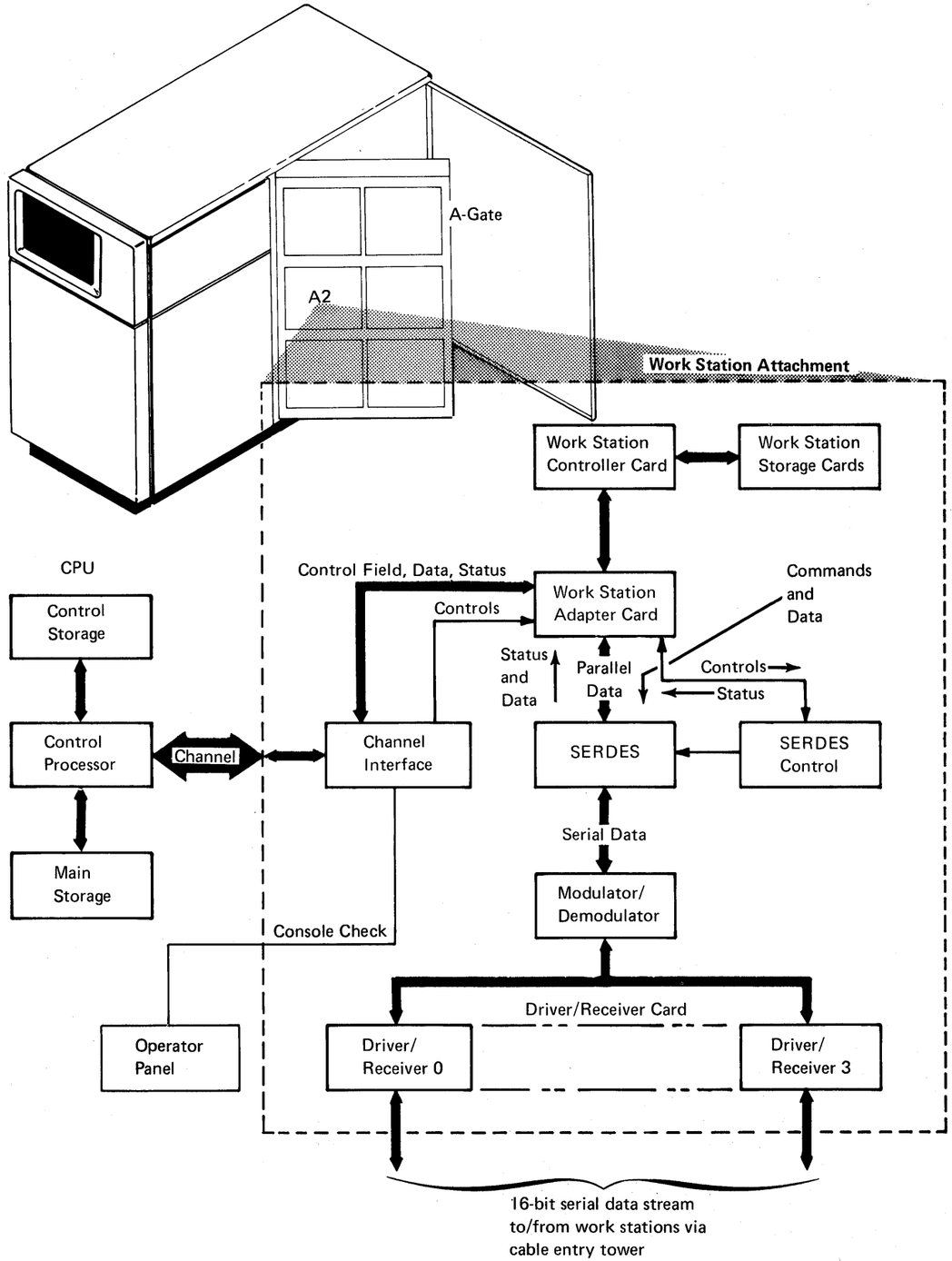
- The movement of data and command information from the channel to the work stations
- The movement of data and status information from the work stations to the channel
- The Console Check light on the operator panel

The attachment also controls and processes keystrokes from the 5251 or 5252 Display Station(s) and control keys from the 5256 Printer(s).

The work station controller card is an important part of the work station attachment; when used with the work station input/output control handler (WSIOCH), it performs all work station control functions. This card is the same as the controller card for the 5211 Printer (the printer is optional). The card is described in the *Attachment Controllers* section of this manual.

The work station controller storage is both read-only storage and read/write storage. At system initial program load time, the processing unit I/O instructions cause the data on the channel to be put into the controller read/write storage. Then, the controller is started and the work station program is executed. This program causes the controller to perform the following functions:

- The controller controls the movement of commands and data to and from main storage and control storage and the work stations.
- The controller periodically polls all work stations and processes work station keystrokes and status conditions.



Displaying a Message

The flow chart on this page shows the operations needed for displaying a message on a 5251 Display Station. The sequence of events for displaying a message is:

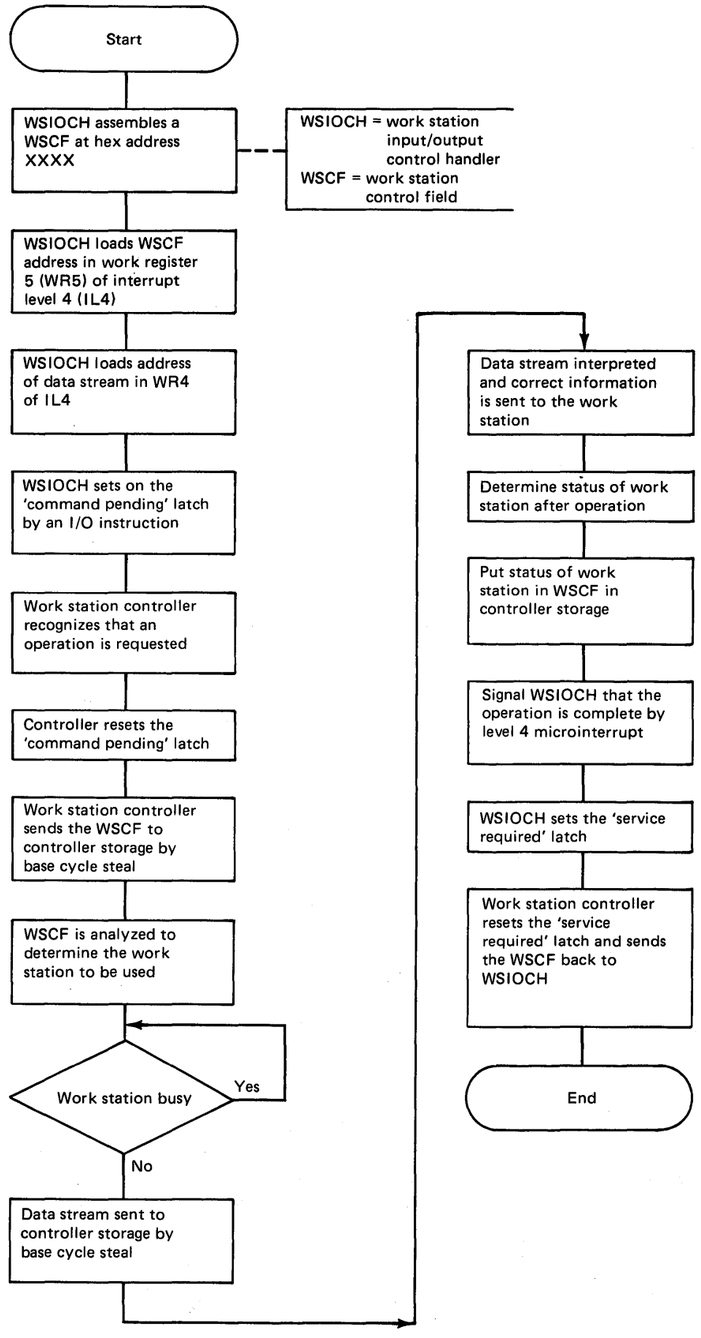
1. The work station input/output control handler (WSIOCH) in control storage sets a 'service required' latch in the attachment with an I/O instruction (through the channel) to signal the controller to send the work station control field back to WSIOCH.

2. The controller samples the 'service required' latch; if it is active, the work station control field is sent back to WSIOCH by base cycle steal. Then, the 'service required' latch is reset and the work station controller interrupts the control processor on interrupt level 4.

3 Then WSIOCH does the following:
   a. Assembles a work station control field in a fixed location of control storage.
   b. Loads work register 5 of interrupt level 4 with the address of the first byte of the work station control field.
   c. Loads work register 4 of interrupt level 4 with the address of the first byte of the main storage data stream associated with the work station control field.
   d. Sets the 'command pending' latch in the attachment with an I/O instruction (through the channel) to signal the controller that a work station control field and its associated data stream are prepared.

4. The controller:
   a. Samples the 'command pending' latch and determines that an operation is to be performed.
   b. Receives the work station control field 1 byte at a time from the host by base cycle steal. The controller signals the processing unit to use work register 5 on interrupt level 4 as the starting address of the work station control field to be received and that the work station control field is in control storage.
   c. Analyzes the work station control field to determine the function to be performed and which work station is specified.
   d. Checks the work station status in the station parameter list to determine if the work station is able to receive the data stream.
   e. Receives the data stream from the host by base cycle steal. The controller signals the processing unit to use work register 4 of interrupt level 4 as the starting address of the data to be received and that the data is in main storage.
   f. Analyzes the data stream and sends the correct information to the work station through the serial interface.
   g. Communicates with the work station through the serial interface to determine the status of the work station after an operation; then, puts the status into the work station control field and sends the work station control field to the host by base cycle steal. Work register 5 on interrupt level 4 is used for this operation.
   h. Resets the 'command pending' latch and signals the WSIOCH routine with a level 4 microinterrupt that the operation is complete.

5. WSIOCH gains control as the result of interrupt level 4 posting the action control word. WSIOCH finds that the work station control field contains a valid device address, and posts the terminal unit block complete.



Transmit and Receive Operations

The work station attachment is either in transmit mode, receive mode, or idle mode (neither transmit nor receive mode). A system reset from the channel or an adapter reset (CCB bits 0 through 6 equal to 110 1100) in the attachment keeps the attachment in idle mode.

To keep the attachment in idle mode, either the 'system reset' signal or the 'adapter reset' signal resets the SERDES control (G in the *Data Flow* figure), and the SERDES (H). These reset lines also de-activate the outputs of the driver select logic (S) and disable the driver/receiver (P) when the attachment is in idle mode. Neither reset line is active in transmit mode or receive mode.

In the following descriptions of transmit and receive operations, the words *controller instruction* are followed, in parentheses, by five numbers. These five numbers are the decode of the controller CBO bits 3 through 7. See *Controller Command Decode* earlier in this section for the decode of these bits.

The descriptions of the transmit and receive operations contain circled alphabetic characters. These characters reference the *Data Flow* figure earlier in this section.

Transmit Operation

Before a transmit operation starts, the 'go', 'transmit enable', and 'receive enable' latches in the SERDES control (G) and the SERDES (H) are in a reset condition. To start a transmit operation, a controller instruction (01000) loads the cable address into the CBI and cable address register (L). Another controller instruction (10000) sets the 'transmit enable' latch and, therefore, the SERDES is no longer in a reset condition. After the 'transmit enable' latch is set, the modulator/demodulator (K) starts to generate the bit synchronization and frame synchronization bits. While these bits are being generated, another controller instruction (00010) causes controller DBO bits 0 through 7 to be loaded into the SERDES (bits 7 through 14) as the command or data byte. The SERDES bits are the same as the bits of the serially transmitted frame; that is, bit 0 of SERDES is bit 0 of the frame, bit 1 is bit 1, and so on.

After bits 7 through 14 of the SERDES are loaded, another controller instruction (00011) causes the new information on controller DBO bits 0 through 7 to be loaded into the remainder of SERDES. The SERDES control is then informed that the SERDES is now loaded with a full frame to be transmitted.

When the SERDES control senses the last frame synchronization bit, it uses the 'serial out' signal (starting with the sync bit) from the SERDES to generate the 'transmit binary data' signal, which is sent to the modulator/demodulator. The modulator synchronizes the 'transmit binary data' signal to the 'transmit clock' signal and then supplies the modulated 'transmit data' signal to the driver (P). The driver selected (by the cable address register and the driver select logic (S)) then generates a differential signal to represent a 1 for the sync bit and transmits this bit on the twinaxial cable.

To transmit the next 11 bits of the frame (bits 4 through 14), the SERDES control shifts the SERDES by 1 bit for each bit transmitted. Then, similar to the operations that occurred for the sync bit, the following occurs for each bit:

1. The SERDES control uses the 'serial out' signal to generate the 'transmit binary data' signal.

2. The modulator synchronizes the 'transmit binary data' signal to the 'transmit clock' signal and supplies the 'transmit data' signal to the driver.

3. The driver generates a differential signal to represent a 0 or a 1 on the twinaxial cable.

The parity bit (bit 3) is not shifted out of the SERDES as the first 12 bits were. Instead, the parity bit is generated by the parity latch in the SERDES control. This latch is reset when either the 'transmit enable' or the 'receive enable' signal is not active. Therefore, the latch is reset before the transmit operation starts but is set when the sync bit is sensed. Then, the latch is alternately reset or set by each 1-bit (for bits 4 through 14) on the 'transmit binary data' signal. After bit 4 has been transmitted, the condition (on or off) of the parity latch determines if the parity bit is 0 or 1. If the latch is on, the parity bit is 1; if the latch is off, the parity bit is 0. In either case, the parity bit generates even parity on bits 3 through 15 of the frame.

After the parity bit is generated, bits 0 through 2 (which are always 000) are generated by the SERDES control and are transmitted as the last 3 bits of the frame. When the frame has been transmitted, the SERDES control sets the 'go' latch, which causes 0-bits (fill bits) to be transmitted until the next frame is transmitted.

The controller senses that the 'go' latch is set and loads the next frame into the SERDES. When the SERDES is loaded, the SERDES control starts to transmit the new frame.

Receive Operation

Before a receive operation starts, the 'go', 'transmit enable', and 'receive enable' latches in the SERDES control (G) and the SERDES (H) are in a reset condition. Also, the driver part of the driver/receiver (P) is disabled.

In addition, the work station attachment must send a poll command to which the work station must respond. When the response from the work station is recognized by the work station attachment, the attachment can start to receive data.

To start a receive operation, a controller instruction (01000) loads the cable address into the CBI and cable address register (L). The output of the cable address register causes the receiver select logic (Q) to select the output of the correct receiver. Another controller instruction (11001) sets the 'receive enable' latch. With this latch set:

- The SERDES is no longer reset, so the attachment can start to receive data.
- The modulator/demodulator (K) is enabled, and it starts checking for the bit synchronization and frame synchronization bits.
- After the frame synchronization bits are recognized, a frame (or frames) of data is (are) received as described in the following paragraphs.

The receiver part of the driver/receiver is always active, and it checks for the four-level differential signal received from the twinaxial cable (see *Differential Data* earlier in this section). Using the four-level differential signal as input, the receiver generates the binary 'receive data' signal.

The receive operation continues with the following sequence:

1. The 'receive data' signal is gated through the receiver select logic to the receive clock generator (R) and to the modulator/demodulator.
2. The output of the receive clock generator is gated to the modulator/demodulator as the 'receive clock' signal.
3. The modulator/demodulator generates the 'receive binary data' signal (from the 'receive data' and 'receive clock' signals) and gates it to the SERDES control.
4. The SERDES control gates the 'receive binary data' signal to the SERDES on the 'serial in' line.

The data on the 'serial in' line is shifted into SERDES 1 bit at a time, starting with bit 15 of the frame (the sync bit). Bit 14 is shifted in next, then bit 13, and so on. As the data is shifted in, the SERDES control checks for even parity on bits 3 through 15 (sync bit, data, work station address, and parity bit). Then the 3 fill-bits (000) are shifted into SERDES.

The SERDES control senses when the SERDES is full (contains 16 bits) and causes the data in SERDES to be sent to the controller DBI buffer. The SERDES control also sets the 'go' latch to inform the controller that the controller DBI buffer contains a valid frame of data.

A controller instruction (X1101) is then issued to sense the condition of the 'go' latch. If the latch is set, two more controller instructions (X0000 and X0001) are issued to sense the contents of the SERDES. Then, assuming that the work station address bits (bits 4 through 6 of the frame) are not set to 111, the receive operation continues with the next frame.

Additional frames are received until bits 4 through 6 of the frame are set to 111, which indicates an end of message (the last frame sent on this receive operation). When the end of message is sensed, the 'receive enable' latch is reset. Also, the SERDES and the 'go' latch are held reset because neither the 'transmit enable' nor the 'receive enable' latch is set.

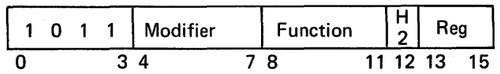
Note: The 'receive enable' latch is reset when the end of message is sensed so that any noise on the twinaxial cable cannot be sensed as the bit synchronization bits or the frame synchronization bits.

COMMANDS

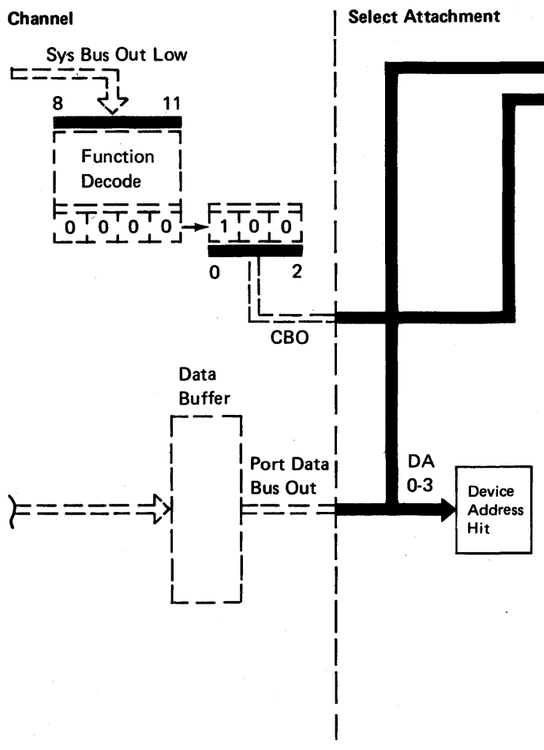
The work station attachment recognizes the following I/O immediate commands from the control processor:

- I/O load
- I/O control load
- I/O sense

These commands are decoded from the function field (bits 8-11) and sent to the attachment on CBO bits 0-2. The format of the I/O immediate command is:

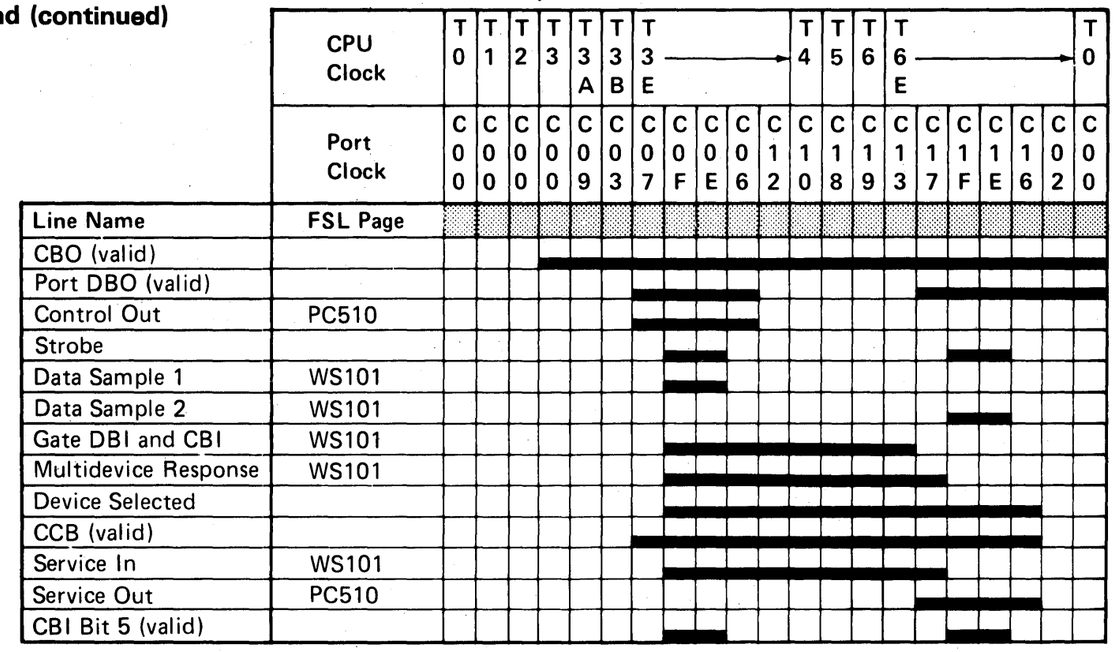


I/O Load Command



CCB 0 1 2	CCB 3 4 5 6	DBO Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
100 I/O Load	0 0 0 0 Set diagnostic mode								
	0 0 0 1 Reset diagnostic mode								
	0 0 1 0 Diagnostic set cycle steal transfer error								
	0 0 1 1 Reset microinterrupt request								
	0 1 0 0 Set microinterrupt request								
	0 1 0 1 Load data buffer	DBO Bits to Data Buffer							
	0 1 1 0 Disable microinterrupt request								
	0 1 1 1 Enable microinterrupt request								

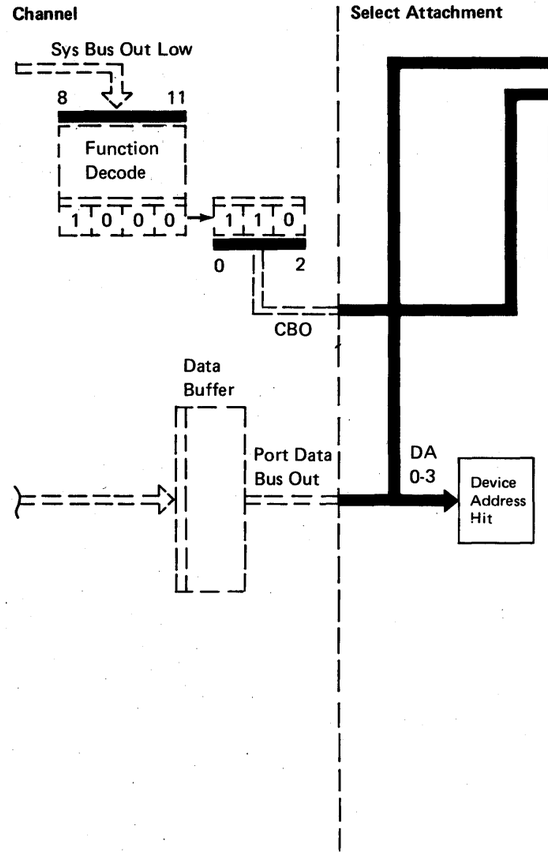
I/O Load Command (continued)



Modifier Port DBO 4, 5, 6, 7	Port DBO Bit	Command	Action Taken	FSL Page
0 0 0 0		Set diagnostic mode	Sets the 'diagnostic mode' latch	WS111
0 0 0 1		Reset diagnostic mode	Resets the 'diagnostic mode' latch	WS111
0 0 1 0		Diagnostic set cycle steal transfer error	Sets the 'cycle steal transfer error' latch for diagnostic purposes	WS111
0 0 1 1		Reset microinterrupt request	Resets the 'microinterrupt request' latch	WS111
0 1 0 0		Set microinterrupt request	Sets the 'microinterrupt request' latch for diagnostic purposes	WS111
0 1 0 1	0-7	Load data buffer	Loads DBO bits 0-7 in the work station controller DBI buffer	WS121
0 1 1 0		Disable microinterrupt request	Resets the 'enable interrupt request' latch	WS111
0 1 1 1		Enable microinterrupt request	Sets the 'enable interrupt request' latch	WS111

I/O Control Load Command

The DBO bits are not used for any I/O control load commands.

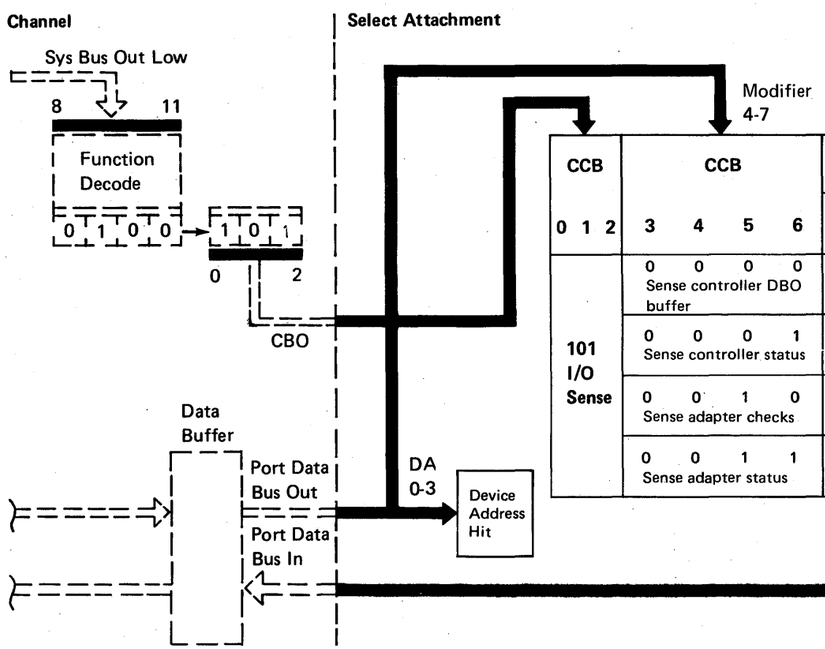


CCB 0 1 2	CCB 3	4	5	6	Command	Action Taken	FSL Page
110 I/O Control Load	0	0	0	0	Set service required	Sets the 'service required' latch	WS111
	0	0	0	1	Set command pending	Sets the 'command pending' latch	WS111
	0	0	1	0	Reset controller load	Resets the 'controller load' latch	WS111
	0	0	1	1	Set controller load	Sets the 'controller load' latch	WS111
	0	1	0	0	Reset console check	Resets the 'console check' latch	WS111
	0	1	0	1	Set console check	Sets the 'console check' latch	WS111
	0	1	1	0	Set controller reset	Sets the 'controller reset' latch	WS111
	0	1	1	1	Reset controller reset	Resets the 'controller reset' latch	WS111
	1	0	0	0	Start controller	Generates the 'start' signal in the controller	WS111
	1	1	0	0	Adapter reset	Resets the adapter	WS111

I/O Control Load Command (continued)

	CPU Clock	T 0	T 1	T 2	T 3	T 3 A	T 3 B	T 3 E →	T 4	T 5	T 6	T 6 E →	T 0
	Port Clock	C 0 0	C 0 0	C 0 0	C 0 0	C 0 9	C 0 3	C 0 7 C 0 F C 0 E C 0 6 C 1 2	C 1 0	C 1 8	C 1 9	C 1 3 C 1 7 C 1 F C 1 E C 1 6 C 0 2	C 0 0
Line Name	**FSL Page**												
CBO (valid)													
Port DBO (valid)													
Control Out	PC510												
Strobe													
Data Sample 1	WS101												
Data Sample 2	WS101												
Gate DBI and CBI	WS101												
Multidevice Response	WS101												
Device Selected													
CCB (valid)													
Service In	WS101												
Service Out	PC510												
CBI Bit 5 (valid)													

I/O Sense Command



CCB 0 1 2	CCB 3 4 5 6	DBO Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
101 I/O Sense	0 0 0 0 Sense controller DBO buffer	Controller DBO							
	0 0 0 1 Sense controller status	Not diagnostic mode	Controller load	Controller reset		Controller ALU carry	Console check	Controller ALU zero	Controller ALU nonzero
	0 0 1 0 Sense adapter checks	Controller busy	Controller op check		Controller DBO/DBI parity check	Serial transfer parity check	Controller memory parity check		
	0 0 1 1 Sense adapter status	Controller XSAR bit 2	CE test pin	Controller card unplugged	Command pending	Microinterrupt disabled		Serial parity trigger	Service required

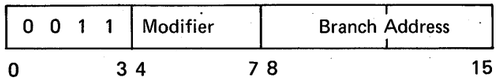
I/O Sense Command (continued)

CPU Clock	T0	T1	T2	T3	T3A	T3B	T3→E	T4	T5	T6	T6→E	T0
Port Clock	C000	C000	C000	C000	C009	C003	C007 C00F C00E C006 C012	C010	C018	C019	C013 C017 C01F C01E C016 C002	C000
Line Name	**FSL Page**											
CBO (valid)												
Port DBO (valid)	PC510											
Control Out												
Strobe												
Data Sample 1	WS101											
Data Sample 2	WS101											
Gate DBI and CBI	WS101											
Multidevice Response	WS101											
Device Selected												
CCB (valid)												
Service In	WS101											
Service Out	PC510											
Port DBI (valid)												
CBI Bit 5 (valid)												

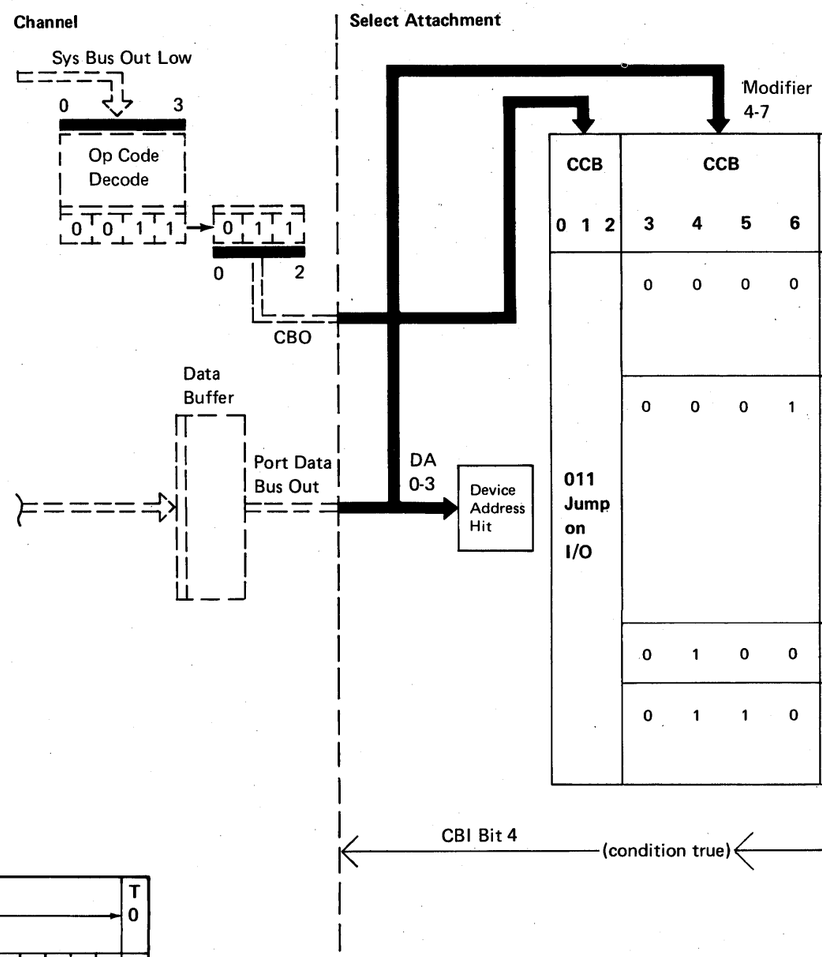
Modifier Port DBO 4, 5, 6, 7	Port DBO Bit	Command	Action Taken	FSL Page
0 0 0 0	0-7	Sense controller DBO buffer	Gates the contents of the data buffer to the channel on port DBI	WS121
0 0 0 1		Sense controller status	Senses the following controller status conditions:	WS121
	0		Not diagnostic mode	
	1		Controller load	
	2		Controller reset	
	4		Controller ALU carry	
	5		Console check	
	6		Controller ALU zero	
	7		Controller ALU nonzero	
0 0 1 0		Sense adapter checks	Senses the following adapter check conditions:	WS121
	0		Controller busy	
	1		Controller operation check	
	3		Controller DBO or DBI parity check	
	4		Serial transfer parity check	
	5		Controller memory parity check	
0 0 1 1		Sense adapter status	Senses the following adapter status conditions:	WS121
	0		Controller storage address register (XSAR bit 2)	
	1		CE test pin	
	2		Controller card unplugged	
	3		Command pending	
	4		Microinterrupt disabled	
	6		Serial parity trigger	
	7		Service required	

Jump on I/O Command

The format of the jump on I/O command is:

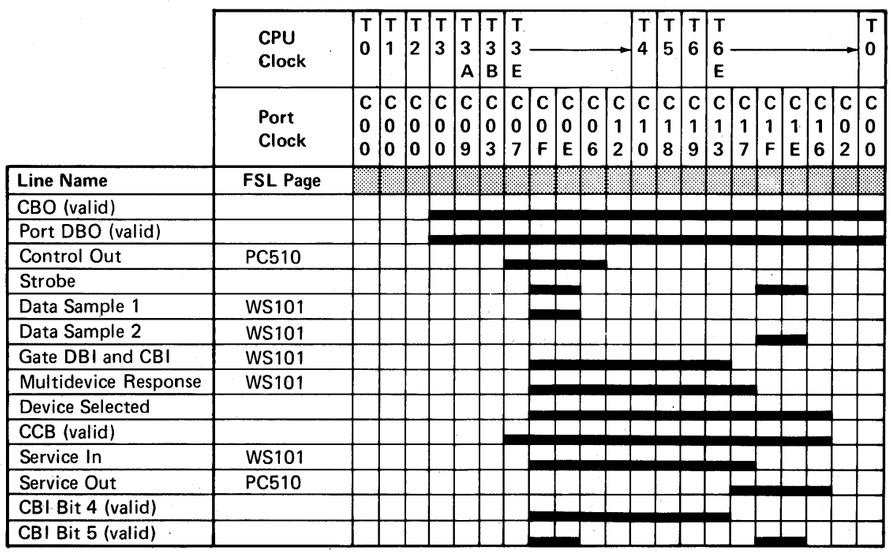


For a jump on I/O command, the work station attachment uses CBI bit 4 to indicate if an I/O condition is met.



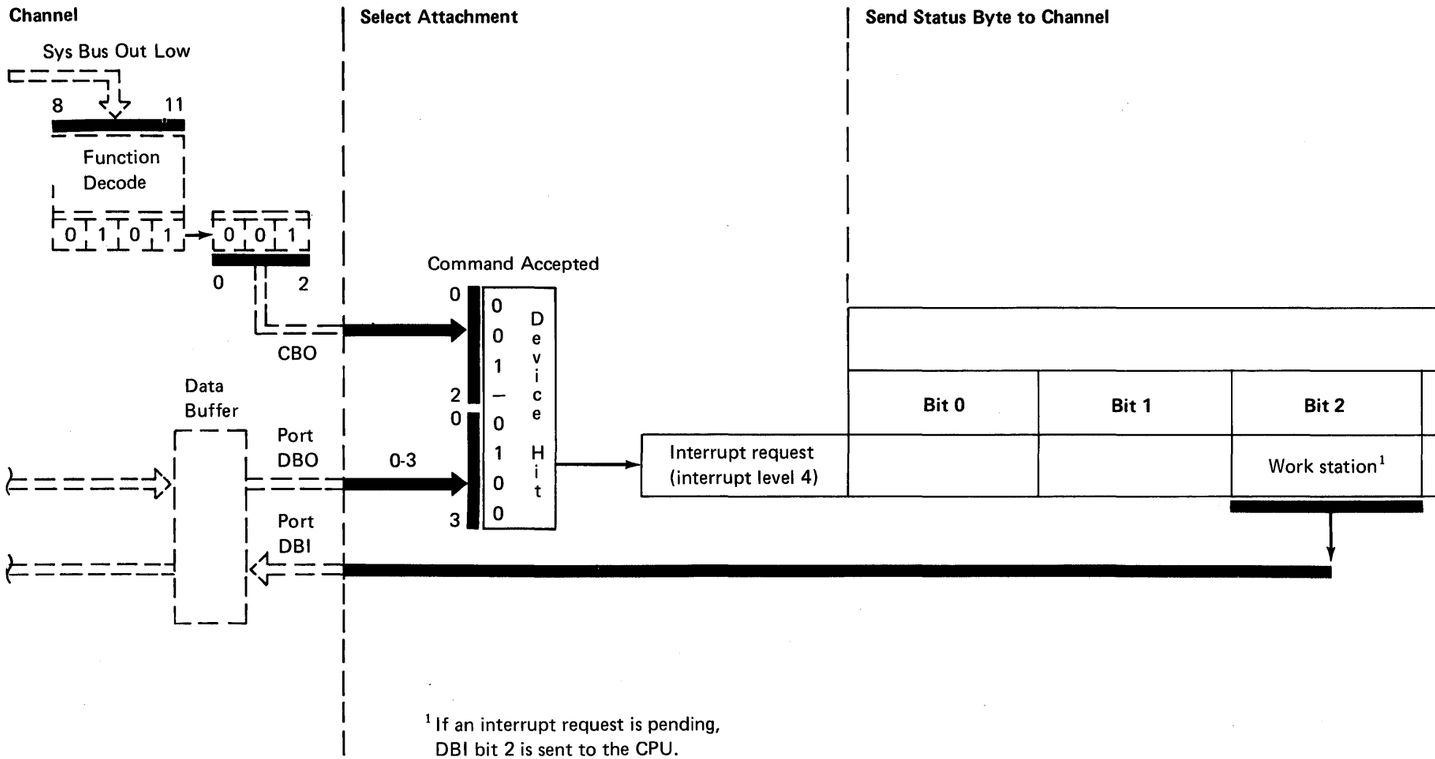
CCB 0 1 2	CCB 3 4 5 6	Condition Tested	Action Taken	FSL Page
011 Jump on I/O	0 0 0 0	Controller check	Activates CBI bit 4 if the 'controller memory parity check' latch, 'controller DBO/DBI parity check' latch, or the 'long time-out check' latch is set.	WS131
	0 0 0 1	Not any check	Activates CBI bit 4 if all of the following latches are reset: • The 'controller DBO/DBI parity check' latch • The 'controller memory parity check' latch • The 'long time-out check' latch • The 'controller operation check' latch	WS131
	0 1 0 0	Diagnostic true[1]	Activates CBI bit 4. Used in diagnostics to guarantee the line will go active.	WS131
	0 1 1 0	Diagnostic false[1]	Does not activate CBI bit 4. Used in diagnostics to guarantee the line will not go active.	WS131

[1] Diagnostic use only.

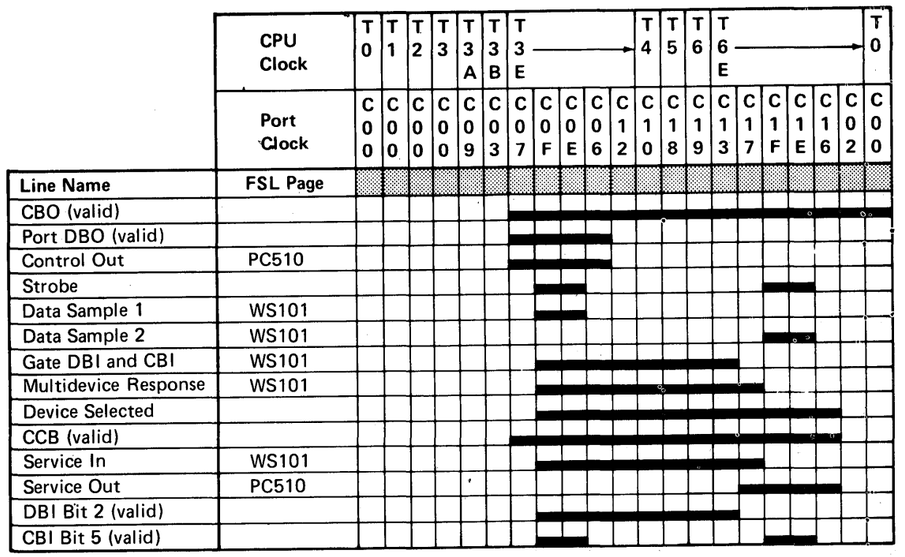


Sense Interrupt Level Status Byte Command

On a sense interrupt level status byte (SILSB) command, the work station adapter activates DBI bit 2.



[1] If an interrupt request is pending, DBI bit 2 is sent to the CPU.

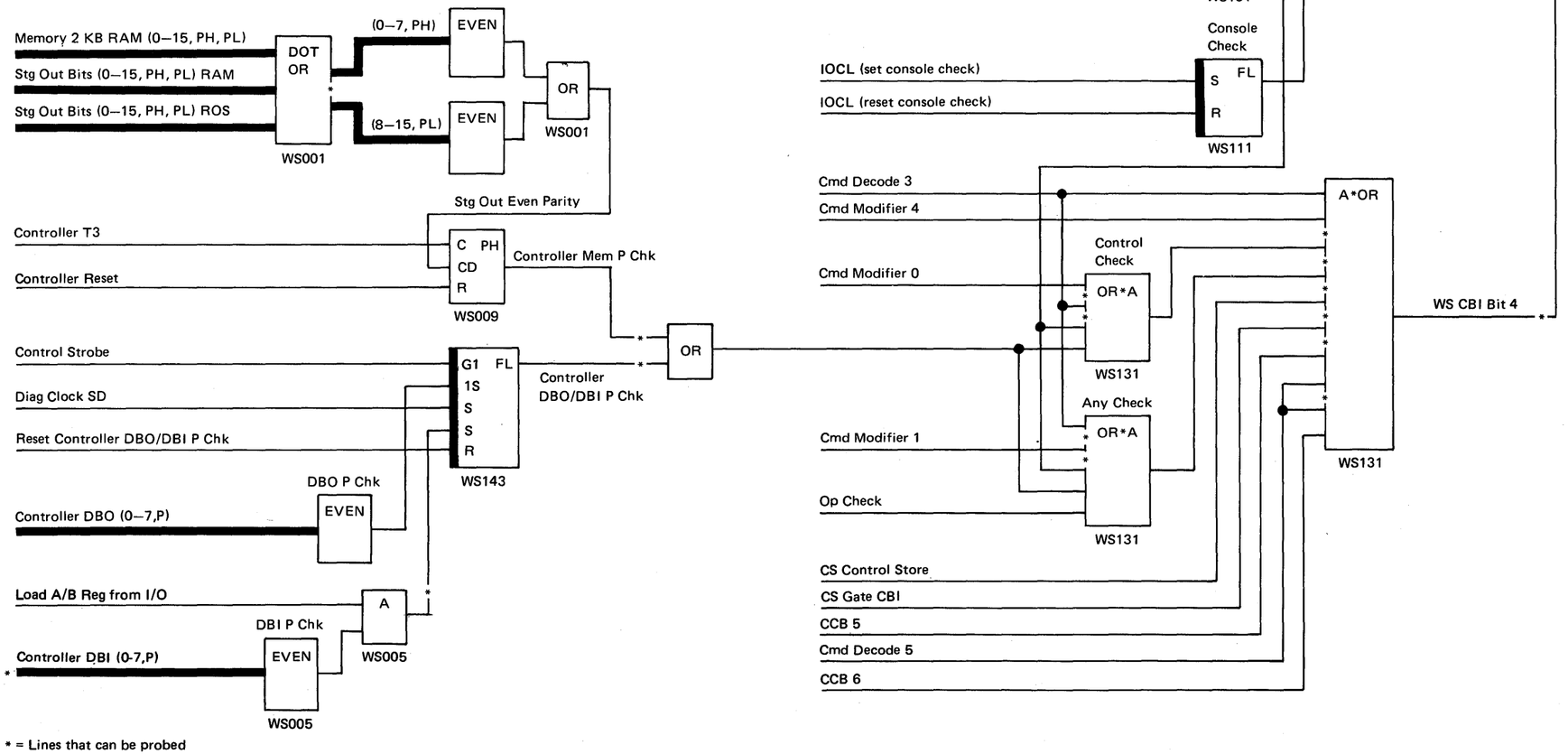


Line Name	FSL Page
CBO (valid)	
Port DBO (valid)	
Control Out	PC510
Strobe	
Data Sample 1	WS101
Data Sample 2	WS101
Gate DBI and CBI	WS101
Multidevice Response	WS101
Device Selected	
CCB (valid)	
Service In	WS101
Service Out	PC510
DBI Bit 2 (valid)	
CBI Bit 5 (valid)	

ERROR CONDITIONS

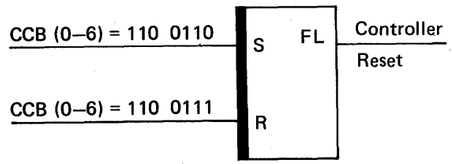
All of the error conditions, described on the following three pages, are shown on the following second-level diagram. Notice that the two outputs (*Console Check* and *WS CBI Bit 4*) at the right of the second-level diagram are the two lines that turn on the Console Check light and the Proc Check light.

Error Condition	Console Check Light	Proc Check Light	Suggested Recovery Action
Controller storage parity check	X	X	Do an initial program load.
Controller DBO/DBI parity check	X	X	Do an initial program load.
Long time-out check	X	X	Do an initial program load and a print dump.
Serial parity check			Turn power to the work station off, then turn power back on. Also, check the cables and cable connections.
Channel DBO parity check		X	Do an initial program load and a print dump.



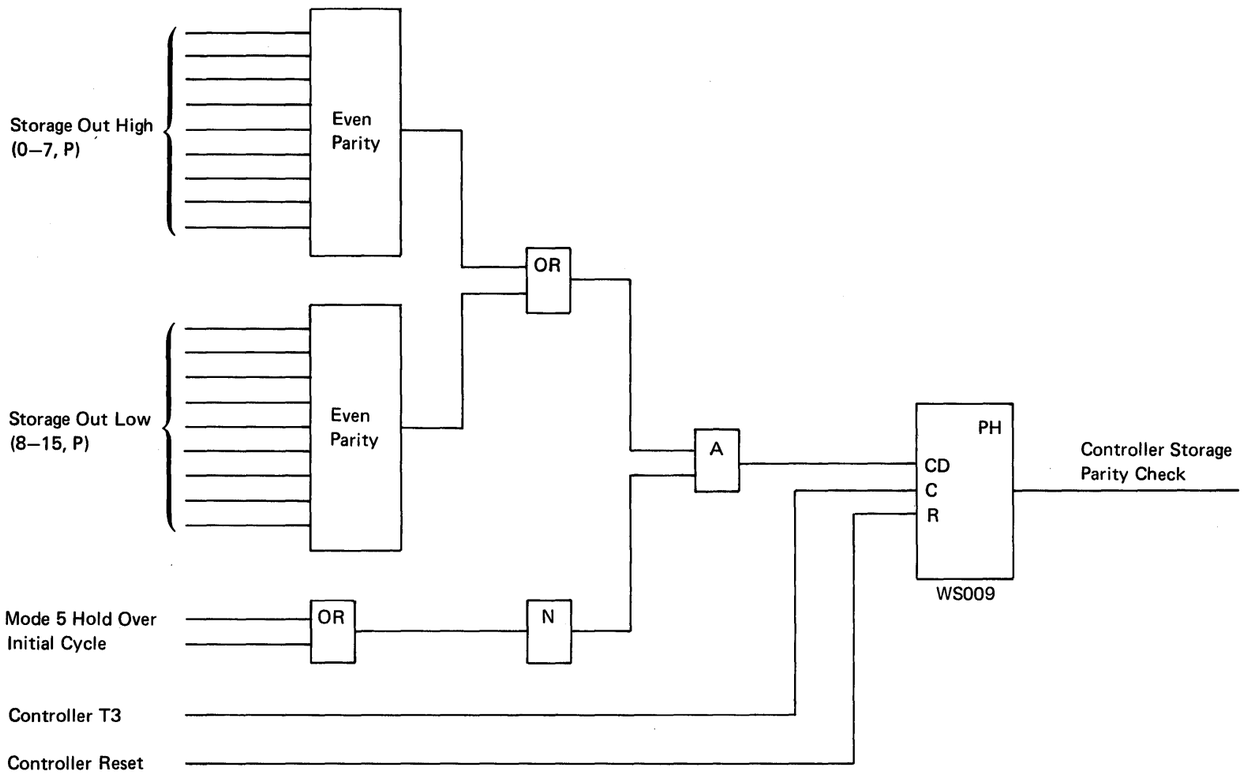
* = Lines that can be probed

Each description of the error conditions includes a second-level diagram. Most of these diagrams show an 'adapter reset' line and a 'controller reset' line. The 'adapter reset' line is generated by an I/O control load instruction where CCB bits 0-6 are decoded as 110 1100. The 'controller reset' line is generated by two I/O control load instructions, as shown in the following figure.



Controller Storage Parity Check

The controller storage output is checked for even parity on both the high and low bytes of an instruction, and on the data read from storage. If either the high or low byte has even parity, the 'controller storage parity check' latch is set and the controller stops at the end of the current cycle. However, an error does not occur if the latch is not sensed. The latch can be sensed by an I/O sense instruction.



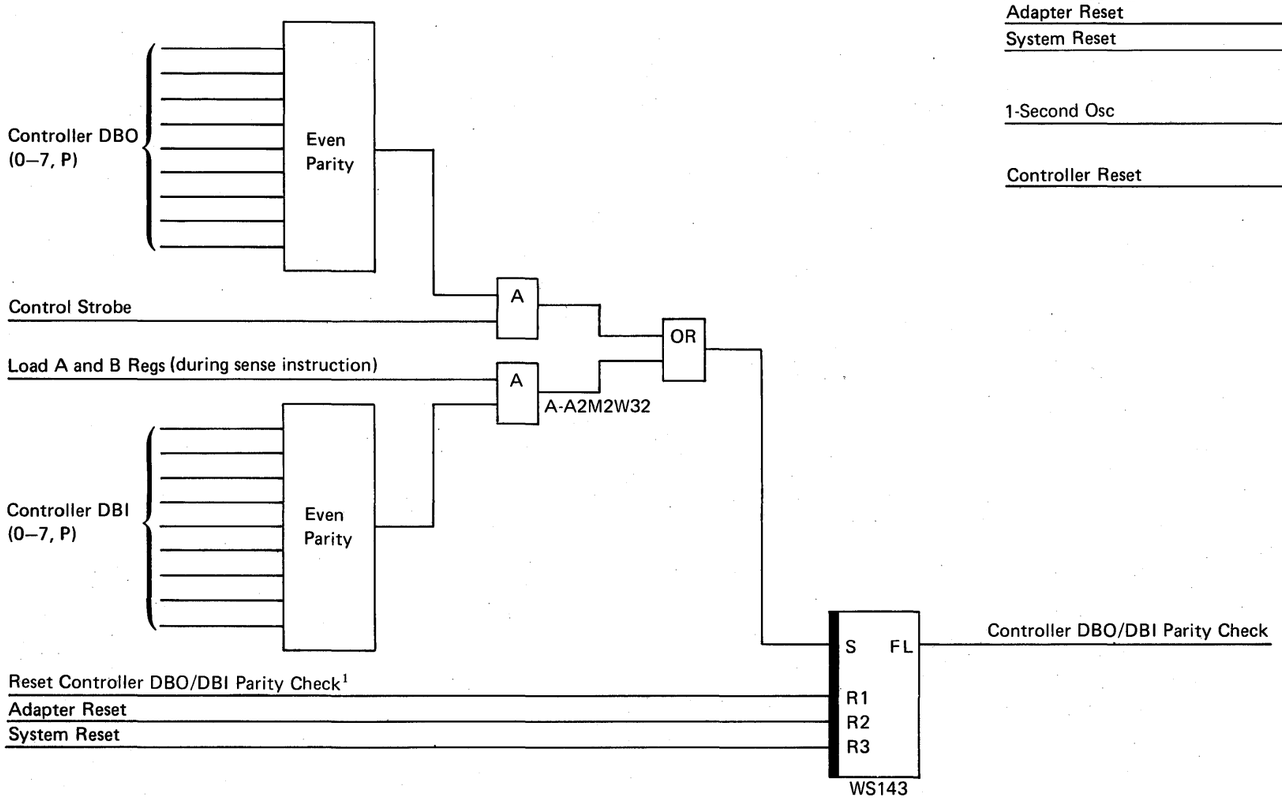
This page intentionally left blank.

Controller DBO/DBI Parity Check

A controller DBO/DBI parity check occurs for either of the following conditions:

- If controller DBO contains even parity during control strobe time of a controller I/O load instruction
- If controller DBI contains even parity when checked during a controller I/O sense instruction

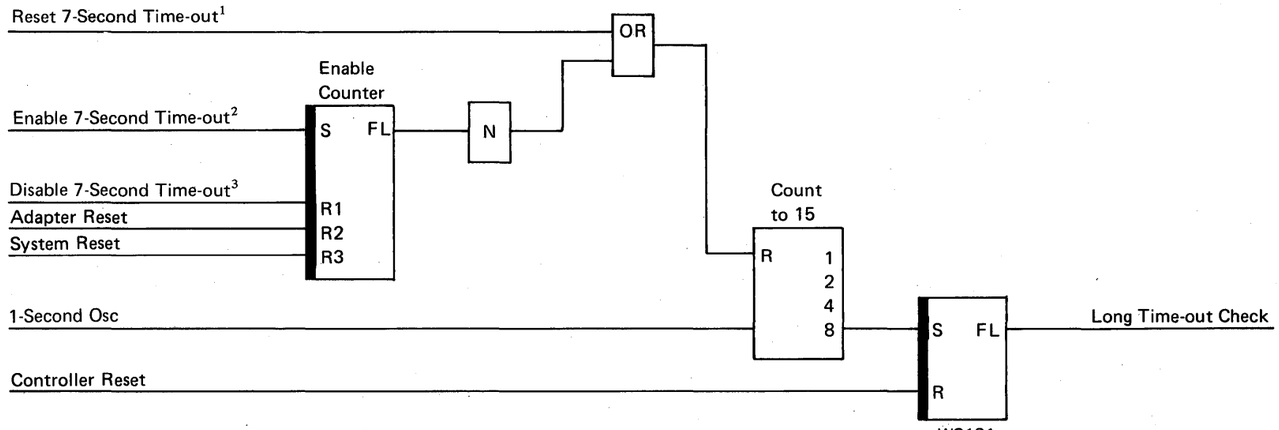
To determine which of these two conditions caused the error, it is necessary to probe A-A2M2W32. If positive, the error was caused by a controller DBI parity check; if negative, the error was caused by a controller DBO parity check. (See the output of the bottom AND block in the figure below.)



[1] Controller CBO (3-7) = 01101

Long Time-out Check

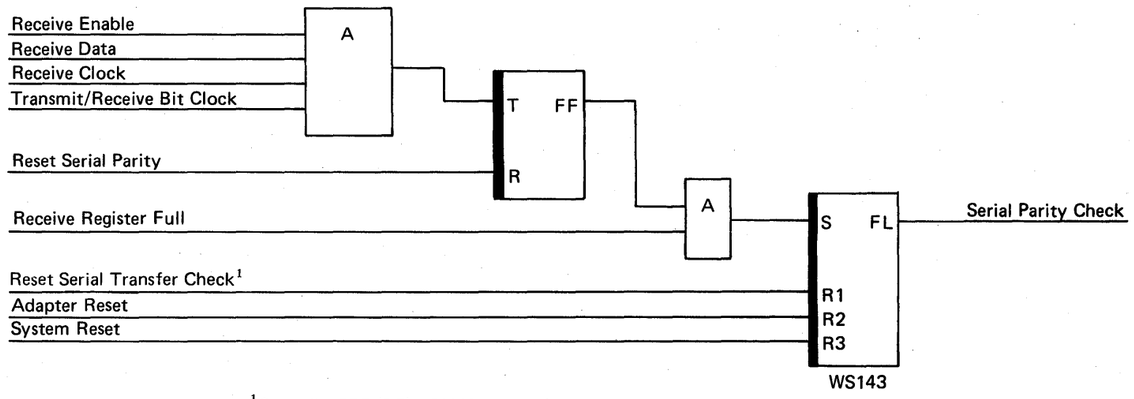
A long time-out check occurs if the controller does not repeatedly reset the counter every 7 seconds. As soon as the counter reaches a count of 8, the 'long time-out check' latch is set. The latch is not reset at the correct time if the controller is in a loop or if it has taken a wrong branch.



[1] Controller CBO (3-7) = 11110
[2] Controller CBO (3-7) = 11000
[3] Controller CBO (3-7) = 11011

Serial Parity Check

A serial parity check occurs when odd parity is sensed on a frame of data received from the work station.

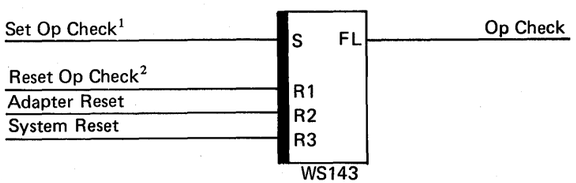


[1] Controller CBO (3-7) = 01110

Work Station Controller Check (Any Check)

A work station controller check occurs when a DBO/DBI parity check, long time-out check, or an op check occurs. An op check is sensed when the microcode senses one of the following hardware errors:

- SERDES time-out
- Cycle steal time-out
- Work station controller interrupt late

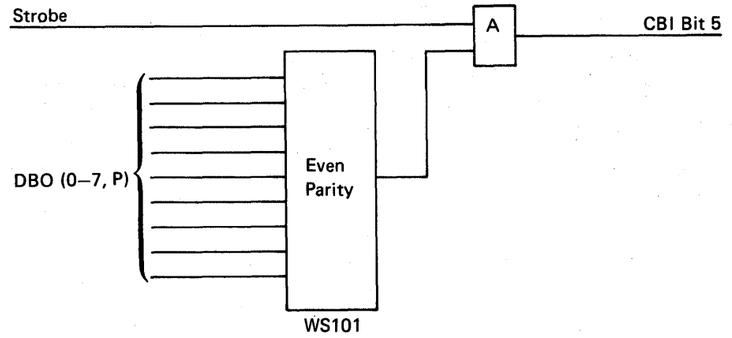


[1] Controller CBO (3-7) = 10111
[2] Controller CBO (3-7) = 01001

Channel DBO Parity Check

A channel DBO parity check occurs when even parity is sensed on the DBO bus during I/O instructions. CBI bit 5 is activated at strobe time to indicate the parity check.

Note: A channel DBO parity check and a controller DBO/DBI parity check can occur at the same time.



BASE CYCLE STEAL

The work station attachment transfers data to or from control storage or main storage via the channel by base cycle steal. When the work station controller determines that data is to be transferred, the work station controller sets a latch that activates the 'base cycle steal request' signal to the channel. By sensing this signal, the channel causes the control processor to stop at the end of the current instruction or at the end of a disk burst cycle steal operation. Then, the channel causes a base cycle steal operation to be executed.

When the base cycle steal operation is executed, 1 byte of data is transferred to the work station attachment from control storage or main storage, or 1 byte of data is transferred from the work station attachment to control storage or main storage. The direction in which data is transferred is determined by the CBI bits to the channel.

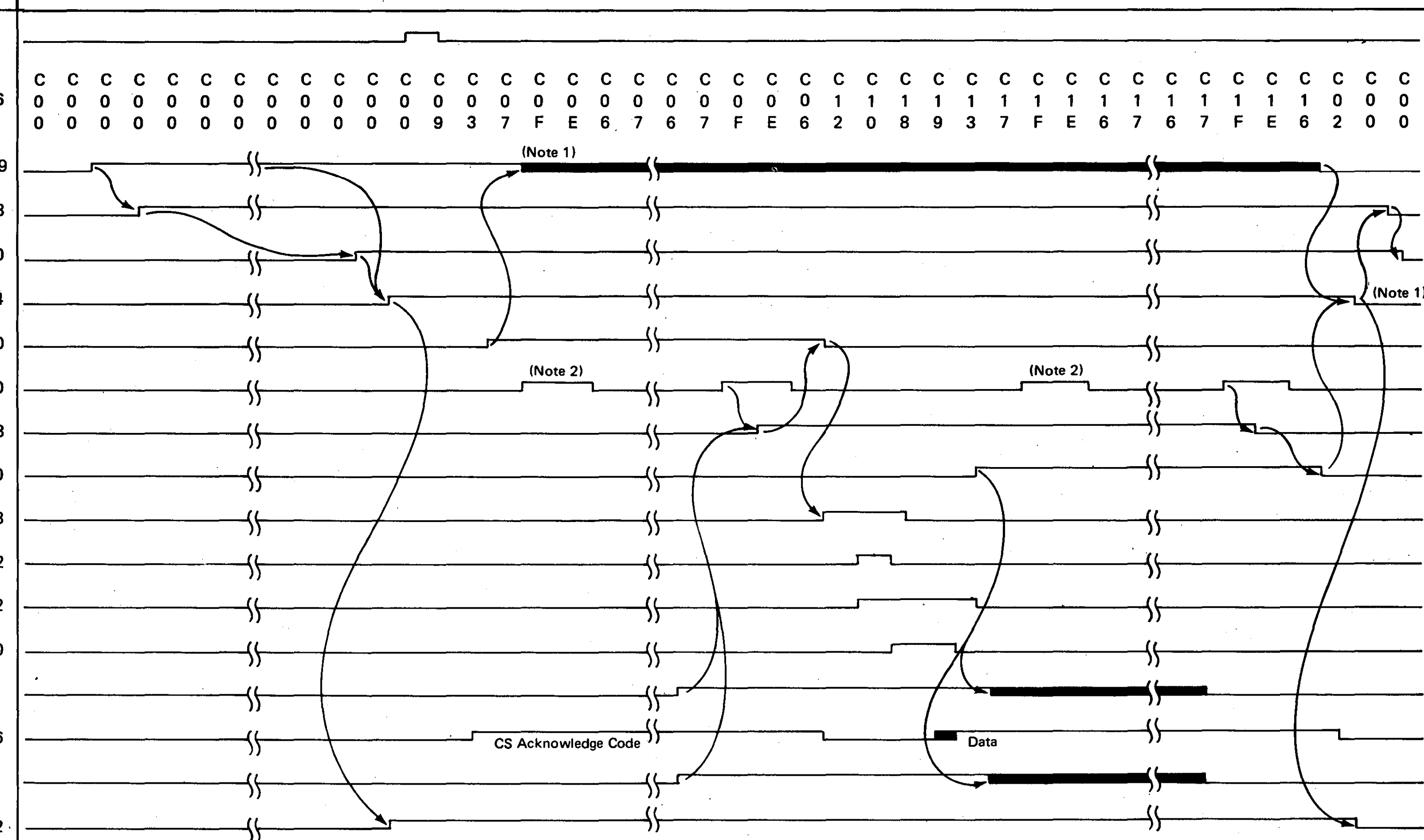
Once the base cycle steal operation is started, the channel control lines (CBO, strobe, service in, and so on) and their timings are the same as for an I/O load or an I/O sense instruction except that:

- CBO bits 0 through 2 are equal to 010. The 'base cycle steal request' latch in the work station attachment is reset when CBO bits 0 through 2 are equal to 010 during 'control out' and 'strobe' time. At this time, the 'base cycle steal request' signal to the channel is de-activated.
- CBI bits let the channel determine (1) the type of instruction (load or sense), (2) the type of storage (control or main), and (3) the storage address register (work register 4 or 5 on interrupt level 4).
- DBO bits 0 through 3 are set to hexadecimal 0 during 'control out' instead of to hexadecimal C, which is the work station address.

The timing relationship of the channel control lines to the 'base cycle steal request' signal is shown in the timing chart on this page.

The advantage of a base cycle steal operation over a typical I/O instruction is that the data is transferred without a control processor program controlling the transfer. To transfer a byte of data during a base cycle steal operation, only 3.2 microseconds (one I/O instruction cycle) of control processor time are needed instead of the 10 to 20 microseconds that are needed for the control processor program.

Line Name	Sequence of Occurrence	FSL Page
T3 Lth		
C-Times (port clocks)		PC526
Base Cycle Steal Request*	1	CH009
Latched Block Processor Clock	2	PC508
T7 Pwrd	3	PC110
Base Cycle Mode (base cycle steal lth)	4	PC534
Control Out Pwrd	7	PC510
Strobe Pwrd	10	PC510
Service In*	11	PC518
Service Out Pwrd	16	PC510
Storage Cycle Request	12	PC508
Clock SAR and X Reg (T8 time)	13	PC142
Storage Cycle Time** (storage function)	14	PC012
CSY Trigger to Channel	15	PC030
Command Bus In (CBI) Valid*	8	
DBO (MPXPO bus out)	6	PC506
DBI (MPXPO data in) Valid (if applicable)*	9	
Command Bus Out (CBO)	5	PC542



Notes:

1. This example shows only one base cycle steal occurring. The entire handshaking sequence would occur again if 'base cycle steal request' remained active beyond the time shown (provided no disk cycle steals were requested).
2. If the I/O device responds to 'control out pwrd' and 'service out pwrd' in synchronous operation, only one strobe will occur under each tag line. Otherwise, up to 7 strobes may occur in each portion of the sequence for asynchronous operation. If 8 strobes occur, a time-out occurs and a blast is generated.

*Signifies signals for the device attachment to the base port card.

**The 'storage cycle time' consists of the time for loading the storage address register (T8), the time for accessing storage (X CSY Tgr), and a time for storing data to the I/O attachment.

INPUT/OUTPUT LINES

The following figure shows the lines between the channel and the attachment, and the twinaxial cables to the work stations. For a description of most of the lines between the channel and the attachment, see *Functional Units* earlier in this section. See *Serial Interface to Work Stations*, also in this section, for a description of the signals transmitted to, and received from, the work stations.