# Contents

# 62PC Disk Drive and Attachment

## INTRODUCTION

### 62PC Disk Drive

The 62PC disk drive has a data storage capacity of 63.9 megabytes. Installation of a second disk drive gives a total data storage capacity of 128.4 megabytes.

Each disk drive has six permanently installed magnetic disks. Data is written to and read from both sides of the disks by data heads attached to an actuator. Each disk drive has 11 data heads and one servo head. The disk data rate is one million bytes per second.

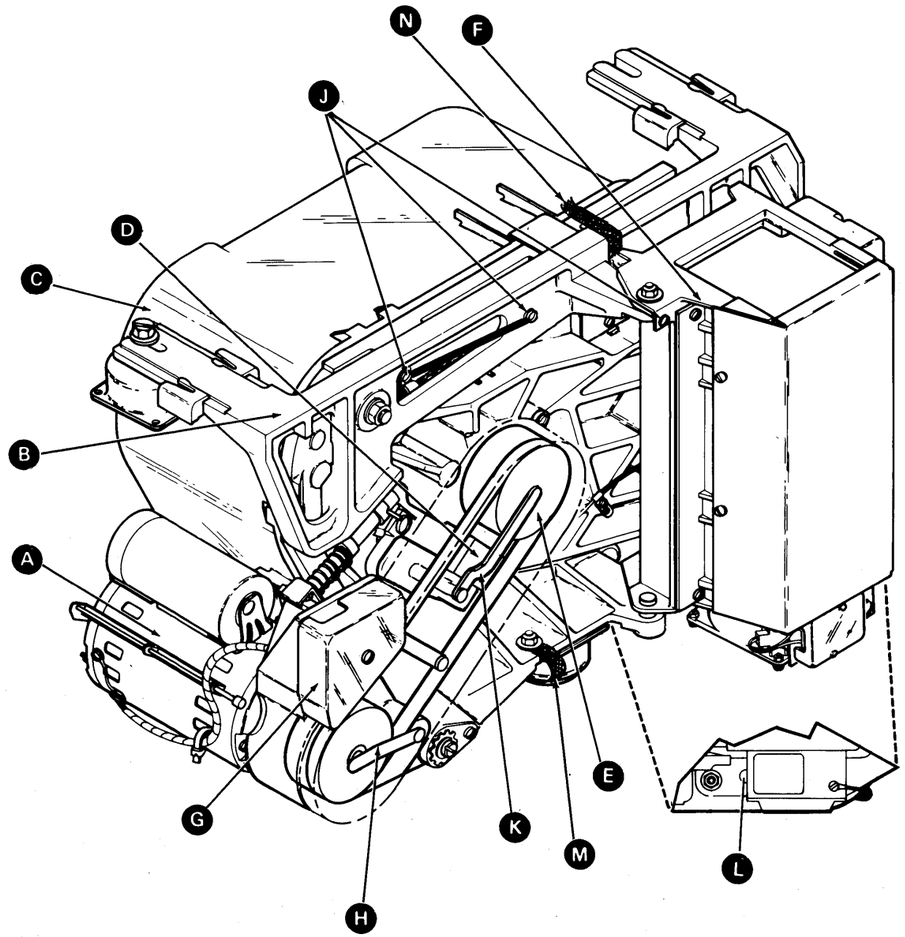
The disk drive contains a subframe (B), disk enclosure (C) and spindle brake (D), drive motor assembly (A), and a card gate (F). For a description of the card gate, see *Circuit Locations* later in this section.

#### Grounding

The disk drive board is grounded by the DC power cable (N) to the system DC ground. The card gate is grounded to the board (ground not shown). The disk enclosure and subframe are grounded to the gate by a wire or strap (J). The subframe is also grounded to the system frame by a strap (M). The drive motor assembly (G) is grounded by the AC power cable to the system AC ground. The spindle (E) is grounded to the disk enclosure by the spindle antistatic brush (K). The motor armature is grounded to the drive motor by the motor antistatic brush (H).

#### Subframe

The subframe is installed vertically in the system unit. Three shock mounts isolate the disk drive from the machine frame. The disk enclosure and spindle brake, the drive motor assembly, and the card gate are installed on the subframe.




#### Disk Enclosure and Spindle Brake

The disk enclosure contains the following assemblies:

- The disk spindle assembly, which contains the disks installed on the disk spindle.
- The spindle brake, which is attached to the disk enclosure casting and quickly stops the disk when power is removed. This limits the time that the servo head and the data heads touch the disk when the disks are turning to prevent head or disk damage. The brake is de-activated when 24 Vdc is present on the brake coil.
- The actuator assembly, which contains the servo head, the data heads, and associated electronic circuits at one end of the actuator and a coil at the opposite end. Actuator movement and location is controlled by current applied to the coil.

*Note:* The disk enclosure is sealed at the factory and should not be opened in the field.

The disk enclosure has a closed-loop, air-moving system that uses blades on the spindle hub to continuously move air through an air filter. Another filter lets the air pressure remain equal as the disk is getting up to speed and as the temperature inside the disk enclosure changes.

#### Actuator Lock

When the actuator lock (L) is turned clockwise as far as it will go, it is in the locked position. The lock prevents damage to the data heads during shipment, installation, or removal of the disk enclosure.

#### Drive Motor Assembly

The drive motor assembly contains the drive motor, pulley, and mounting frame. The assembly pivots on the subframe and a belt tensioner keeps the drive belt tight. The drive motor quickly starts the disk when power is turned on and turns the disk at 3,125 revolutions per minute. The motor has a manual thermal reset switch.

### Attachment

The disk drive attachment includes two adapters: the channel adapter and the common adapter.

- The channel adapter interfaces both to the system channel and to the common adapter. The channel adapter moves data and control information to and from the common adapter, and changes the data from 1-byte length on the system channel to 2-byte length on the common adapter interface. Interrupt level 4 and cycle steal control circuits are in the channel adapter.
- The common adapter executes the disk operations requested by the system. The common adapter controls access and data operations for one or two 62PC disk drives attached to the system. There are two common adapter cards. The channel interface card (A-A2D2) contains a microcontroller, microcode stored in read-only storage, an oscillator and clock circuits, and controls for the channel adapter/common adapter interface. The disk drive interface card (A-A2C2) contains the buffer storage, which is used to store control information and up to three sectors of data; the SERDES, which changes the 2-byte data to serial data; read and write control circuits; and controls for the common adapter/disk drive interface.

## Circuit Locations

The circuits for a disk drive are located in two separate areas: inside the disk enclosure and inside the card gate attached to the disk drive subframe. If a second disk drive is installed, it will have another set of the same circuits.

The attachment circuits are located on the A-A2 board. Additional attachment circuits are not needed if a second disk drive is installed.

The circuits inside the disk enclosure include:

- Read and write head selection
- Read preamplifiers for each head
- Write drivers for each head
- Servo preamplifier

The circuits in the disk drive card gate include:

- Two logic cards (4 wide 3 high), two servo cards (one card 4 wide 3 high, the other card 2 wide 3 high), and a data channel card (4 wide 2 high)
- A driver card for the actuator coil

*Note:* The card gate board is E-A1 for disk drive A and E-B1 for disk drive B.

The circuits on the A-A2 board include:

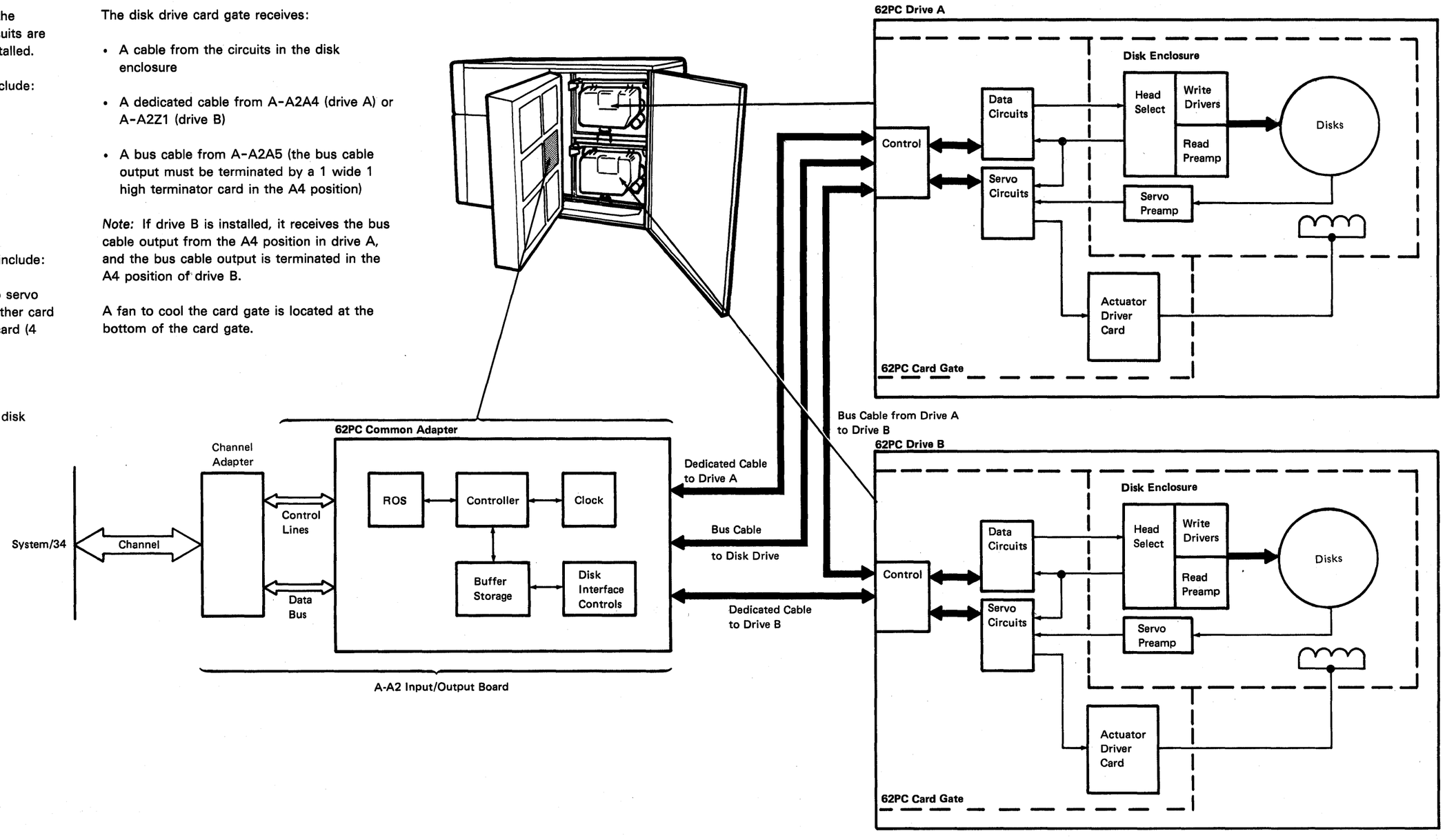
- The channel adapter card in A-A2E2
- The 62PC common adapter cards in A-A2D2 and A-A2C2 (these cards are connected by top card connectors)

The disk drive card gate receives:

- A cable from the circuits in the disk enclosure
- A dedicated cable from A-A2A4 (drive A) or A-A2Z1 (drive B)
- A bus cable from A-A2A5 (the bus cable output must be terminated by a 1 wide 1 high terminator card in the A4 position)

*Note:* If drive B is installed, it receives the bus cable output from the A4 position in drive A, and the bus cable output is terminated in the A4 position of drive B.

A fan to cool the card gate is located at the bottom of the card gate.



## DISK FORMAT

The disk surfaces are divided into cylinders, tracks, and sectors.

## Cylinders

A cylinder is the area that passes under all the data heads in one disk revolution. The 62PC disk drive has 358 data cylinders, 1 alternative cylinder, and 1 CE cylinder.

## Tracks

### Servo Track

A servo track is the area that passes under the servo head in one disk revolution. There is one servo track on the dedicated servo surface for each cylinder. The servo tracks are patterns written on the disk at the factory. These patterns contain clock pulses and position pulses. The clock pulses are used to generate the write clock, and the position pulses are used to keep the heads over the specified track. Patterns of missing clock pulses are used to generate the index and sector pulses.

### Data Track

A data track is the area that passes under a single data head in one disk revolution. The 62PC disk drive has 11 data tracks per cylinder. Each data track has a sample servo area, which contains additional servo information, at the start of each sector.

### Guard Band

The guard band is an area on the servo track surface that contains clock and position pulses, but does not have missing clock pulses to generate index or sector pulses. The behind home area on the data track surface is comparable to the guard band area.

### Landing Zone

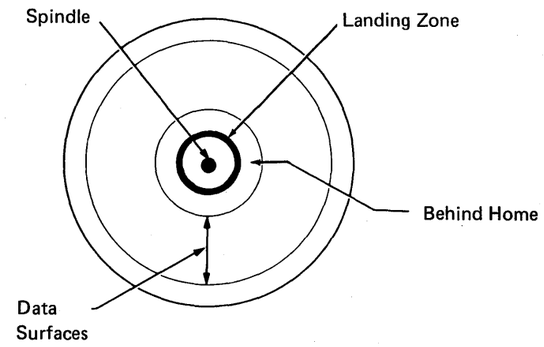
The landing zone is an area on all disk surfaces where the servo head and data heads are positioned (by a spring) when the disk drive is powered off, and when the disk drive is powered on until the disk is up to the correct speed. The landing zone is the area nearest the center of the disk.

| | |
|---|---|
| Disk capacity in megabytes | 64.9 (Note 1) |
| Number of tracks | 3,960 (Note 2) |
| Number of data heads | 11 |
| Number of cylinders | 360 (Note 3) |
| Number of sectors per track | 33 (Note 4) |
| Number of records per track | 64 |
| Bytes per record | 256 |

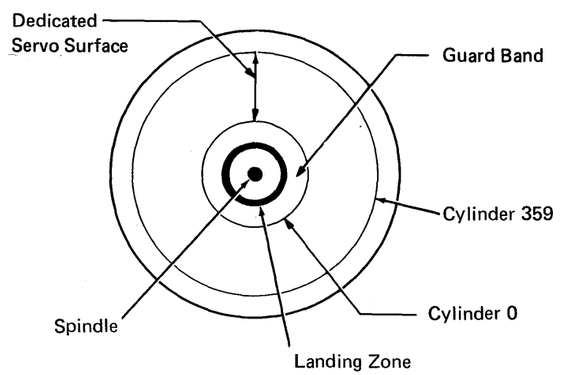
*Notes:*

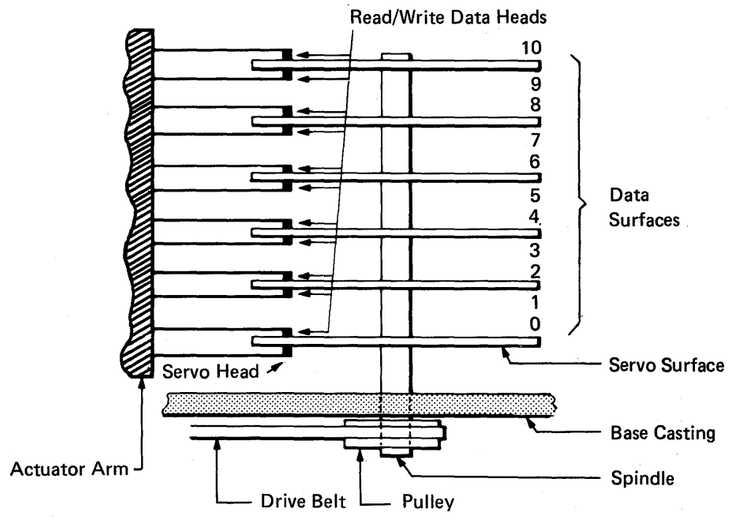
1. 63.9 megabytes available for customer use
2. 3,938 tracks for customer use
   - 11 tracks reserved as alternative tracks
   - 11 tracks reserved for CE use
3. 358 cylinders for customer use
   (cylinders 0 to 357, hex 0 to 165)
   - 1 cylinder reserved as alternative cylinder
     (cylinder 358, hex 166)
   - 1 cylinder reserved as CE cylinder
     (cylinder 359, hex 167)
4. 32 sectors available for customer use

**Data Surface**




**Servo Surface**



## Sectors

A sector is a division of a data track. Each data track is divided into 33 physical sectors, and each physical sector contains two data records. Only 32 of the physical sectors on each track are used for data storage. Therefore, there are 64 records for data storage on each track. Each record contains 256 bytes. Therefore, there are 16,384 bytes per track and 180,224 bytes per cylinder.

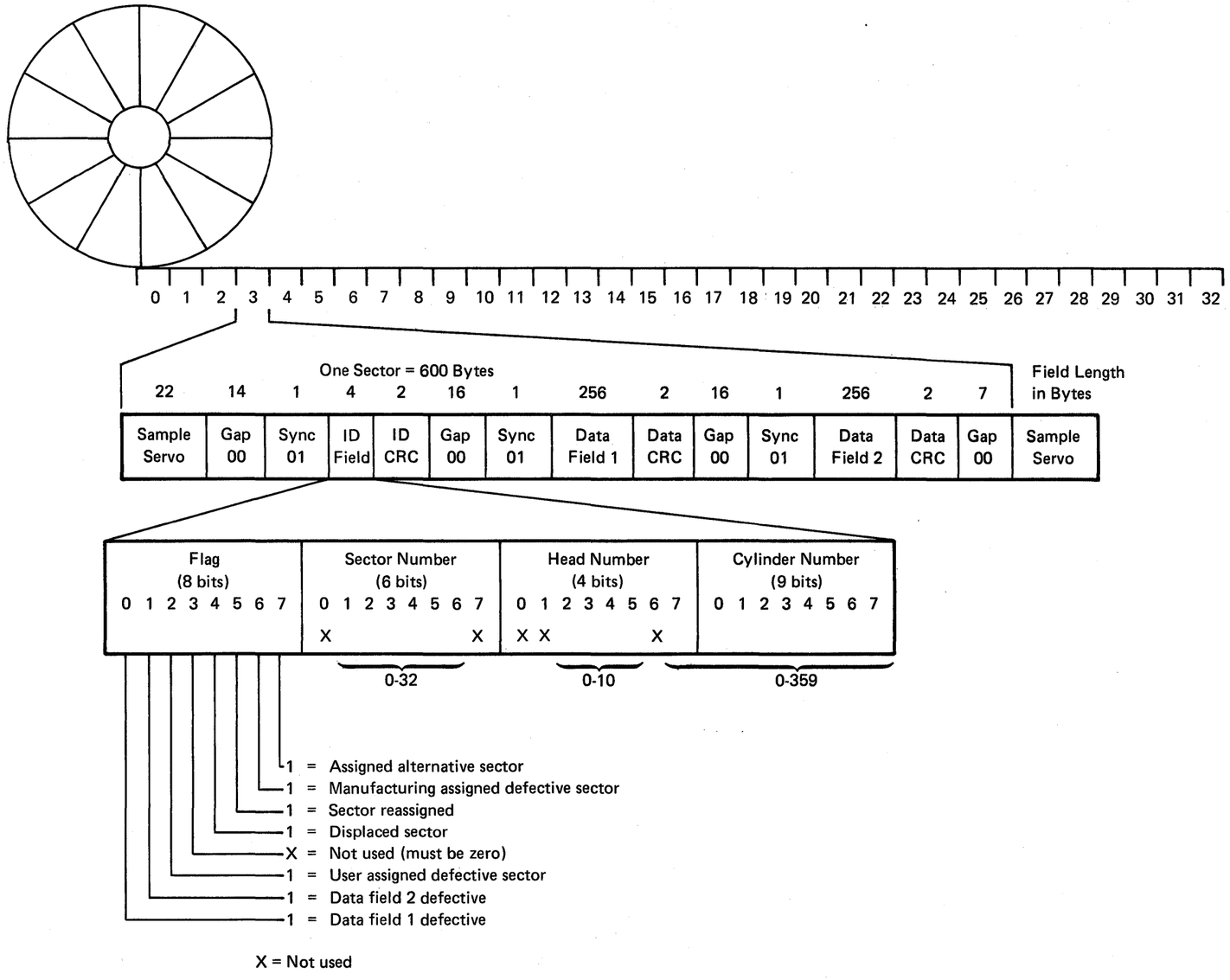
Each physical sector has an identification address that contains the cylinder, head, and sector numbers. This address is recorded in the identification field at the physical location of the sector on the disk. Each track has 33 identification fields.

Each of the 33 physical sectors is 600 bytes long and contains three main fields: the identification field and two data fields. Each data field contains a data record, and both records may be read or written after an ID hit. The records are numbered so that they may be operated on consecutively. Therefore, physical sector 0 (on head 0) contains records 0 and 1, physical sector 1 (on head 0) contains records 2 and 3, and so on. All 64 records on a track can be operated on in one revolution of the disk. Because a read or write operation can continue from one head to another head, the sectors for the next data head are delayed by a quarter revolution, or eight physical sectors, to permit time for head switching and for the servo circuits to become stable after the head switch.

When the disks are assembled, each sector is tested. If a sector is damaged, an alternative sector is assigned for the damaged sector. Because only 32 of the 33 physical sectors on a track are used, the first damaged sector becomes the not used sector. On head 0, the 32 physical sectors that are used are numbered 0 through 31 sequentially, starting from the index pulse and skipping over the damaged sector, which is numbered 32. The damaged sector and all sectors located after it and before the next index pulse must have the displaced sector bit on in the identification field. If two or more sectors on a track are damaged, alternative sectors are assigned on the alternative cylinder. The identification field of the damaged sector contains the address of the alternative sector.

A sector that becomes damaged after the disk has been manufactured is assigned to the alternative cylinder. It becomes the not-used sector on the same track only when the sector fails during the disk initialize program.

Some areas of the disk are reserved. For more information, see the *Data Areas Handbook*.

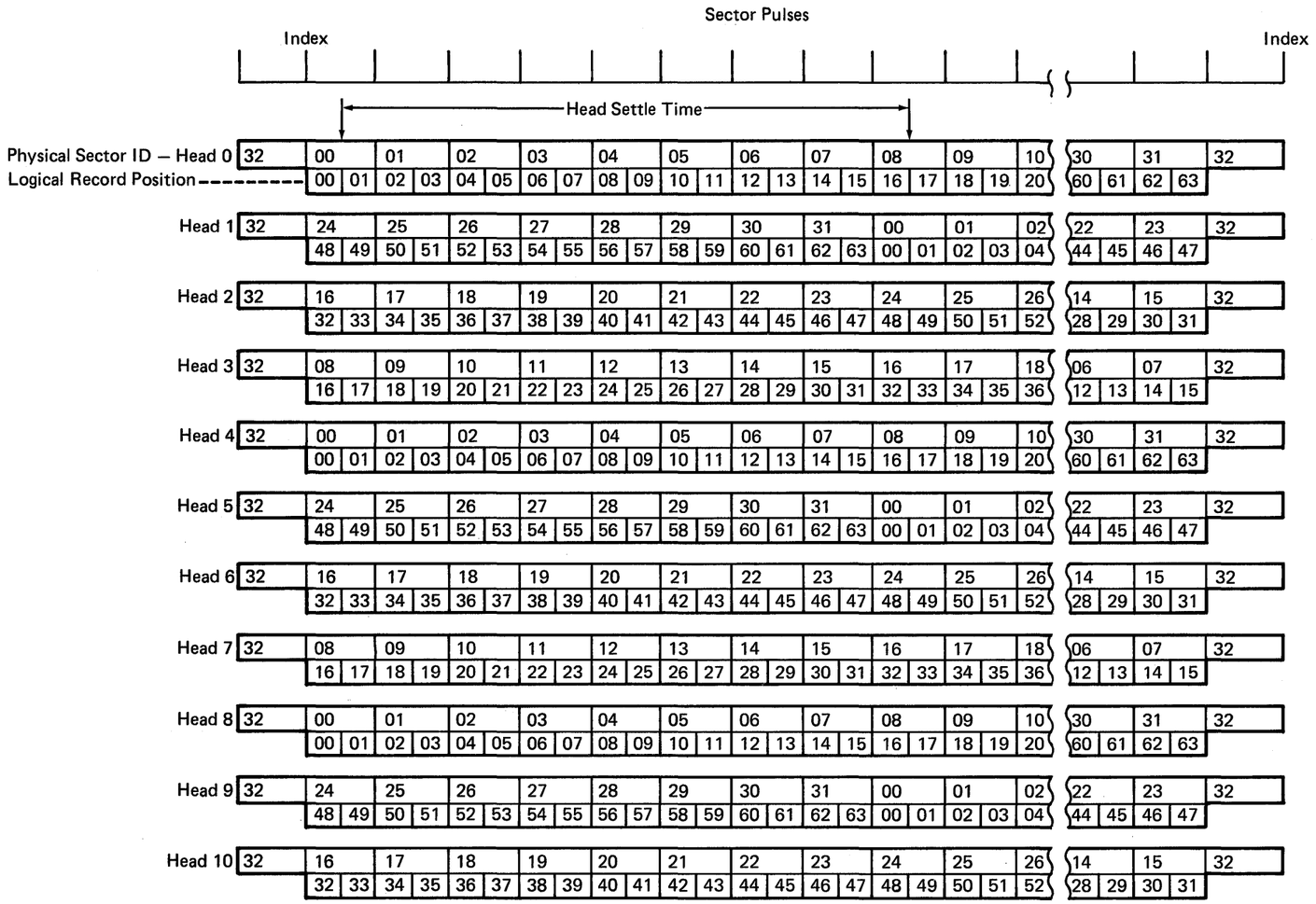



*Note:* Flag bits 0 and 1 may be set in the ID field of an assigned alternative sector to indicate that the data written in the sector was obtained from a defective sector:

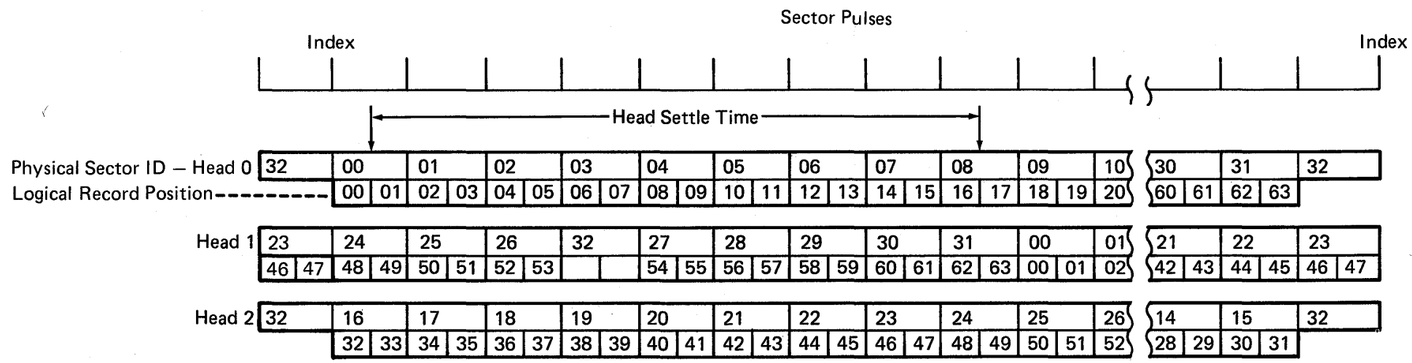
Bit 0 = 1 indicates that data field 1 was read with a CRC check.

Bit 1 = 1 indicates that data field 2 was read with a CRC check.

If the ID field of the defective sector was read with a CRC check, both bit 0 and bit 1 are set to 1. Flag bits 0 and 1 may also be set in the ID fields of sectors that were defective when the disk was manufactured, but these are ignored by the disk attachment.



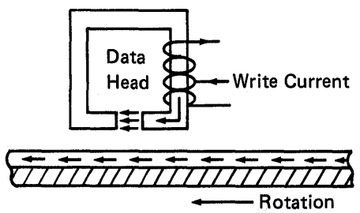

This example shows a displaced sector (head 1, physical sector 27). The flag byte of the displaced sector is 00001010 and the flag byte for sectors 27 through 31 and 0 through 23 is 00001000. If there was a second defective sector (for example, sector 30), an alternative sector would be assigned on cylinder 358 and the flag byte for sector 30 would be 00001110.

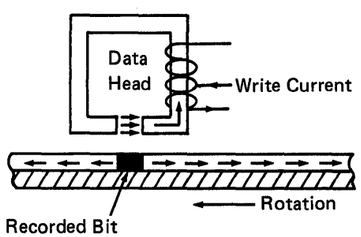


## Data Recording

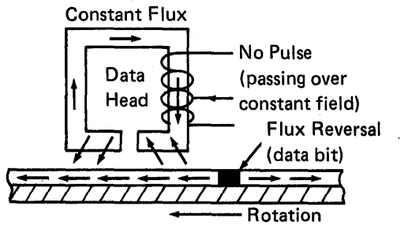
### Reading and Writing Data

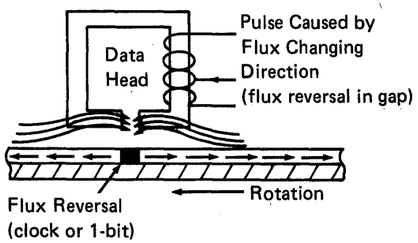
During a write operation, data is recorded by reversing the direction of the current in the coil, which reverses the direction of the magnetic flux in the data head gap. When the flux in the data head gap reverses, there is a magnetic change of direction on the disk surface. Each change of direction on the disk represents a recorded 0-bit or 1-bit.

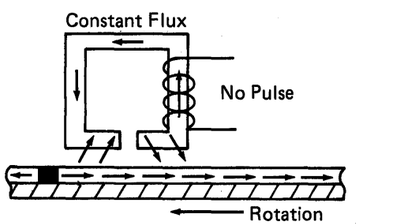



During a read operation, when the disk surface is recorded in one magnetic direction, constant flux flows and the coil senses no output voltage. But, when a 180-degree horizontal flux change passes the gap, the flux through the ring and coil reverses and generates a voltage output pulse. The timing of these pulses indicates if a 0-bit or a 1-bit was written.

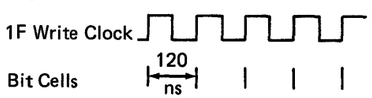


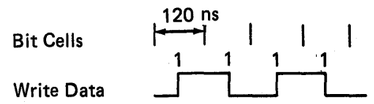

For write operations, the attachment uses clock pulses read from the servo tracks to generate the write clock. On read operations, the read clock is generated using a voltage controlled oscillator synchronized with the data read from the disk.

The time during which a data bit can be written is known as a bit cell. A bit cell is 120 nanoseconds long and is generated by the '1F write clock' line.

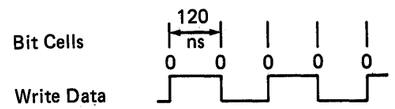



*Writing 1-Bits and 0-Bits*

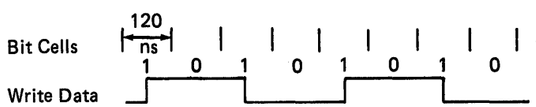
One-bits are always written in the center of a bit cell.



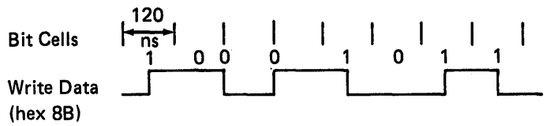

With one exception, 0-bits are always written at the start of a bit cell.




The exception is when the 0-bit immediately follows a 1-bit. In this case, no bit is written and the 0-bit is represented by no magnetic change of direction during its bit cell time.



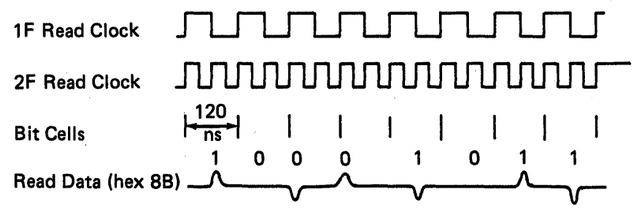

Both methods of writing 0-bits and the method of writing 1-bits can be shown when writing hexadecimal 8B.




Any data written is transmitted on the data transmission line from the disk drive card gate to the write circuits located on the actuator arm. For each change on the data transmission line, a 0-bit or 1-bit is written on the disk. These changes cause the current to be switched in the data head, which results in a change of magnetic direction on the disk.

*Reading Data Bits*

A voltage controlled oscillator generates the read clock when data is read from the disk drive. The oscillator is synchronized during the 16-byte gap field preceding the data field. The phase lock oscillator loop then keeps the oscillator in sync with the read data (1-bits and 0-bits) so the data can be decoded.



## DATA PROTECTION

### Data Unsafe Conditions

To prevent data from being destroyed by accident, there are circuits to protect data by not permitting data to be written under unsafe conditions. These unsafe conditions are described in *Error Conditions* later in this section.

When any of these unsafe conditions occur, write current is turned off and the data head select lines are de-activated. The disk drive sends an interrupt to the common adapter, and all commands from the common adapter (except sense commands) are ignored until the condition is reset.

### Not Ready

Data cannot be written when the disk drive is not ready.

The 'not ready' latch is activated by the following:

- A seek time-out has occurred.
- The servo clock counter has lost synchronization.
- The 'illegal actuator move' line is active.

For more information, see *Error Conditions* later in this section.

### Power On and Power Off

During power on, the actuator is held against the inner stop until the disk is up to speed.

A correct power on will be followed within 25 seconds by an interrupt from the disk drive after the heads have moved to home (cylinder 0 head 1).

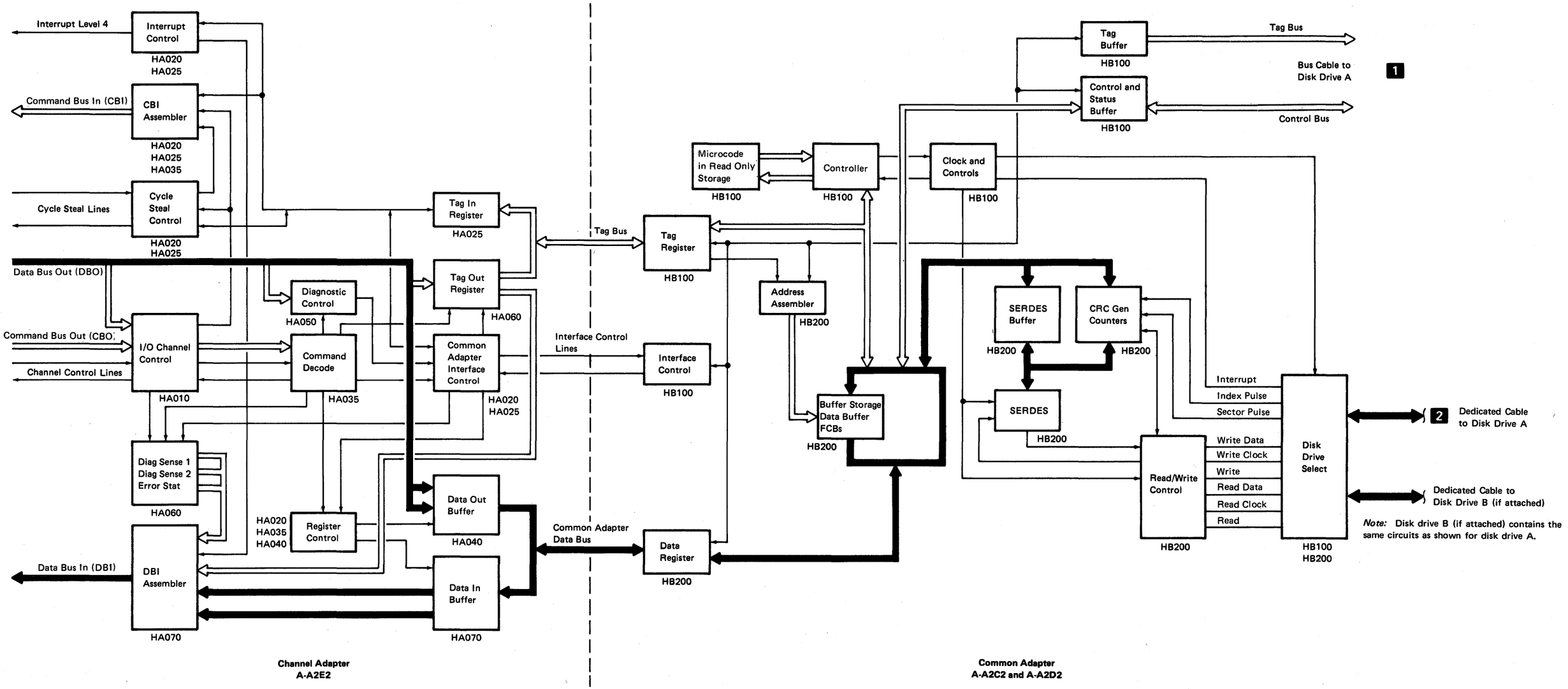
During a normal power off, the actuator is moved by a spring to the inner stop where it is held by a magnet. This locates the heads over the landing zone. The spindle brake is applied when its hold voltage is removed.

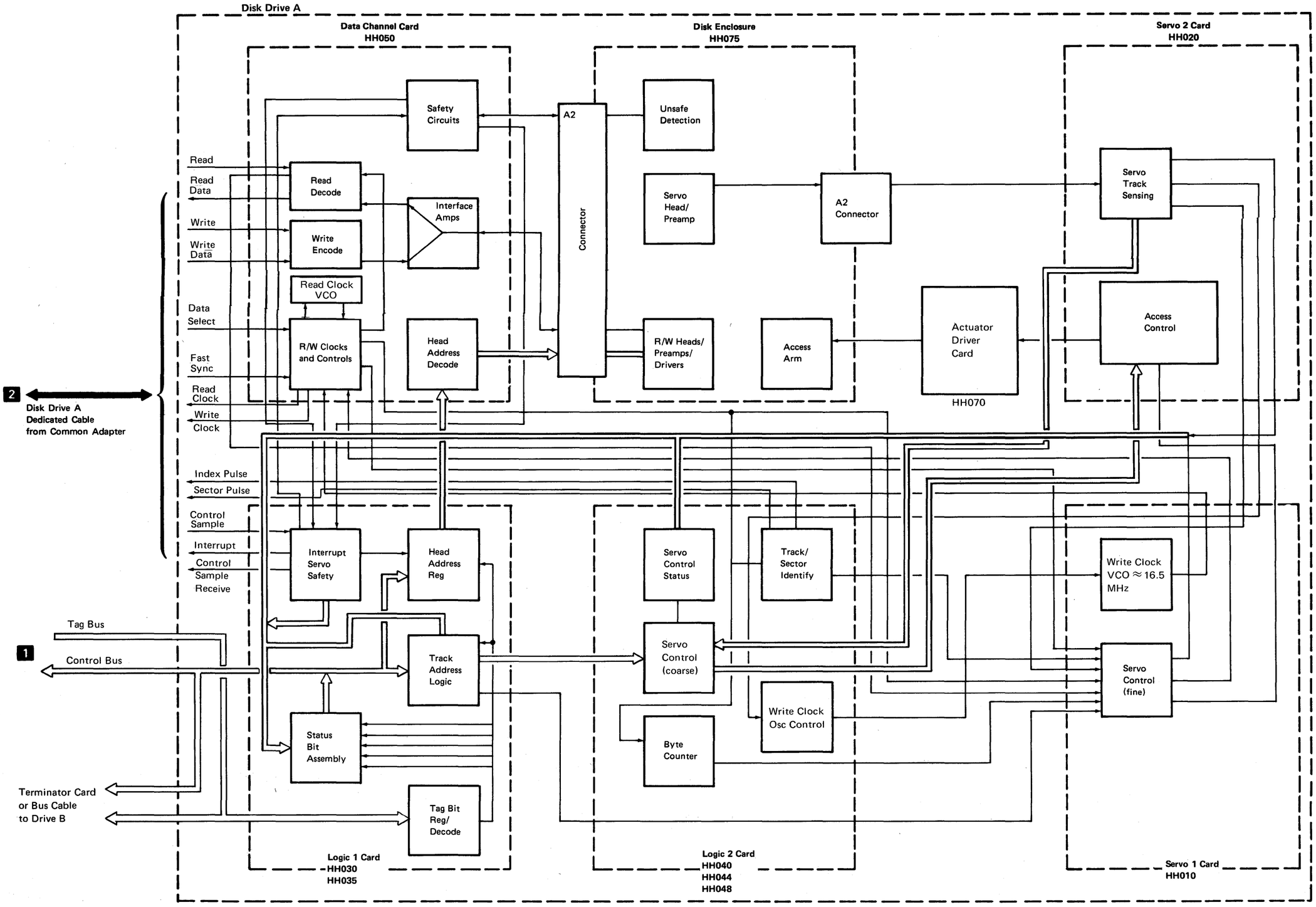
If a DC supply goes outside its limits, the system immediately removes all AC and DC voltages to decrease the possible loss of data. The actuator is moved to the inner stop as during a normal power off.

## DATA FLOW

The data flow in the disk attachment occurs across three interfaces:

- System I/O channel interface
- Common adapter interface
- Disk drive interface




Disk Drive A
Data Channel Card
HH050
Disk Enclosure
HH075
Servo 2 Card
HH020
Safety Circuits
A2
Unsafe Detection
Read
Read Data
Read Decode
Interface Amps
Servo Head/ Preamp
A2 Connector
Servo Track Sensing
Write
Write Data
Write Encode
Connector
Read Clock VCO
Data Select
R/W Clocks and Controls
Head Address Decode
R/W Heads/ Preamps/ Drivers
Access Arm
Actuator Driver Card
HH070
Access Control
Fast Sync
Read Clock
Write Clock
2
Disk Drive A Dedicated Cable from Common Adapter
Index Pulse
Sector Pulse
Control Sample
Interrupt
Control Sample Receive
Interrupt Servo Safety
Head Address Reg
Servo Control Status
Track/ Sector Identify
Write Clock VCO ≈ 16.5 MHz
Tag Bus
1
Control Bus
Track Address Logic
Servo Control (coarse)
Servo Control (fine)
Status Bit Assembly
Write Clock Osc Control
Byte Counter
Terminator Card or Bus Cable to Drive B
Tag Bit Reg/ Decode
Logic 1 Card
HH030
HH035
Logic 2 Card
HH040
HH044
HH048
Servo 1 Card
HH010

## System I/O Channel Interface

I/O instructions start the operations by using the DBO, the CBO, and the 'control out', 'service out', 'MPXPO strobe', and 'service in' control lines. Data is moved on the DBO for a write and on the DBI for a read. Cycle steal is used to move the data to and from the disk attachment. The 'block processor clock' and 'disk burst mode' lines are used to control the cycle steal operations.

## Common Adapter Interface

Control information on the CA (common adapter) data bus starts the operations. The tag lines specify the information on the CA data bus and the 'request out', 'acknowledge request out', and 'strobe out' lines control the movement of this information.

Data is also moved on the CA data bus. The tag lines specify the direction of the data movement and the 'CA burst mode', 'request in', 'acknowledge request in', 'strobe in', and 'strobe out' lines control the movement of this data.

### State Counter

The common adapter uses a 3-bit state counter to control data operations on the interface to the channel adapter. The state counter can change only 1 of its 3 bits each time it advances. The state counter starts at 0 and is advanced by the conditions on the interface lines and by internal timing pulses. The state counter is shown on timing charts later in this section to aid in understanding the logic, but the lines of the state counter cannot be probed.

### Controller

The common adapter has a controller which starts and executes all disk operations that are received from the control processor. The controller receives instructions from read-only storage (ROS), and uses a 9-bit controller bus and other internal control lines to send data or commands to a disk drive, to the channel adapter, or to buffer storage.

### Buffer Storage

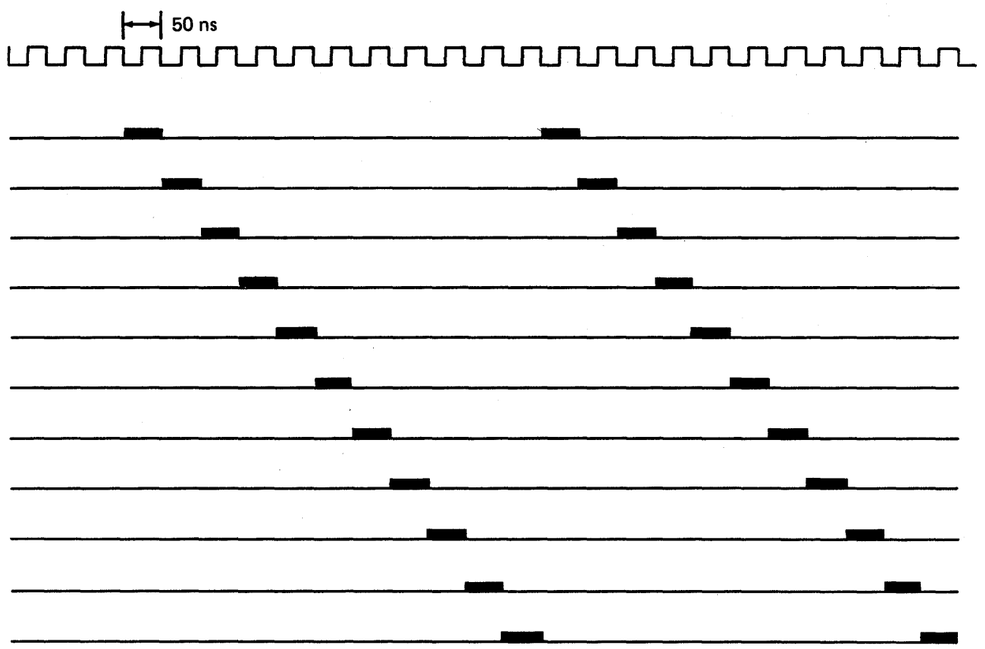
The common adapter buffer storage supplies temporary storage for data, commands, and status information. The buffer permits the controller to sense, to update, and to change the information while performing the control functions required for a specific operation. The buffer storage is divided into the following areas:

- ID compare field: This 4-byte field contains the ID of the sector for this operation. The contents of the ID compare field are compared to the IDs that are read from the disk. If the IDs are the same, an ID hit occurs.
- Controller work area: This 124-byte area is used by the controller as a work area.
- File control block(s): This area contains a 32-byte file control block (FCB) for each disk drive attached to the system. The FCB is used to store status and command information that is received from the control processor.
- Data buffers: These three 256-byte fields are used to store data that was read from or is to be written on the disk.

### Oscillator and Clocks

The common adapter has a 20-megahertz oscillator that generates eleven 50-nanosecond timing pulses. These timing pulses are used to sequence internal common adapter functions. Access to buffer storage is permitted during each sequence of timing pulses.

| Test Point | FSL Page | Line Name |
|---|---|---|
| TC Z33 | HD525 | 20-MHz Oscillator |
| TC W12 | HD525 | +T0 |
| TC X02 | | +T1 |
| TC W33 | HD525 | +T2 |
| NP | HD525 | +T3 |
| TC W32 | HD525 | +T4 |
| TC W28 | HD525 | +T5 |
| C2 S07 | HD525 | +T6 |
| TC Y11 | HD525 | +T7 |
| NP | HD525 | +T8 |
| TC W30 | HD525 | +T9 |
| NP | HD525 | +TA |




TC = Top card connector
C2 = A-A2C2 card
NP = Cannot be probed

## Disk Drive Interface

The disk drive interface has two parts. The first part is used for access and sense commands and contains:

- A 9-bit control bus, which moves track addresses to the disk and senses status information from the disk.
- Four tag lines, which indicate the direction of the data and the type of data on the control bus.
- Two control lines (the 'control sample' and 'control sample received' lines), which control the movement of this data.

All of these lines, except for the 'control sample' line, are in the bus cable to the disk.

The second part of the disk drive interface is used for reading and writing data. The lines used to read and write data are located in the dedicated cable to the disk.

## Write Data Flow

The data for a write operation is sent to the channel adapter on the DBO, and is loaded into the channel adapter data out buffer. Because the common adapter data bus is a 16-bit bus, 2 bytes of data from the DBO must be loaded into the data out buffer before data can be moved to the common adapter. In the common adapter, the data word is stored in the data register until it can be moved into buffer storage. The data is moved in blocks of 128 words, or 256 bytes, where each block is the data for one record on the disk. Data blocks may be moved immediately following each other until all three data buffers in buffer storage are full. When an ID hit occurs for the first sector to be written on the disk, the data is moved from buffer storage to the SERDES buffer, and then moved to the SERDES, where the data is serialized. The write data is moved to the disk drive under the control of the write clock. The data is written on the disk in synchronization with the write clock, which is controlled by the servo clock pulses on the dedicated servo surface.

## Read Data Flow

When the 'read' line is active, the read clock is used to decode the data from the analog signals received from the disk. The read data circuits decode the pulses and send the serial data to the common adapter. This data is converted to word length in the SERDES and moved to the SERDES buffer until it can be moved into buffer storage. When a full sector has been moved to buffer storage, the common adapter starts to move the data in burst mode to the channel adapter, which starts the cycle steal operation when the 'disk burst mode' line becomes active. The data is converted to byte length in the channel adapter data in buffer, and is sent to the control processor on the DBI.

## OPERATIONS

The disk drive and attachment perform the following data operations:

- Read ID
- Read data
- Read diagnostic
- Read verify
- Write ID
- Write data
- Scan equal
- Scan high or equal
- Scan low or equal

In addition, the special operations for diagnostic purposes are:

- Load seek required address direct
- Load seek control direct
- Sense file direct
- Sense file diagnostic sense bytes 1, 2, and 3
- Sense file diagnostic sense bytes 1, 2, and 3 256 sample test
- Sense file diagnostic wrap
- Disk speed timing diagnostic
- Timed seek diagnostic
- Read data with data repeat
- Read common adapter buffer storage

### Operation Sequence

For each command received from the control storage program, the disk attachment (channel adapter and common adapter) and the disk drive perform an operation or operations. The sequence for these operations is:

1. Each operation is started by loading the common adapter file control block (FCB) for the requested disk drive. The FCB contains all the information needed by the common adapter to complete the operation. For more information, see *Common Adapter File Control Block* and *Loading and Sensing the Common Adapter File Control Block* later in this section.

2. An adapter start is sent to the common adapter. This instructs the common adapter to execute the operation contained in the FCB.

3. The common adapter controller executes the adapter start if no higher priority internal command than the adapter start is requested. During the operation, the common adapter can send interrupts to the channel adapter and move data between the channel adapter, common adapter, and disk drive. For more information, see *Interrupts* later in this section.

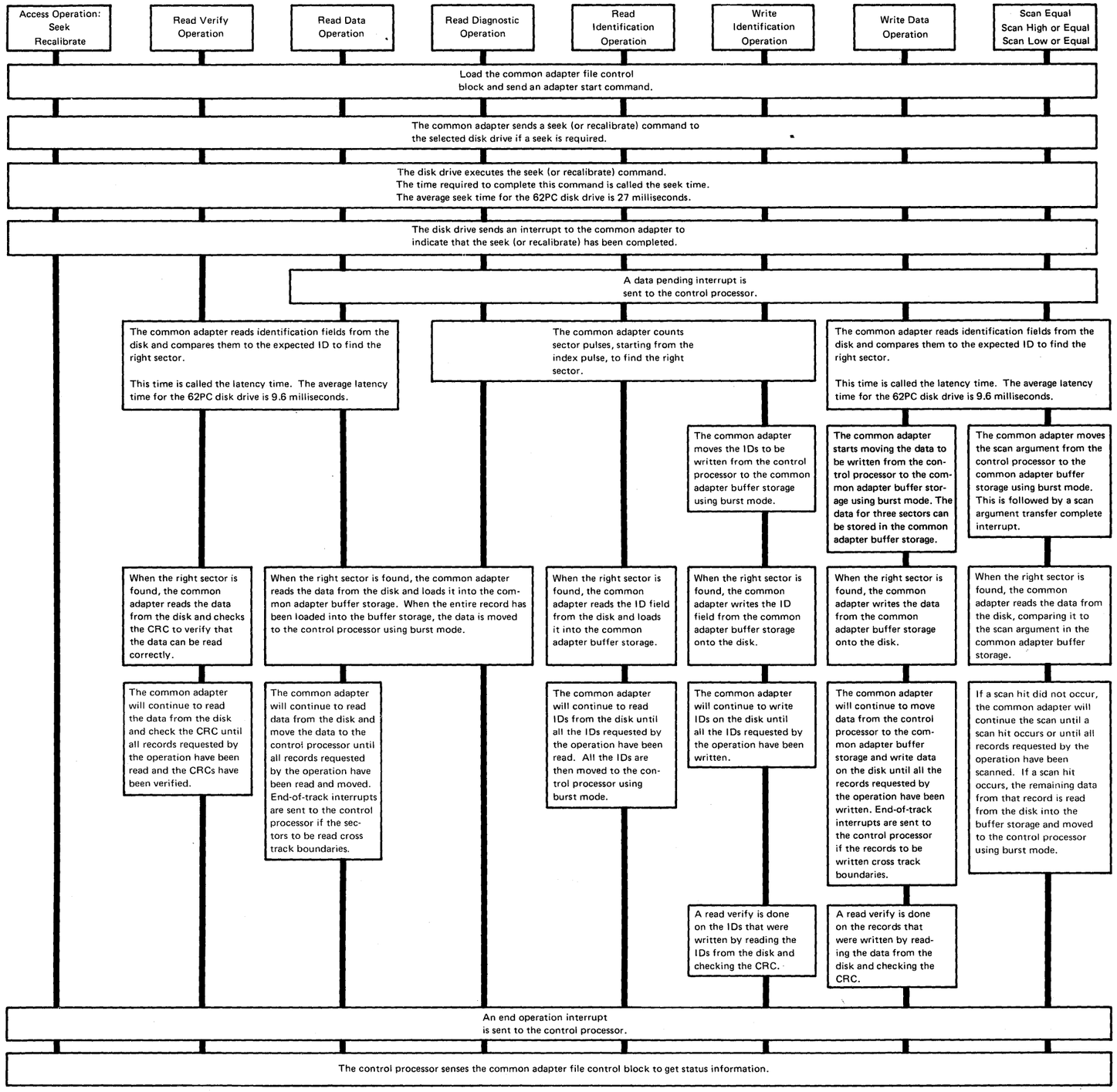
4. The common adapter indicates the operation is complete by sending an end operation interrupt to the control processor. If an error occurred, the operation may be terminated immediately and the end operation interrupt indicates an error condition. For more information, see *Interrupts* later in this section.

5. Sense and status information is obtained by sensing the FCB. For more information, see *Common Adapter File Control Block* and *Loading and Sensing the Common Adapter File Control Block* later in this section.

The following chart shows the sequence of operation for:

- Access (seek and recalibrate)
- Read verify
- Read data
- Read diagnostic
- Read ID
- Write ID
- Write data
- Scan (equal, high or equal, and low or equal)

## Operation Sequence



**Read ID**

The ID fields on the disk are read for the read ID, read data, write data, and scan operations, but are sent to the control processor only on a read ID operation. The CRC of the ID field is checked every time the ID field is read, but a CRC error on an ID field is set only on a read ID operation. When ID fields are read during a read data, write data, or scan operation, a CRC check in the ID field of the desired sector causes a no-record-found error.

The read ID operation can read up to 33 IDs (one complete track). The record number or sector number in the FCB indicates the sector to be read starting from the index pulse. Because of the head switch delay, the sector following the index is not always sector 0 (for more information, see *Sectors* earlier in this section). If an error occurs during a read ID operation, the operation stops at the sector in error.

The read ID special operation is used to read IDs without stopping on the error. An ID of hexadecimal FFFFFFFF is moved to the control processor to indicate a sector with an error.

The read ID extended operation is used if the ID field was written extended (64 bytes later than a normal ID) because of a failure on the disk at the normal ID location.

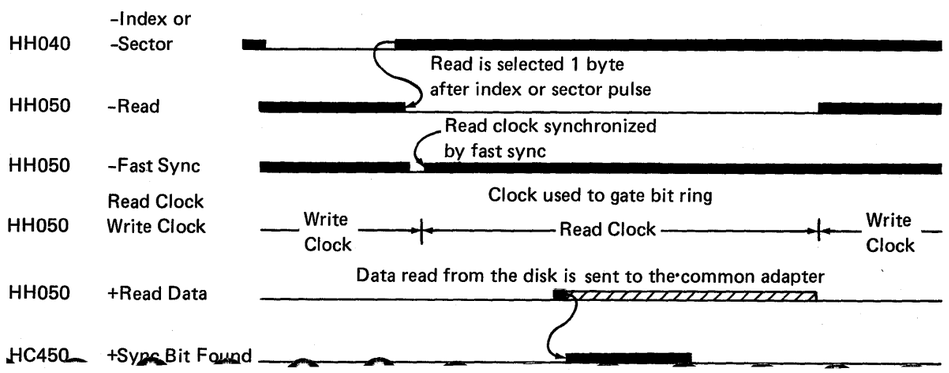
The command byte for a read ID operation is:

| | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Read ID | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| Read ID Special | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| Read ID Extended | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

*Note:* If bit 12 is set to 1, the common adapter does not automatically perform a seek.

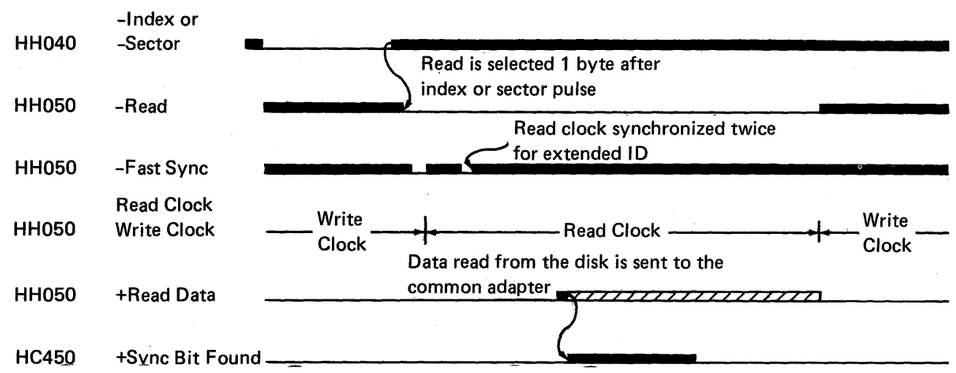
**Read ID**

| | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) |
|---|---|---|---|---|---|---|
| Format | | | | | | |
| Number of Bytes | 22 | 14 | 1 | 4 | 2 | 16 |
| Sequence Counter | 0357 0 | 1 | 2 | | | 3 |
| | Fine adjust of data head over track | Sync clock | Look for sync bit | Read ID and compare with ID compare field in buffer storage | Read CRC and compare with CRC register | Read circuit recovery |




**Read ID Extended**

| | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) |
|---|---|---|---|---|---|---|
| Format | | | | | | |
| Number of Bytes | 22 | 78 | 1 | 4 | 2 | 16 |
| Sequence Counter | 0357 0 | 1 | 2 | | | 3 |
| | Fine adjust of data head over track | Sync clock | Look for sync bit | Read ID and compare with ID compare field in buffer storage | Read CRC and compare with CRC register | Read circuit recovery |



### Read Data

The read data operation reads 256-byte data records from the data fields on the disk. One read data operation may read up to 256 consecutive records. The common adapter automatically performs seek operations to place the actuator at the correct track (if bit 12 of the command byte is 0). If the addressed sector has an alternative sector assigned, the common adapter performs a seek to the alternative cylinder and finds the alternative sector. The alternative sector is read and the actuator is returned to the original track if there are more sectors to be read by this operation. If a CRC check occurs, the operation ends with the sector in error, but that data is still moved to the control processor.

### Read Diagnostic

The read diagnostic operation reads data from the disk without an identification field compare. The common adapter counts sector pulses, starting from the index, to find the correct sector.

In the read data operation, the data fields from a sector can be read only if an ID hit occurs. If an identification field becomes damaged after the data is written, an ID hit might not occur. The read diagnostic operation can then recover the data. This operation moves the data read from the disk to the control processor even if an error occurs during the read operation.

### Read Verify

The read verify operation is the same as the read data operation, except that the data read from the disk is not moved to the control processor. This operation is used to verify that the data record being read from the disk can be read without an error.

The command bytes for read data, read diagnostic, and read verify operations are:

| | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Read Data | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| Read Diagnostic | 0 | 1 | 0 | 1 | 0 | 1 | 1 | DF |
| Read Verify | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |

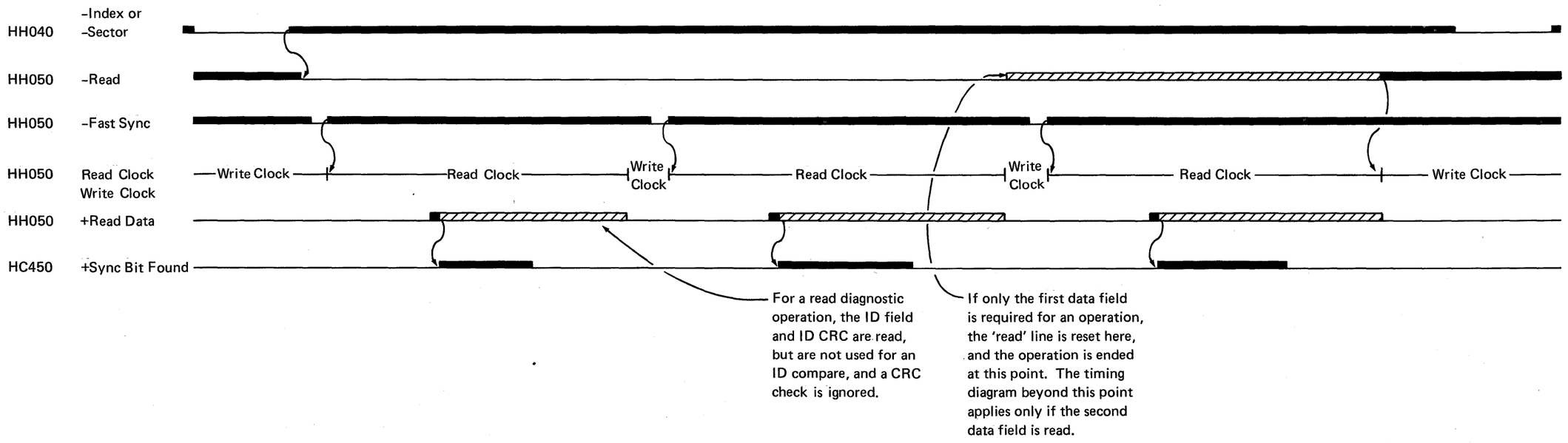
DF = Data field
0 = data field 1
1 = data field 2

This figure shows the reading of one or two data records from a sector on the disk. If only the second data field is to be read for an operation, the common adapter will read both data fields, but will not move the first record into buffer storage.

These timings apply to read data, read verify, read diagnostic, and scan operations.

**Read Data/Read Verify/Read Diagnostic/Scan High/Low/Equal**

| Format → | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) | Sync (hex 01) | Data Field 1 | Data CRC | Gap (hex 00) | Sync (hex 01) | Data Field 2 | Data CRC | Gap (hex 00) | Sample Servo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Number of Bytes → | 22 | 14 | 1 | 4 | 2 | 16 | 1 | 256 | 2 | 16 | 1 | 256 | 2 | 7 | 22 |
| Sequence Counter → | 0357 \| 0 | 1 | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | |
| | Fine adjust of data head over track | Sync clocks | Look for sync bit | Read ID and compare with field in FCB | Read CRC and compare with CRC register | Sync clocks | Look for sync bit | Read data from disk and store in buffer storage | Read CRC and compare with CRC register | Sync clocks | Look for sync bit | Read data from disk and store in buffer storage | Read CRC and compare with CRC register | Read circuit recovery | Fine adjust of data head over track |



### Write ID

The write ID operation writes identification fields in sectors of the disk. This operation is used when a permanent error is found in a sector and the identification field must be written again to indicate that the sector is damaged, and to identify the location of the alternative sector to which the data has been moved. If the damaged area is within the identification field, the identification field can be written 64 bytes later than a normal ID, so that the ID will not be written on the damaged area (write ID extended).

The write ID operation can write up to 33 IDs (one complete track). The record number or sector number in the common adapter file control block (FCB) indicates the first sector to be written, starting from the index. If an error occurs during a write ID operation, the operation ends and that sector is indicated in the record or sector number byte of the FCB.

The command byte for a write ID operation is:

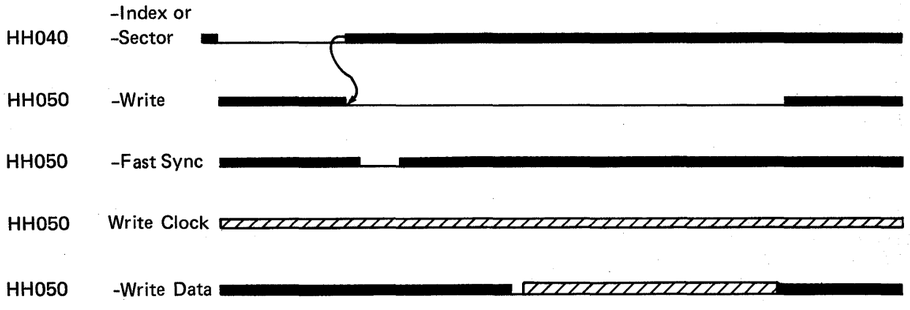
| | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Write ID | 0 | 1 | 1 | 0 | 0 | 1 | 0 | RV |
| Write ID Extended | 0 | 1 | 1 | 0 | 0 | 1 | 1 | RV |

RV = read verify

*Note:* If bit 12 is set to 1, the common adapter does not automatically perform a seek.

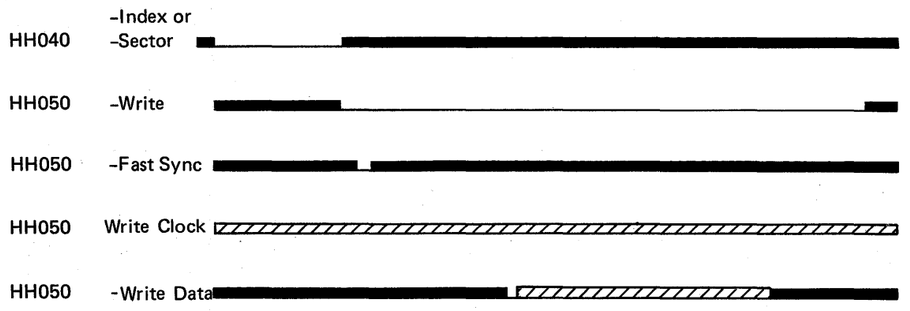
**Write ID**

| | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) |
|---|---|---|---|---|---|---|
| Number of Bytes | 22 | 14 | 1 | 4 | 2 | 16 |
| Sequence Counter | 0357 | 0 | 1 | 2 | | 3 |
| | File adjust of data head over track | Write zeros and sync clock | Write sync bit (hex 01) | Write ID and generate CRC | Write CRC | Write circuit recovery |




**Write ID Extended**

| | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) |
|---|---|---|---|---|---|---|
| Number of Bytes | 22 | 78 | 1 | 4 | 2 | 484 |
| Sequence Counter | 0357 | 0 | 1 | 2 | | 3 |
| | File adjust of data head over track | Write zeros and sync clock | Write sync bit (hex 01) | Write ID and generate CRC | Write CRC | Write circuit recovery |



## Write Data

The write data operation writes 256-byte data records to the data fields on the disk. Up to 256 consecutive records can be written by one write data operation. The common adapter automatically performs seek operations to place the actuator at the correct track (if bit 12 of the command byte is 0). If the addressed sector has an alternative sector assigned, the common adapter performs the seek to the alternative cylinder and finds the alternative sector. The alternative sector is written and the actuator returns to the original track if there are more sectors to be written by this operation. If a CRC check occurs, the operation ends with the sector in error.

A read verify is done after a write operation to ensure that the data is written correctly, but the operation can be modified so that the read verify is not done. This operation can also be modified to write the same data to each data record.

The command byte for a write data operation is:

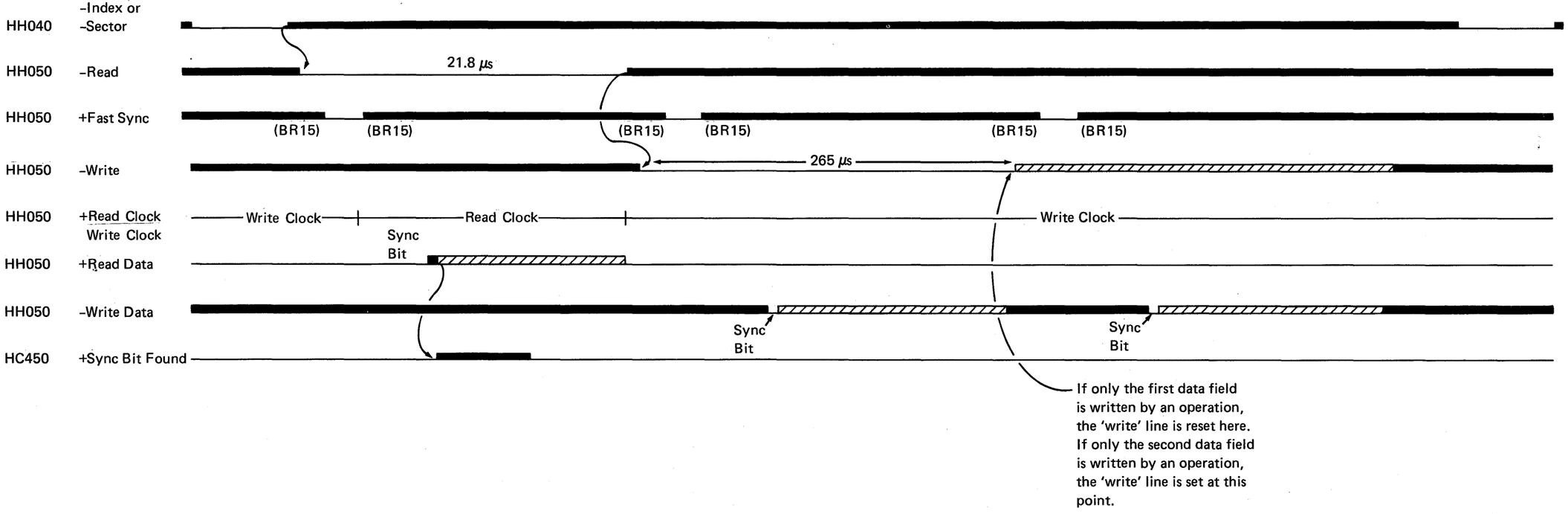
| | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Write Data | 0 | 1 | 1 | 0 | 0 | 0 | DR | RV |

DR = repeat
RV = read verify

This figure shows the writing of the first or both data fields to a sector on the disk. If only the second data field is to be written for an operation, a dummy read is done on the first data field (see read data timings), and then the indicated timings for data field 2 apply.

**Write Data**

| Format → | Sample Servo | Gap (hex 00) | Sync (hex 01) | ID F S H C | ID CRC | Gap (hex 00) | Sync (hex 01) | Data Field 1 | Data CRC | Gap (hex 00) | Sync (hex 01) | Data Field 2 | Data CRC | Gap (hex 00) | Sample Servo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Number of Bytes → | 22 | 14 | 1 | 4 | 2 | 16 | 1 | 256 | 2 | 16 | 1 | 256 | 2 | 7 | 22 |
| Sequence Counter → | 0357 0 | 1 | 2 | | | 3 | 4 | | | 5 | 6 | | | 7 | |
| | Fine adjust of data head over track | Sync clocks | Look for sync bit | Read ID field and compare with field in FCB | Read CRC and compare with CRC register | Sync clocks and write zeros | Write sync bit | Write data on disk and generate CRC | Write CRC on disk | Sync clocks and write zeros | Write sync bit | Write data on disk. Develop CRC | Write CRC on disk | Write circuit recovery | Fine adjust of data head over track |



**Scan Equal, Scan High or Equal, Scan Low or Equal**

These operations compare all or part of a data record on the disk to data from the control processor and determine if the condition tested for is present. The scan equal command tests for data on the disk being equal to data from the control processor; the scan high or equal tests for data on the disk being higher or equal to data from the control processor; and, the scan low or equal tests for data on the disk being lower or equal to data from the control processor.

The attachment uses a 256-byte scan field (from the control processor), which contains one or more compare fields, to do the compare. A compare field can be 1 to 256 bytes long, and is terminated by a hexadecimal FF, unless the compare field is 256 bytes long. When hexadecimal FF occurs in the compare field, or when the end of a 256-byte scan field is reached, a scan hit occurs if the scan condition has been met. When a scan hit occurs, any remaining data for that record is read from the disk and loaded into the common adapter buffer storage after a 2-byte delay. This 258-byte block of data (which includes the scan field up to the hexadecimal FF and the byte that follows it, and the data read from the disk) is then moved to the control processor.

If a scan hit does not occur during the hexadecimal FF, the scan is continued to compare data from the disk to any additional compare fields in the scan field. The scan continues until a scan hit occurs or the end of the 256-byte scan field is reached. A scan operation can be done on 1 to 256 sectors but, if a scan hit occurs, the operation ends with that sector.

The command bytes for the scan operations are:

| | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Scan Equal | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Scan High or Equal | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| Scan Low or Equal | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

The timings for scan operations are the same as for read operations.

## Diagnostic Operations

The operations shown in the following chart are used for diagnostic purposes by the diagnostic control program (DCP).

| Diagnostic Operation | Common Adapter Command Bytes 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|
| Load seek required address direct | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Load seek control direct | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Sense file direct | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Sense file diagnostic sense bytes 1, 2, and 3 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| Sense file diagnostic sense bytes 1, 2, and 3 256 sample test | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| Sense file diagnostic wrap | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Disk speed timing diagnostic | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| Timed seek diagnostic | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| Read data with data repeat | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| Read common adapter buffer storage | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |

**Load Seek Required Address Direct**

**Load Seek Control Direct**

These two operations perform the two steps that the common adapter completes to send a seek command to the selected disk drive. The common adapter sets up the seek control word the same way as for an access operation. For more information, see *Access Operations* later in this section.

The load seek required address direct operation moves the low byte of the seek control word to the disk drive across the control bus. When this operation is complete, the common adapter sends an end operation interrupt to the control processor. This operation does not cause the disk drive to start the seek.

The load seek control direct operation moves the high byte of the seek control word to the disk drive across the control bus. When this operation is complete, the common adapter sends an end operation interrupt to the control processor. This operation causes the disk drive to start the seek, but the end operation interrupt is sent before the seek operation is completed.

**Sense File Direct**

This operation causes the common adapter to perform a sense of the selected disk drive, similar to the sense performed following an interrupt. After the sense byte is moved to the low byte of FCB word 6, an end operation interrupt is sent to the control processor. For more information, see *Access Operations* later in this section.

**Sense File Diagnostic Sense Bytes 1, 2, and 3**

This operation causes the common adapter to sense the 3 diagnostic sense bytes from the disk drive. The common adapter also senses these bytes when a disk drive error occurs during an operation.

These 3 bytes are used by the disk MDI MAPs to isolate disk problems and are also stored in ERAP.

**Sense File Diagnostic Sense Bytes 1, 2, and 3 256 Sample Test**

This operation senses the same 3 bytes as the sense file diagnostic sense bytes 1, 2, and 3 operation, but quickly senses each byte 256 times. Each byte is compared to determine if any of the bits changed during the test. Each bit that changed is set to 1 in FCB words 13 and 14 (high byte only), and an end operation interrupt is sent to the control processor.

These 3 bytes are used by the disk MDI MAPs to isolate disk problems.

**Sense File Diagnostic Wrap**

This operation causes the common adapter to perform a control bus sense with a disk drive tag of 3. The disk drive sends the low byte of the seek control word to the common adapter. The common adapter moves the byte to the low byte of FCB word 14 and sends an end operation interrupt to the control processor. The timing of this operation is the same as for a sense operation. For more information, see *Access Operations* later in this section.

This test is used with the load seek required address direct operation to wrap data to the disk drive and back, to test the control bus.

**Disk Speed Timing Diagnostic**

This operation causes the common adapter to time 20 revolutions of the disk and put the results in FCB word 13. The time can be determined by multiplying the value in FCB word 13 by 22.6 microseconds.

**Timed Seek Diagnostic**

This operation is the same as the seek operation, except that the controller times the seek. The results are put in FCB word 13. The time can be determined by multiplying the value in FCB word 13 by 22.6 microseconds.

**Read Data with Data Repeat**

This operation is the same as the read data operation, except that only the last data record read from the disk is sent to the control processor.

**Read Common Adapter Buffer Storage**

This operation reads the contents of the common adapter buffer storage. Data is moved to the control processor the same way as in a read data operation. The buffer storage is divided into four parts, and the part that is read is determined by bits 6 and 7 of the record or sector count of the FCB as shown below:

| Bit 6 | 7 | Part Read |
|---|---|---|
| 0 | 0 | File control block area and controller work area |
| 0 | 1 | Data buffer 1 |
| 1 | 0 | Data buffer 2 |
| 1 | 1 | Data buffer 3 |

## Common Adapter File Control Block

The file control blocks (FCBs) are located in the common adapter buffer storage. Each FCB contains 16 words of 2-byte length. The FCB is divided into two parts: the first six words are for command data, and the remaining 10 words are for sense and status data. The control processor moves the command data to the common adapter using the disk attachment commands (see *Commands* later in this section). The FCB is updated with the current disk status by the common adapter at the end of each sector, so that the FCB will indicate the next sector to be operated on or when the operation is complete. Operation status and error information is obtained, at the end of the operation, by sensing this FCB using the disk attachment commands.

| File Control Block Word | High Byte | Low Byte | |
|---|---|---|---|
| 0 | (reserved) | Command byte | Command Data |
| 1 | Record sector count | Flag byte | |
| 2 | Cylinder number | | |
| 3 | Head byte | Record/sector number | |
| 4 | (reserved) | (reserved) | |
| 5 | (reserved) | (reserved) | |
| 6 | File status word | | Sense and Status Data |
| 7 | Error sense word | | |
| 8 | (not used) | | |
| 9 | (not used) | | |
| 10 | Current head and cylinder number | | |
| 11 | Previous head and cylinder number | | |
| 12 | Interrupt status word | | |
| 13 | File diagnostic sense bytes 1 and 2 | | |
| 14 | File diagnostic sense byte 3 | Diagnostic wrap byte | |
| 15 | (not used) | | |

*Command Byte:* This byte indicates the operation that is to be performed. The operation and command byte bit definitions are shown earlier in this section. The following table shows the command bytes for normal operations.

| Operation | Common Adapter Command Bytes 8 | 9 | 10 | 11 | *12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|
| Seek | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Recalibrate | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Read Data | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| Read verify | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| Read ID | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| Read ID extended | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Read diagnostic | 0 | 1 | 0 | 1 | 0 | 1 | 1 | DF |
| Write data | 0 | 1 | 1 | 0 | 0 | 0 | DR | RV |
| Write ID | 0 | 1 | 1 | 0 | 0 | 1 | 0 | RV |
| Write ID extended | 0 | 1 | 1 | 0 | 0 | 1 | 1 | RV |
| Scan equal | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Scan low or equal | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| Scan high or equal | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |

DF = data record (0 = record 1, 1 = record 2)
DR = data repeat
RV = read verify
*If bit 12 = 1 the automatic seek will be inhibited.

*Record/Sector Count:* This byte contains one less than the number of records or sectors to be operated on by the command specified in the command byte. The read data, read verify, write data, and scan commands can operate on 1 to 256 records, so the record/sector count can range from 0 to 255 (hexadecimal 0 to FF). For read and write identification commands, the record/sector count must be between 0 and 32 (hexadecimal 0 and 20). The record/sector count must be 0 for a read diagnostic command. If an operation ends with an error, this byte will contain the number of records or sectors that have not been operated on.

*Flag Byte:* This byte is used for the ID field compare for a read data, read verify, write data, or scan operation. (The flag byte bit definition is given in *Sector* under *Disk Format* earlier in this section.) If common adapter alternative sector processing is used for an operation, the common adapter will not use this flag byte, but will set up the expected flag byte for the alternative sector ID field compare.

*Cylinder Number:* This word contains the number of the cylinder that the operation is to access. This number is also used to generate the expected ID field if an ID compare will be done. This number can range from 0 to 359 (hexadecimal 0 to 167), and uses only bits 7 through 15; therefore, bits 0 through 6 must be zero.

| | Bits 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FCB Word 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C | C | C | C | C | C | C | C | C |

*Head Byte:* This byte contains the number of the head that is to be used by the operation. This number is also used to generate the expected ID field if an ID compare will be done. Because there are 11 heads, this number can range from 0 to 10 (hexadecimal 0 to A). Only bits 4 through 7 are used, and bits 0 through 3 must be zero.

| | Bits 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| FCB Word 3 | 0 | 0 | 0 | 0 | H | H | H | H |

*Record/Sector Number:* For a read data, read verify, write data, or scan operation, this byte contains the starting record number that is to be operated on, which can range from 0 to 63 (hexadecimal 0 to 3F). For a read or write ID, or a read diagnostic operation, this byte contains the starting sector number that is to be operated on, which can range from 0 to 32 (hexadecimal 0 to 20). This number is used to generate the expected ID field if an ID compare will be done. Bits 8 and 9 must be zero.

| | Bits 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|
| FCB Word 3 | 0 | 0 | R | R | R | R | R | R |

*Note:* Error conditions indicated by the following sense and status data are given in more detail in *Error Conditions* later in this section.

*File Status Word:* The bits have the following meanings if they are on:

Bit 0 — An error occurred during the operation.

Bit 1 — Forced end operation. (A disk attachment command forced the operation to end.)

Bit 2 — The disk drive read/write hardware is in use.

Bit 3 — The common adapter alternative sector processing was used during the operation.

Bit 4 — Not used (zero).

Bit 5, 6, 7 — These bits are set by configuration jumpers on the common adapter channel interface card.

011 = 65 Mb disk drive attached
100 = Disk drive not attached

Bit 8 — Always on (one).

Bit 9 — Brake applied. (The disk brake is applied.)

Bit 10 — Track unavailable.
(The cylinder requested by the seek command was beyond cylinder 359.)

Bit 11 — 62PC command error.
(A parity error occurred on the disk drive interface during the access command to the disk.)

Bit 12 — Data unsafe.
(A data unsafe condition occurred during the operation.)

Bit 13 — Seek incomplete.
(Following an access command, a seek complete interrupt was not returned to the common adapter.)

Bit 14 — Home.
(The disk is home; cylinder 0 and head 1 selected.)

Bit 15 — Disk not ready.
(The disk is in a not ready condition.)

*Error Sense Word:* The bits have the following meanings if they are on:

Bit 0 — CRC check.
(The CRC generated from the ID or the data being read from the disk did not match the CRC read from the disk.)

Bit 1 — Common adapter parity check.
(A parity check occurred in the common adapter, as described by the low byte of common adapter file control block word 14:

- Bit 13=1 — Buffer storage to controller parity check occurred.
- Bit 14=1 — Data buffer to SERDES parity check occurred.
- Bit 15=1 — ROS parity check occurred.)

Bit 2 — Channel interface parity check.
(A parity check occurred on the common adapter interface, as described by the low byte of common adapter file control block word 14:

- Bit 11=1 — Tag bus parity check.
- Bit 12=1 — Data bus parity check.)

Bit 3 — Write gate return check.
(Write current was not sensed when write was selected.)

Bit 4 — No record found.
(The sector being addressed was not found after two index pulses and the sector did not have an alternative sector.)

Bit 5 — Not valid command parameters.
(The command words loaded into the common adapter file control block were not valid.)

Bit 6 — Missing sector pulse error.
(A missing sector pulse was sensed by the disk drive circuits during a read ID, write ID, or read diagnostic operation.)

Bit 7 — Time-out check.
(A time-out check occurred in the common adapter, as described by the low byte of common adapter file control block word 14:

- Bit 8=1 — Seek complete interrupt not received.
- Bit 9=1 — Common adapter read/write hardware time-out.
- Bit 10=1 — Code executed by the controller was not valid.)

Bit 8 — Disk drive not attached.
(The operation was sent to a disk that the configuration jumpers indicate is not attached. See bits 5, 6, and 7 of the file status word.)

Bits 9-12 — Not used (zero).

Bit 13 — End of disk.
(The read, write, or scan operation extended beyond the last data sector on cylinder 357.)

Bit 14 — Not used (zero).

Bit 15 — 62PC interface error.
An error occurred on the 62PC interface, as described by the low byte of common adapter file control block word 14:

- Bit 8=1 — 62PC cable continuity is open.
- Bit 9=1 — 62PC did not set control sample received.
- Bit 10=1 — A parity check occurred on incoming data on the 62PC control bus.
- Bit 11=1 — Disk interrupt did not reset.

*Current Head and Cylinder Number:* This word contains the seek control word that was sent to the 62PC disk drive during the latest seek command. The common adapter generates the seek control word from the head byte and cylinder number. The head byte and cylinder number are located in the word as shown:

| | Bits | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| FCB Word 2 | 0 | 0 | H | H | H | H | 0 | C | C | C | C | C | C | C | C | C |

*Note:* This word can be set to hexadecimal 8000 as a result of some disk drive or common adapter failures.

*Previous Head and Cylinder Number:* The current head and cylinder number (FCB word 10) is moved to the previous head and cylinder number (FCB word 11) when the common adapter sends a seek command to the disk.

*Interrupt Status Word:* The latest interrupt status word sent to the system is stored in the file control block. The bits have the following meanings if they are active:

Bit 0 — End operation interrupt.
(This interrupt indicates that the command in the FCB was completed.)

Bit 1 — End of track interrupt.
(This interrupt indicates that the read data or write data operation reached the end of a track.)

Bit 2 — Data pending interrupt.
(This interrupt indicates that the common adapter is ready to start moving data to or from the system.)

Bit 3 — Not used (zero).

Bit 4 — Scan field transfer complete interrupt
(This interrupt indicates that the scan field was moved to the common adapter, and that the common adapter is ready to move data back to the system if a scan hit occurs.)

Bit 5 — Any error.
(This bit is used during the end operation interrupt to indicate that an error occurred during the operation.)

Bit 6 — Not used (zero).

Bit 7 — Disk select bit.
(Bit 7=0 for disk drive A, bit 7=1 for disk drive B.)

Bit 8 — Scan not hit.
(Used during the end operation interrupt to indicate the status of a scan operation.)

Bit 9 — Scan equal hit.
(Used during the end operation interrupt to indicate the status of a scan operation.)

Bits 10-15 — Not used (zero).

*File Diagnostic Sense Bytes 1, 2, and 3, and Diagnostic Wrap Byte:* These words are sensed by the common adapter when a 62PC error occurs. These bytes are also used with diagnostic commands. The following table shows the meanings of the bytes for these diagnostic commands.

The only time the common adapter resets these words is on either a power on reset or a system reset.

| Diagnostic Command | Contents of Word 13 | Contents of Word 14 (see Note 1) |
|---|---|---|
| Sense file diagnostic sense bytes 1, 2, and 3 | Disk drive diagnostic sense bytes 1 and 2 (see Note 2). | Bits 0 through 7 contain disk drive diagnostic sense byte 3 (see Note 2). |
| Sense file diagnostic sense bytes 1, 2, and 3 256 sample test | The results of the 256 sample test of disk drive diagnostic sense bytes 1 and 2 (see Note 2). | The results of the 256 sample test of disk drive diagnostic sense byte 3 are set in bits 0 through 7 (see Note 2). |
| Sense file diagnostic wrap | | Bits 8 through 15 contain the wrap byte from the disk drive (see Note 2). |
| Disk speed timing | The timing for 20 disk revolutions. | |
| Timed seek | The timing for a seek. | |

*Notes:*
1. Bits 8 through 15 are also used to further define error conditions indicated in the error sense word (FCB word 7).
2. For the bit definitions of the disk drive diagnostic sense bytes and wrap byte, see *Bus Cable* under *Lines Between the Common Adapter and the 62PC Disk Drive* later in this section.

This page intentionally left blank.

## Loading and Sensing the Common Adapter File Control Block

The common adapter file control block is loaded and sensed by disk attachment commands. See *Commands* later in this section.

### Loading the Common Adapter File Control Block

To move a file control block word to the common adapter, three I/O commands are needed:

- A command to load the channel adapter tag register (IOCL 0). The channel adapter loads the data bus out (DBO) into the tag register when the 'service out' and 'strobe' lines are on while this command is executed. The tag byte is set up by some preceding commands, to indicate a load (bit 1=1), the file control block to be loaded (bit 3 selects disk drive A or B), and which word of the file control block is to be loaded (bits 4 through 7). The tag byte is then stored in an LSR, which this command must specify.

- A command to load the high byte of the file control block word to the channel adapter data out register[1] (IOL 0 or IOL 8). The channel adapter loads the data bus out (DBO) into the high byte of the data out register when the 'service out' and 'strobe' lines are on while this command is executed. The DBO must contain the high byte of the file control block word to be loaded.

- A command to load the low byte of the file control block word into the channel adapter data out register and to move both the high and low bytes into the common adapter[1] (IOL 1 or IOL 9). The channel adapter loads the data bus out (DBO) into the low byte of the data out register when the 'service out' and 'strobe' lines are on while this command is executed. The DBO must contain the low byte of the file control block word to be loaded. When the channel adapter resets the 'service in' line, the 'request out' line is set and the tag byte is sent to the common adapter on the tag bus. The common adapter responds to the command by setting the 'acknowledge request out' line. The channel adapter then sends the file control block (FCB) word to the common adapter, where the 'strobe out' pulse latches the FCB word. The common adapter stores the FCB word in buffer storage on the next available storage cycle.

The disk attachment can control operations on two disk drives. These operations can be overlapped. This makes it possible for data being moved to the common adapter to interrupt a load file control block sequence. If data cycle steal to the common adapter occurs after the high byte of a file control block word has been loaded and before the low byte has been loaded and sent to the common adapter, the data out register will contain cycle steal data instead of the high byte of the file control block word. If this occurred, the word will not be loaded correctly. To test for this condition, the channel adapter has a latch (the retry latch) that is reset when a load file control block high byte command is completed. This latch is set when the control processor sets the 'burst mode' line in response to the setting of the 'block processor clock' line by the disk attachment, indicating that cycle steal has occurred. After the load file control block low byte command has been completed, the condition of this latch is tested by a jump on I/O condition command (JIO 6) to determine if a cycle steal has occurred. If a cycle steal has occurred, the jump to load the word again is taken. If only one disk drive is installed, this condition cannot occur, so the jump is never taken.

The figures on the following pages show the flowchart, the logic, and the timing of the load file control block sequence.

After one word has been loaded, the LSR containing the tag byte is increased by 1 to select the next command data word to be loaded into the file control block. This is followed by a test to determine if the last command data word needed, word 3, has been loaded. If it has not, loop B is taken to load the next word. When all four command data words have been loaded, a start adapter command (using IOL 5) is sent to the common adapter. This command causes the common adapter to execute the operation specified by the command data words in the file control block.

### Sensing the Common Adapter File Control Block

The file control block words are usually sensed as they are needed. Three I/O instructions are used to sense a word from the common adapter:

- A command to load the tag register (IOCL 0). This is the same as for the load file control block sequence, except that the tag byte must indicate that a sense will be executed (bit 1=0).

- A command to sense the high byte of a file control block word[2] (IOS 2 or A). This command causes the channel adapter to set the 'request out' line to the common adapter and to send the contents of the tag register to the common adapter on the tag bus. The common adapter responds with the 'acknowledge request out' line and sets the 'buffer storage cycle request' line. The next available buffer storage cycle reads the requested file control block word from buffer storage and sends it to the channel adapter, where both bytes are loaded into the data in register with the 'strobe in' pulse. The 'strobe in' line also resets the 'request out' line. The channel adapter then sends the high byte of the file control block word to the control processor on the DBI during service in.

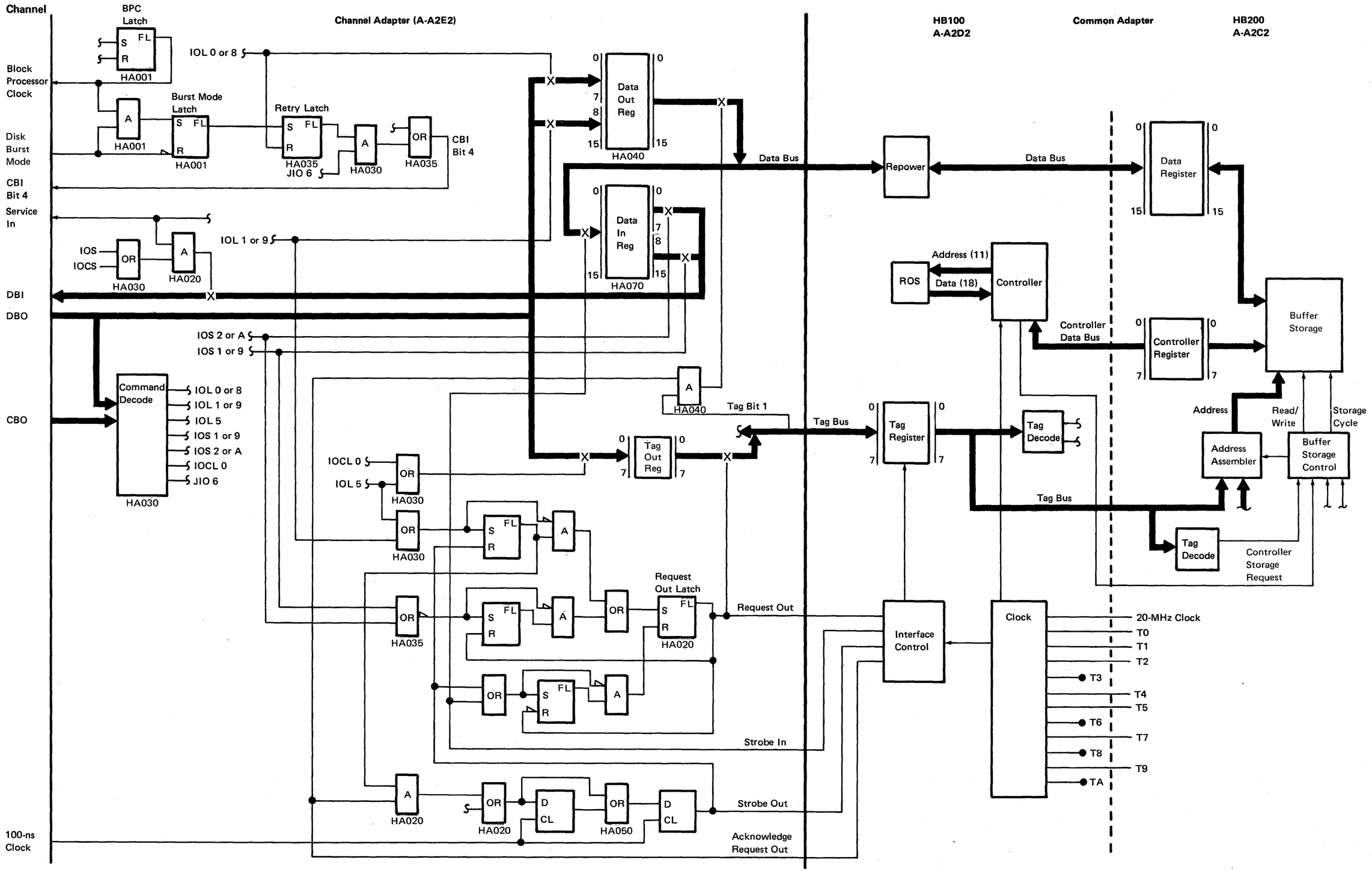
- A command to sense the low byte of a file control block word[2] (IOS 1 or 9). This command is the same as the sense file control block high byte, except that the low byte is sent to the control processor instead.

The timing for the sense file control block commands is given in *Common Adapter Commands* later in this section.

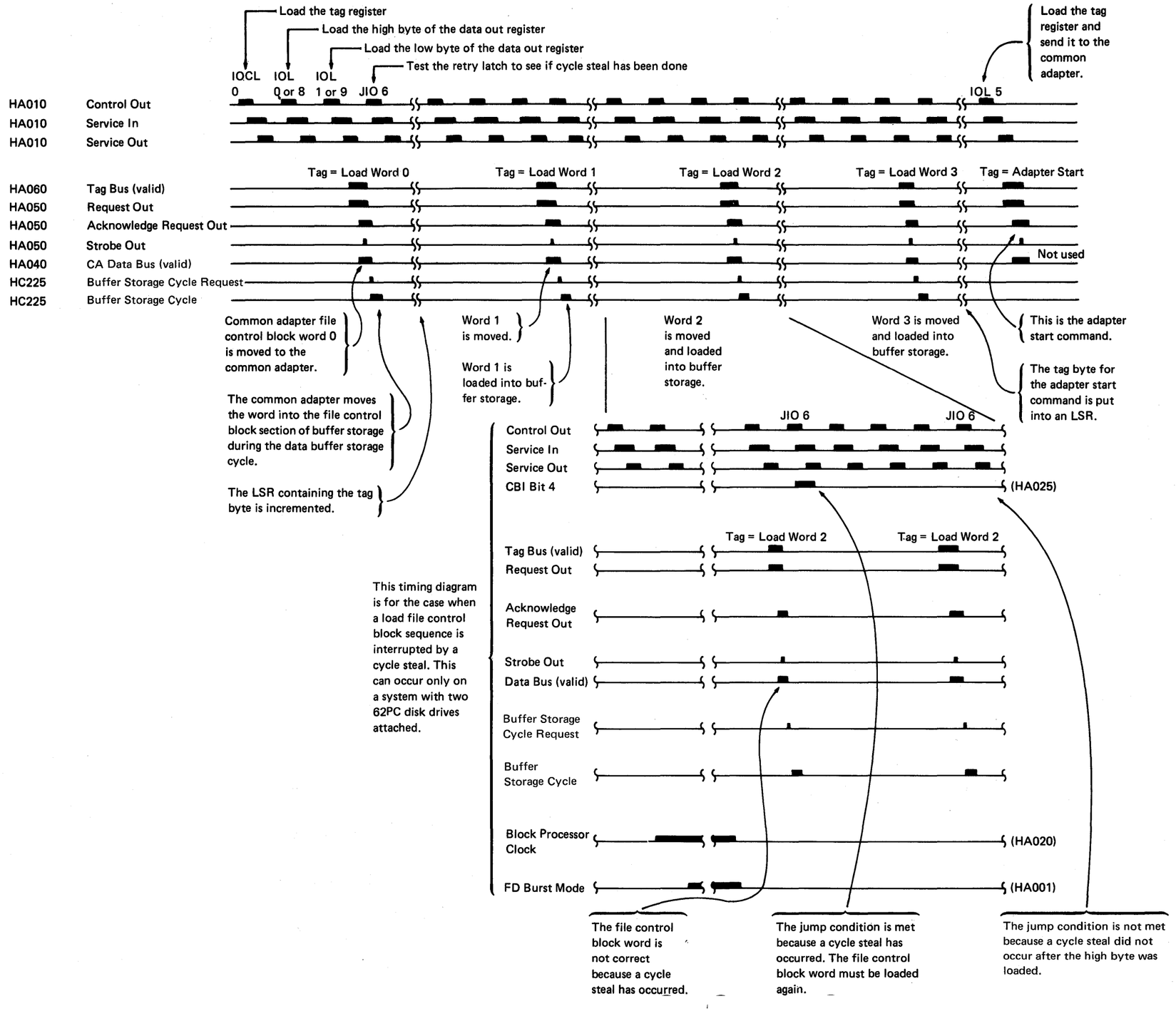
---

[1]WTCH and WTCL I/O storage commands are decoded as IOL commands by the channel adapter and are also used to load the file control block.

[2]RDCH and RDCL I/O storage commands are decoded as IOS commands by the channel adapter and are also used to sense the file control block.


Channel
BPC Latch
Channel Adapter (A-A2E2)
HB100 A-A2D2
Common Adapter
HB200 A-A2C2
Block Processor Clock
Disk Burst Mode
CBI Bit 4
Service In
DBI
DBO
CBO
HA001
Burst Mode Latch
Retry Latch
HA035 JIO 6
HA030
HA035
IOL 0 or 8
IOL 1 or 9
IOS
IOCS
HA020
Data Out Reg
HA040
Data In Reg
HA070
Data Bus
Repower
Data Register
Address (11)
Data (18)
ROS
Controller
Controller Data Bus
Controller Register
Buffer Storage
Command Decode
IOS 2 or A
IOS 1 or 9
IOL 5
JIO 6
IOCL 0
Tag Bit 1
Tag Bus
Tag Out Reg
Tag Register
Tag Decode
Address
Read/ Write
Storage Cycle
Address Assembler
Buffer Storage Control
Request Out Latch
Request Out
Strobe In
Strobe Out
Acknowledge Request Out
Interface Control
Clock
20-MHz Clock
T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
TA
Controller Storage Request
100-ns Clock

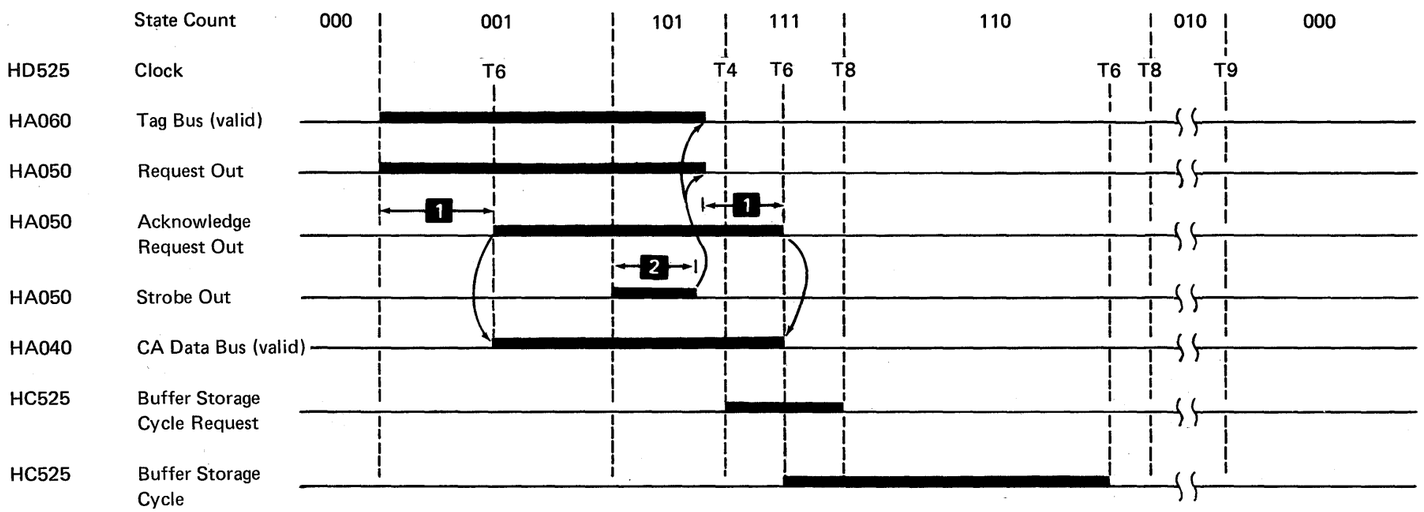
## Loading and Sensing the Common Adapter File Control Block (continued)



**Load FCB Command**

| State Count Changed by the Following Conditions | Clock | State Count | State |
|---|---|---|---|
| | | 000 | Idle, waiting for a command |
| The 'request out' line is active. | – | 00X<br>001 | Time delay |
| The 'strobe out' line is active and the 'tag bus' lines = write FCB. | – | X01<br>101 | Waiting for the next T4 |
| Wait for T4. | T4 | 1X1<br>111 | Requesting an FCB cycle |
| FCB cycle is active on the channel interface. | T8 | 11X<br>110 | Performing a write FCB cycle |
| FCB cycle is complete. | T8 | X10<br>010 | Waiting |
| End of load FCB command. Data transmission is complete. | T9 | 0X0<br>000 | End of command |

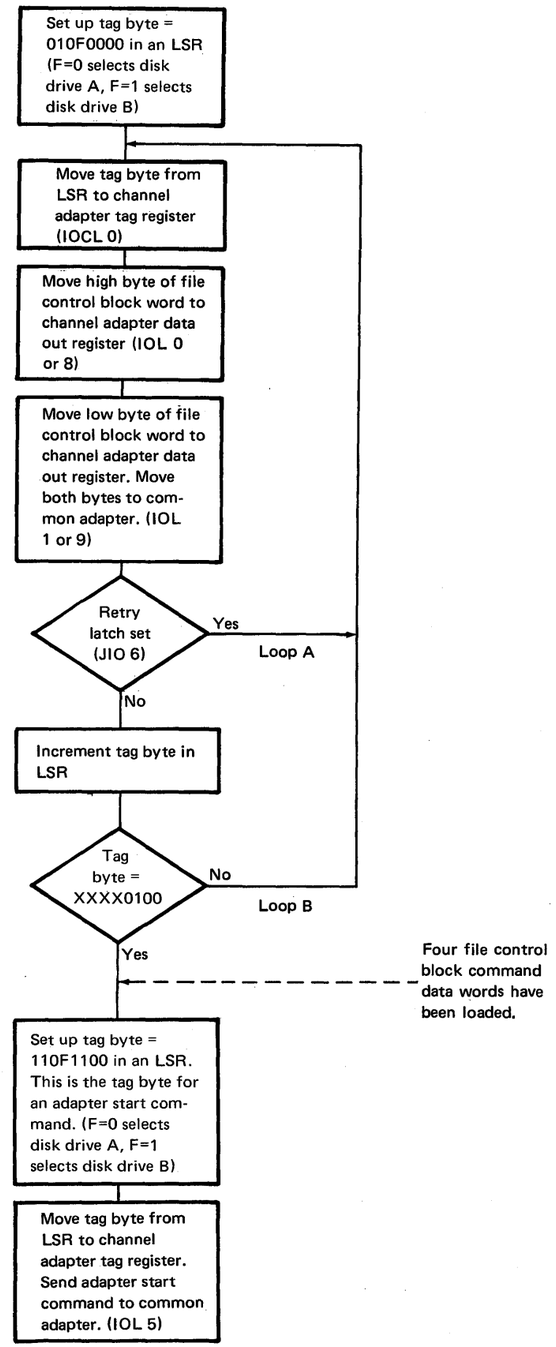
**Load FCB Command**




1 Timing = 1.5 μs maximum, 300 ns minimum

2 Timing = 200 ns

**Load File Control Block Sequence Flowchart**



## Interrupts

The disk attachment uses interrupt level 4 to indicate to the control processor the need for service. The interrupt control hardware is located in the channel adapter. If an interrupt condition occurs in the channel adapter, the 'interrupt request' latch will be set. If the 'interrupt enable' latch in the channel adapter is set, the 'interrupt level 4' latch is set, sending the interrupt to the control processor. The 'interrupt enable' latch is set and reset by disk attachment commands.

The interrupt control logic in the channel adapter is shown in the diagram. There are three ways that the 'interrupt request' latch can be set: an interrupt request from the common adapter, the condition that the 'data pending' latch is set and the 'channel busy' latch is not set, or specific channel adapter errors on the common adapter interface.

An interrupt request from the common adapter occurs when an operation has an interrupt condition. Common adapter interrupt conditions are:

- Data pending: The common adapter is ready to start moving data.
- Scan field transfer complete: The movement of the scan field to the common adapter has been completed.
- End of track: A read data or write data operation has come to the end of a track before completing the operation.
- End operation: The operation has been completed by the common adapter.

The common adapter sends the interrupt to the channel adapter by setting the 'request in' line and by setting an interrupt condition on the tag bus:

| Bits 0 1 2 3 P | |
|---|---|
| 0 0 0 F P | Interrupt request from the common adapter. |
| | Bit 3=0 Interrupt is for a disk drive A operation. |
| | Bit 3=1 Interrupt is for a disk drive B operation. |
| | Bit P Odd parity for bits 0 to 3. |

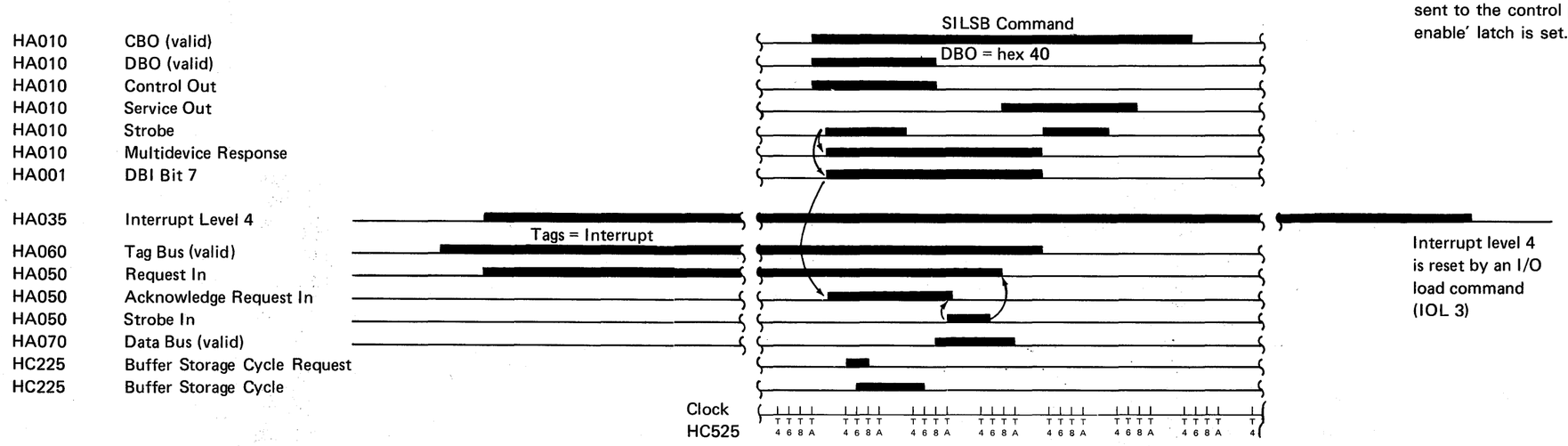
This interrupt condition causes the channel adapter to send an interrupt to the control processor if the 'interrupt enable' latch is set, but the channel adapter does not respond to the common adapter 'request in' line at this time. When a sense interrupt level status byte (SILSB) command is executed to determine which I/O devices are requesting service on interrupt level 4, the channel adapter sets bit 7 on the DBI to indicate a request. The SILSB command also causes the channel adapter to respond to the common adapter interrupt request, by setting the 'acknowledge request in' line and loading the interrupt word from the common adapter data bus into the channel adapter data in register with the 'strobe in' pulse from the common adapter. See *Interrupt Status Word* earlier in this section.

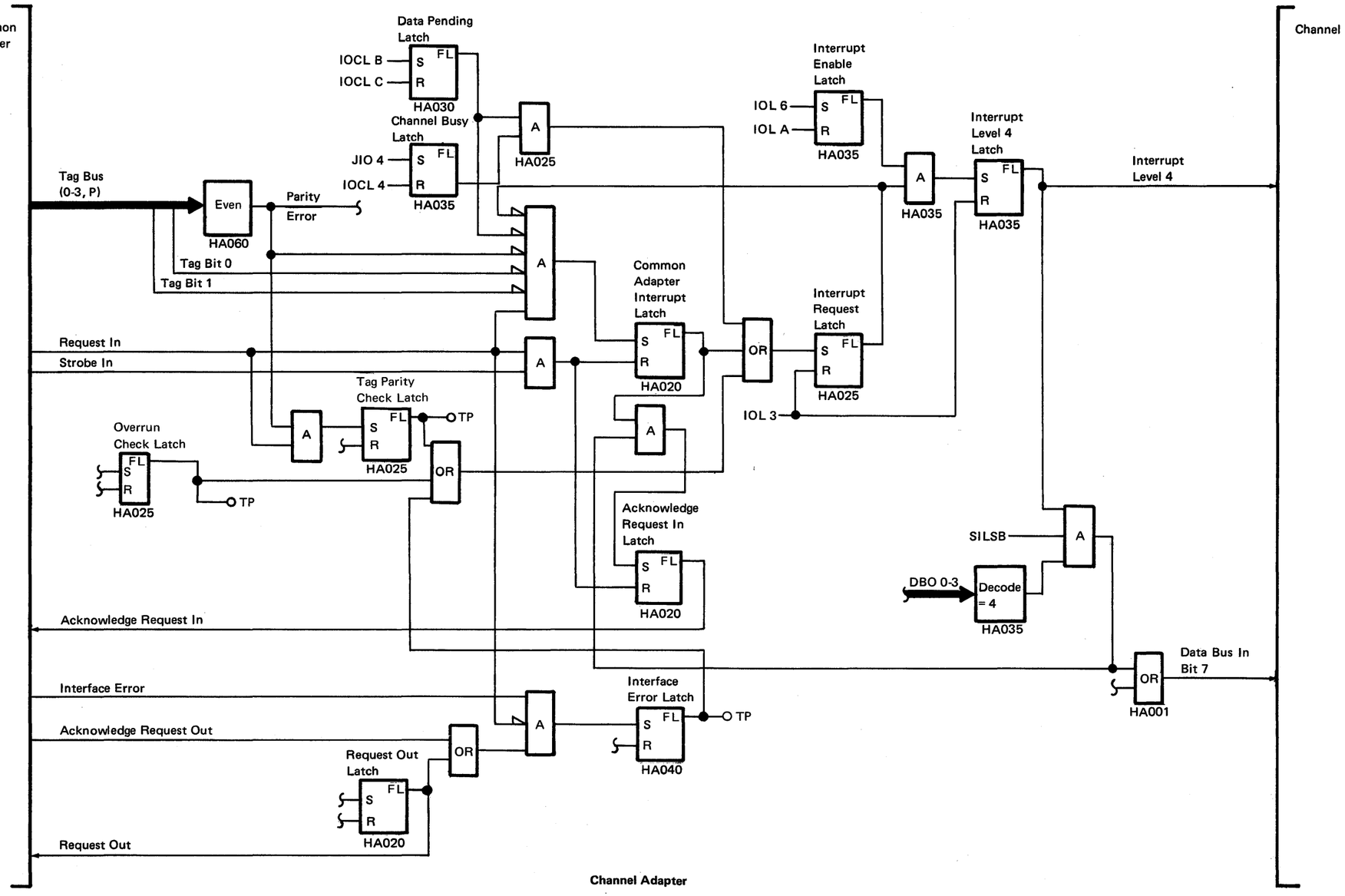
A data pending interrupt indicates that the common adapter is ready to start moving data using cycle steal, so the disk microcode must check to see if the cycle steal channel is available. The status of the channel is stored in the 'channel busy' latch, which is located in the channel adapter. The latch may be set by either the disk or the diskette to indicate that the channel is being used. Before either I/O device can use the cycle steal channel, it must test this latch to see if it has already been set by the other I/O device. This test is done by a disk attachment command, jump if channel is not busy (JIO 4). If the 'channel busy' latch is already set, the jump condition is not met, and the I/O device must wait until the latch is reset. If the latch is not set, the jump condition is met, and the command (JIO 4) also sets the latch to indicate that the I/O device is now using the channel. When the I/O device completes using the channel, the 'channel busy' latch is reset using the disk attachment command, reset channel busy latch (IOCL 4).

If the disk is the I/O device that is resetting the 'channel busy' latch, the disk microcode sends the status of the 'channel busy' latch to the diskette microcode, so that the diskette can use the channel if it is needed. If the diskette is the I/O device that is resetting the 'channel busy' latch, this status is not needed. The channel adapter has a 'data pending' latch, which is set if the disk attachment needs the channel when the channel is being used by the diskette. When the 'data pending' latch is set, it sets the 'interrupt request' latch when the diskette resets the 'channel busy' latch. This interrupt permits the disk microcode to set up the storage addresses for the data cycle steals that follow.

The common adapter can send a second interrupt request to the channel adapter before the disk microcode interrupt level 4 routine (FDIOCH) is completed. Under these conditions, the second interrupt request will not be recognized by the channel adapter until the first interrupt is reset. An interrupt from the common adapter is also ignored by the channel adapter when the 'data pending' latch is set. This indicates that the data pending interrupt was not completed because the diskette was using the cycle steal channel.

The third condition that sets an interrupt occurs when specific error conditions are set in the channel adapter. A channel overrun check, tag parity check, or interface error condition sets the 'interrupt request' latch. See *Error Conditions* later in this section. The interrupt is sent to the control processor if the 'interrupt enable' latch is set.






Channel Adapter

## Burst Mode Cycle Steal

The read data, write data, read identification, write identification, and scan operations need to move data between the control processor and the disk attachment. To move this data, burst mode is used. The disk channel adapter sets the 'block processor clock' latch, which causes the processor for the selected storage (control processor or main processor) to stop; this action permits data to be moved directly to or from storage. When the processor is stopped, the 'disk burst mode gated' line is set by the control processor, to indicate that cycle steal may start. The processor does not execute any more instructions until the 'block processor clock' latch and the 'disk burst mode gated' line are reset. The channel adapter sets the 'disk cycle steal request' line when the channel adapter is ready to move the first byte of data, and resets this line at the start of the cycle that moves the last byte of data for that block of data.

The cycle steal logic in the channel adapter does not permit the 'block processor clock' latch to be set under the following three conditions:

- The channel adapter 'interrupt request' latch is set. After the channel adapter responds to a data pending interrupt from the common adapter, the common adapter could be ready to start moving data before the interrupt routine is done. Therefore, when the 'interrupt request' latch is on, the 'block processor clock' latch cannot be set and cycle steal does not occur. Data cycle steal can start immediately after the interrupt is reset.

- The channel adapter 'data pending' latch is set. A data pending interrupt from the common adapter will be followed by a request to start moving data. If the cycle steal channel was not available, the 'data pending' latch is set, so that an interrupt will be set by the channel adapter as soon as the channel is not busy. To prevent the data cycle steal until the diskette is no longer using the cycle steal channel, the 'block processor clock' latch cannot be set when the 'data pending' latch is set. Data cycle steal starts after the 'data pending' latch and the interrupt caused by the 'data pending' latch is reset.

- Interrupt level 2 is set on the channel. The movement of cycle steal data to or from the disk attachment could cause the processor to be stopped for up to 1 millisecond. This action causes a problem for an I/O device (such as communications) that needs more frequent servicing. To meet the needs of other I/O devices, the disk attachment does not move more than 258 bytes of data without resetting the 'block processor clock' latch. If more data is to be moved, the 'block processor clock' latch is immediately set again, unless interrupt level 2 is set. The 'block processor clock' latch cannot be set when interrupt level 2 is set, but can be set immediately following the resetting of interrupt level 2. This permits interrupt level 2 to override disk attachment cycle steal on a data record boundary, which permits more frequent servicing of other I/O devices.

The common adapter starts a cycle steal request by setting the 'CA burst mode' and 'request in' latches and the tag bus to indicate the type of request.

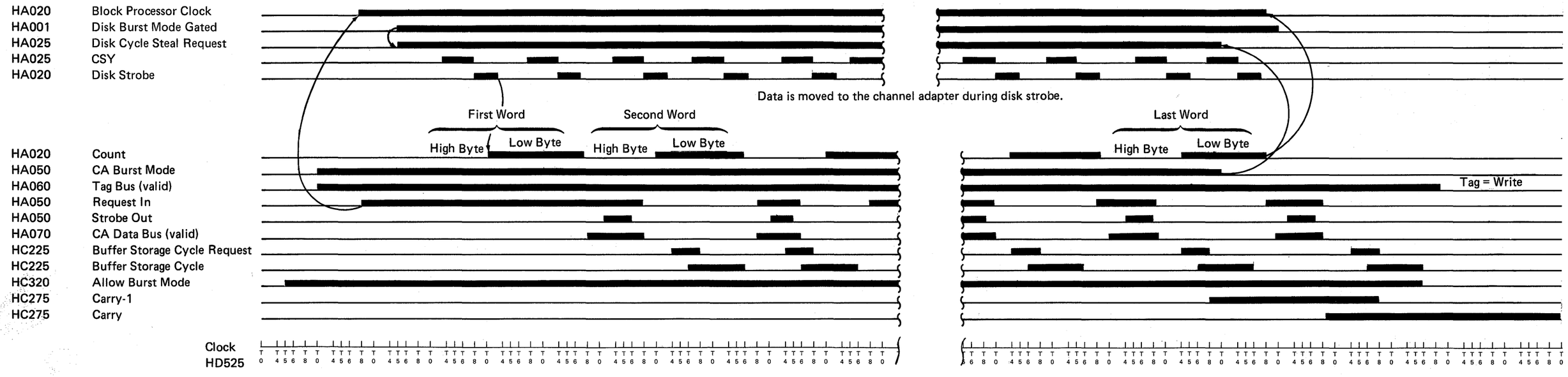
| Bits<br>0 1 2 3 P | |
|---|---|
| 1 1 0 F P | Write |
| 1 0 0 F P | Read |

| | |
|---|---|
| Bit 3=0 | Data is for a disk drive A operation. |
| Bit 3=1 | Data is for a disk drive B operation. |
| Bit P | Odd parity for bits 0 to 3. |

Data Bus Out
Data Bus In
Block Processor Clock
Disk Strobe
Disk Burst Mode
Channel
Tag Bit 0
Tag Bit 1
(3)
Even
HA060
BPC Latch
S FL R
HA025
Tag = Read
Tag = Write
PH CD C R
HA025
HA025
A
HA035
N
HA070
Data Out Reg
HA040
Data In Reg
HA070
CA Data Bus
Repower
Data Bus
Data Register
Address (11)
Data (18)
ROS
Controller
Controller Data Bus
Controller Register
Buffer Storage
Address
Read/ Write
Storage Cycle
Address Assembler
Buffer Storage Control
Increment
Tag Bus
Tag Register
Strobe In
Request Out
Request In
Acknowledge Request In
Interface Control
Clocks
20-MHz Clock
T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
TA
Carry
Carry-1
Allow Burst Mode
HB100 A-A2D2
HB200 A-A2C2
Burst Mode Latch
S FL R
HA001
OR
T FF R
HA020
Count
TP
Channel Adapter (A-A2E2)
Common Adapter

The tag bus indicates write when data is being moved to the common adapter during a write data, write identification, or scan operation. The channel adapter sets the 'block processor clock' latch when the cycle steal request is received from the common adapter. When the 'disk burst mode gated' signal is received from the control processor, the channel adapter sets the 'disk cycle steal request' latch to indicate that the channel adapter is ready to receive the first byte of data. The 'disk strobe' pulse is used to load the data into the high byte of the data out register. At the end of the 'disk strobe' pulse, the 'count' latch is switched, so that the data during the next 'disk strobe' pulse is loaded into the low byte of the data out register. When the low byte is loaded, the channel adapter responds to the 'request in' line by sending the data word to the common adapter, where the 'strobe out' pulse is used to load the data word into the data register and then moved to the common adapter buffer storage during the next available data buffer storage cycle. When the word is loaded into the data buffer storage, the 'request in' line is set again to indicate that the common adapter is ready for the next byte. The next 2 bytes of data from the channel are then loaded into the high and low bytes of the data out register, and this cycle is repeated until the block of data is moved to the common adapter. When the common adapter has received all but the last word, it resets the 'CA burst mode' latch, which causes the channel adapter to reset the 'disk cycle steal request' latch after receiving the high byte of the last word from the control processor. When the low byte of that word is moved to the channel adapter, the 'block processor clock' latch is also reset, which ends the cycle steal operation.

**Cycle Steal Timings for Write**



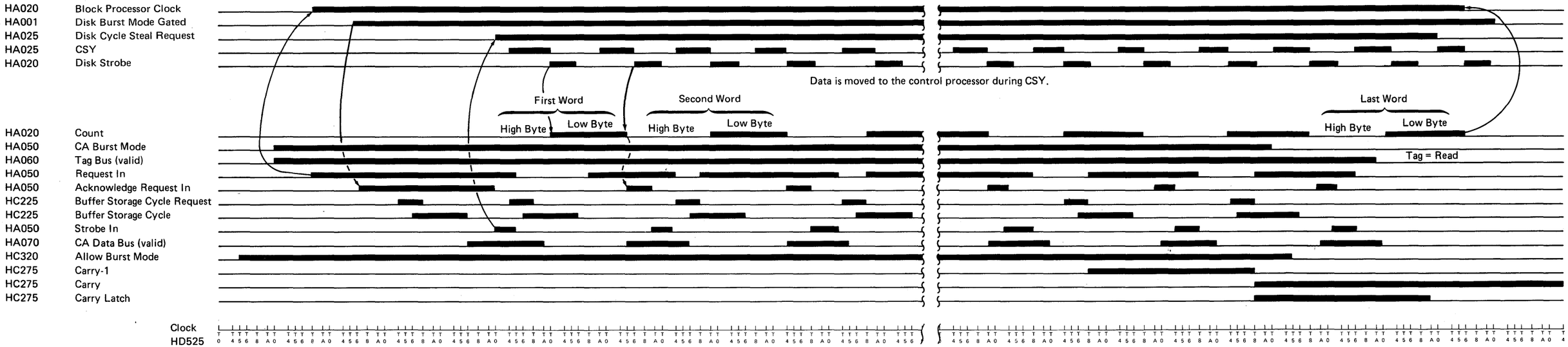

The 'allow burst mode' latch is set in the common adapter during the 'T5' pulse. The next 'T0' pulse sets the 'CA burst mode' latch and sends valid tag codes on the tag bus. The 'request in' latch is set during the following 'T8' pulse. When the common adapter receives the 'strobe out' pulse, the word of data to be written is loaded into the data register. The common adapter sets the 'buffer storage cycle request' latch during the next 'T4' pulse. The buffer storage cycle starts with the 'T6' pulse and ends with the next 'T6' pulse. The word of data is sent to buffer storage during this cycle. The 'request in' latch is set again during the 'T8' pulse. During the buffer cycle before the last request in, the 'CA burst mode' latch is reset by the 'T0' pulse.

The tag bus indicates a read when data is moved to the control processor during a read data, read diagnostic, read identification, or scan operation. The channel adapter sets the 'block processor clock' latch when the cycle steal request is received from the common adapter. When the 'disk burst mode gated' line is received from the control processor, the channel adapter sets the 'acknowledge request in' latch. The common adapter then requests a buffer storage cycle. The data word is received from buffer storage and sent out on the common adapter data bus. The channel adapter loads the data word from the common adapter into the data in register with the 'strobe in' pulse. The common adapter sets the 'request in' line again and requests another buffer storage cycle so that data will be available immediately when the next 'acknowledge request in' pulse is received.

The first 'strobe in' pulse also causes the 'disk cycle steal request' latch to be set. The high byte of the data in register is sent to the control processor on the data bus in, and is loaded into storage during the 'CSY' pulse.

At the start of the 'disk strobe' pulse, the 'count' latch is switched, to cause the low byte of the data in register to be sent to the control processor during the next cycle. The second 'disk strobe' pulse switches the 'count' latch again, to cause the channel adapter to set the 'acknowledge request in' latch. The next word can now be moved from the common adapter to the data bus in register. This cycle is repeated until the block of data is moved. While the common adapter is sending the last word of data, the 'request in' and 'CA burst mode' latches are reset. When the high byte of the last word is sent to the control processor, the 'disk cycle steal request' latch is reset. The 'block processor clock' latch is reset when the low byte is sent, to end the cycle steal operation.

**Cycle Steal Timings for Read**




The 'allow burst mode' latch is set in the common adapter during the 'T5' pulse. The next 'T0' pulse sets the 'CA burst mode' latch and sends valid tag codes on the tag bus. The 'request in' latch is set during the following 'T8' pulse. When the common adapter receives the 'acknowledge request in' pulse, the common adapter sets the 'buffer storage cycle request' latch during the next 'T4' pulse. The buffer storage cycle starts with the 'T6' pulse and ends with the next 'T6' pulse. The word of data to be sent to the control processor is received from buffer storage during this cycle. The 'strobe in' pulse, which starts during the 'TA' pulse, sends the data word to the channel adapter data in buffer. The common adapter immediately gets the next word of data from buffer storage and sets the 'request in' latch again during the 'T8' pulse.

## Servo Tracks and Servo Track Follow

### Access Locating

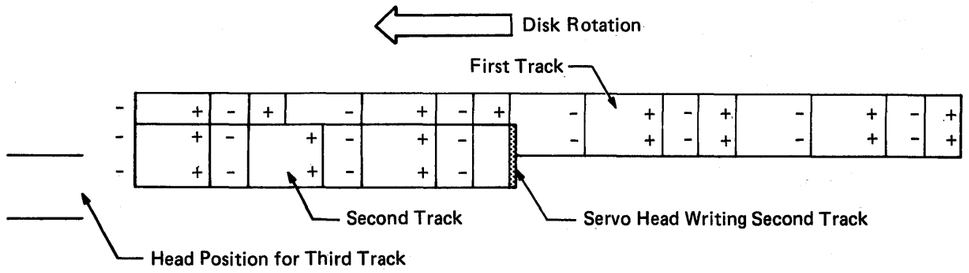
The dedicated servo surface, located on the disk surface nearest the disk enclosure casting, controls the primary access location. Signals read from this surface by the servo head are used to position the data heads on the actuator over the selected data cylinder.

The sample servo, a field on the data surfaces at the start of each sector, is used for exact data head alignment. Because of the two servos, the servo head may be slightly off the servo track when the data heads are correctly aligned.

The sample servo signals generate the 'sample servo error' signal. This signal is set by the sample servo circuit every time an index pulse or sector pulse occurs.

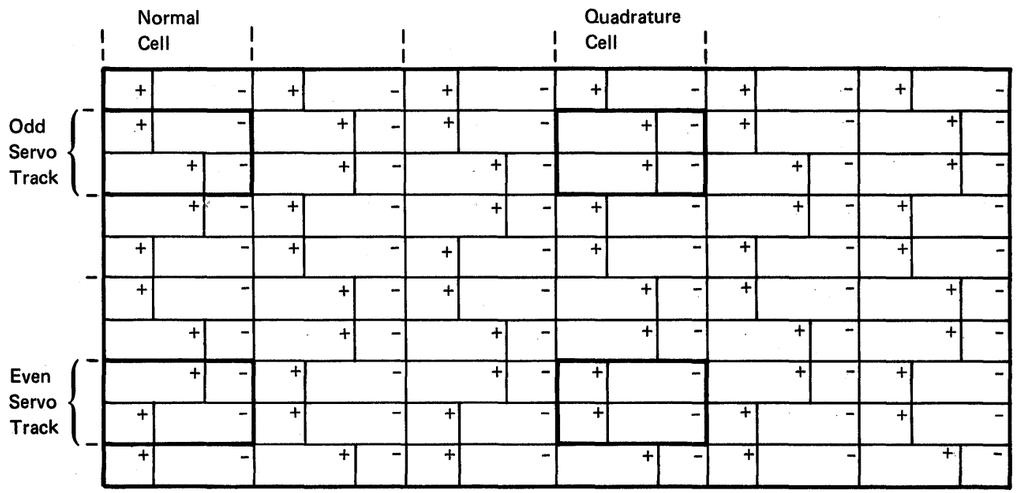
### Dedicated Servo

The dedicated servo surface is written during disk enclosure assembly and cannot be written in the field. The servo surface is written by the servo head, which is the only time the servo head is used for writing. The servo head writes the first track at the outer limit of the servo surface, then moves inward one half track and writes the second track, which overlaps half of the first track.




*Note:* The + or – indicates the polarity of the pulses induced in the data head when reading.

This process is continued until all of the servo surface is written. When the servo surface writing is complete, the following patterns are on the servo surface.

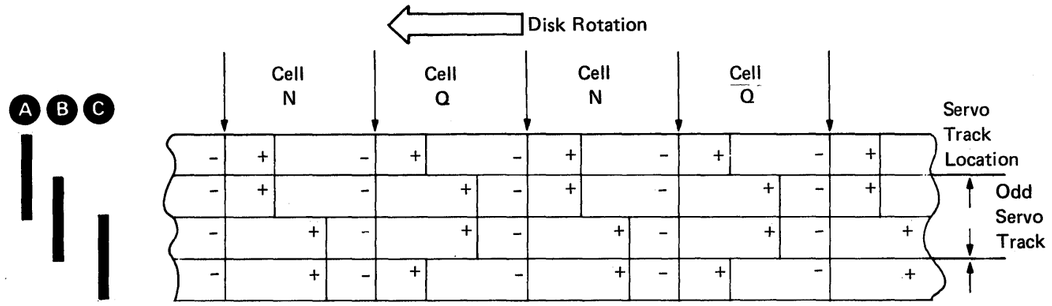


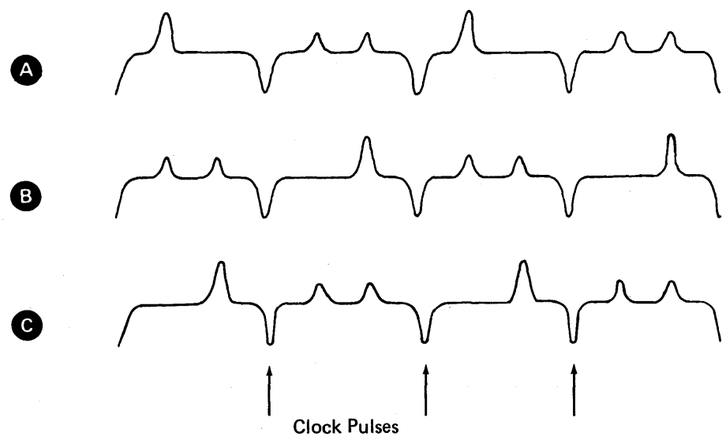
Two of the half track patterns are used as a servo track, which is aligned with the data tracks for a specific cylinder. Each servo track is made up of a series of cells, both normal cells and quadrature cells. Each cell normally has a clock pulse at the start, which is used for timing the read, write, and seek circuits. These clock pulses are always generated. There are 300 cells per sector, so that each cell is equal to 2 data bytes on the data tracks.

The normal cells contain position pulses that are used to keep the servo head on track. The quadrature cells are used to identify the servo track as an odd track or an even track. A quadrature cell on an even track has a pulse during the first part of the cell. A quadrature cell on an odd track has a pulse during the second part of the cell.

The servo output changes as the servo head moves away from the center of the track. The following shows the servo head output at three positions (A, B, C) of the servo head over an odd track.

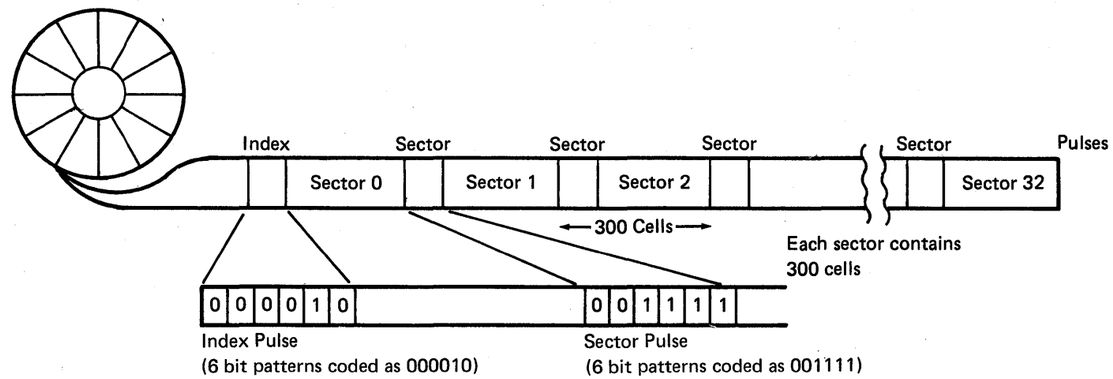
The servo signals for B and C are the same except for a delay. The correct servo track is identified by changing the cells at the start of each sector. The servo logic uses these changed cells to synchronize to servo patterns. The changed servo patterns contain four cells with a missing clock pulse.

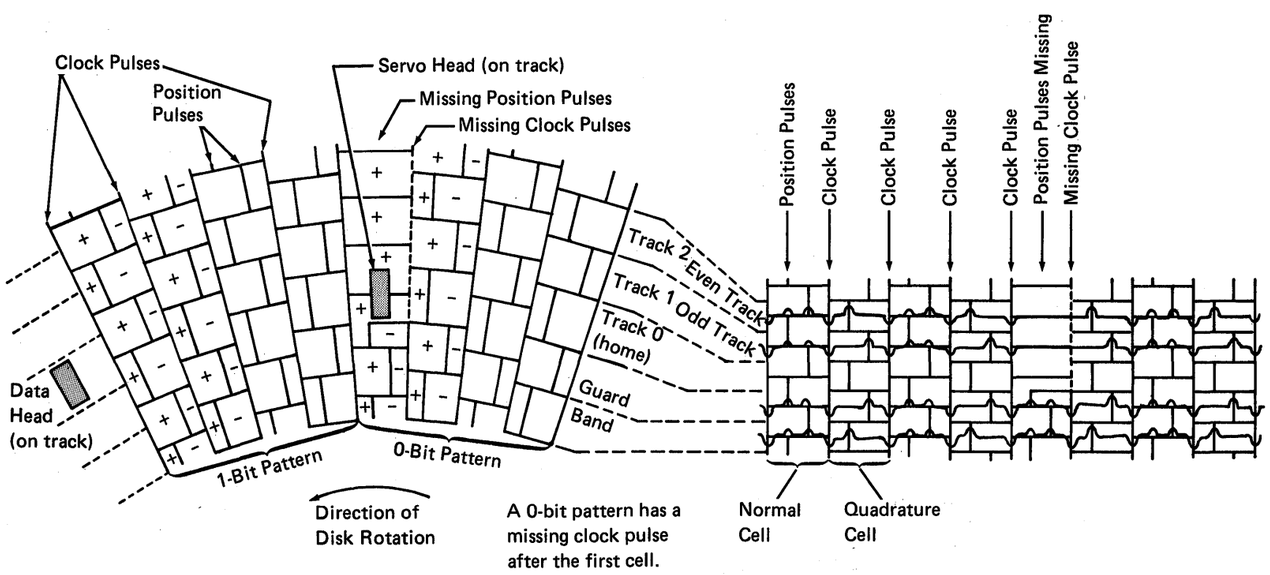


## Servo Tracks and Servo Track Follow (continued)

The changed group of four cells represents a 0-bit pattern and a normal group of four cells represents a 1-bit pattern. Six bit patterns are written at the start of each sector. The first sector of a track has a bit pattern of 000010; this is the index pulse. The following sectors of that track have a bit pattern of 001111; these are sector pulses.



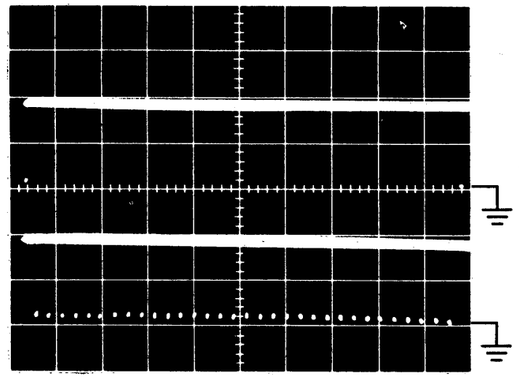



The following waveshapes show the index and sector pulses.

The first waveshape shows the index and sector pulses that should always be present.

Chan 1 -Index D2S13
Chan 2 -Sector D2S10

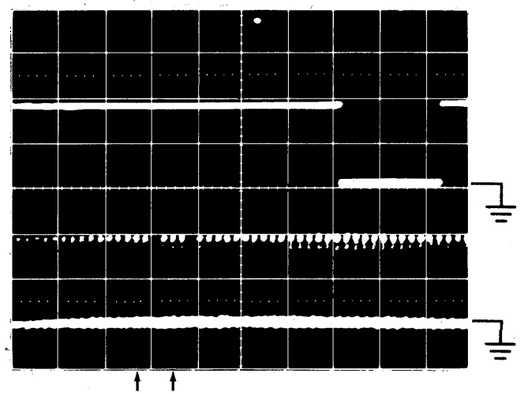
Voltage: 0.2 V/div
Time: 2 ms/div
Trigger: Internal
DC Input



Index and sector pulses are generated from missing clock pulses on the servo track.

Chan 1 -Sector D2S10
Chan 2 +Servo Clock SS F2B04

Voltage: 0.2 V/div
Time: 0.1 ms/div, X10 magnification
Trigger: Internal
DC Input




Adjust the position so the missing clock pulses can be seen.

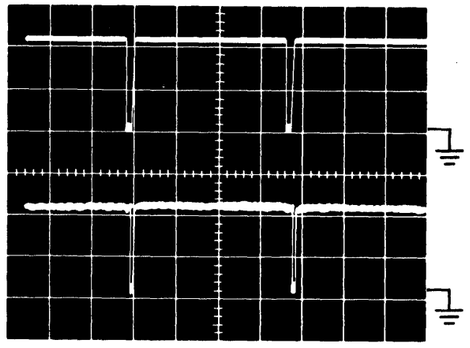
*Note:* A X10 grounded probe was used on the disk drive board (E-A1 or E-B1) to obtain these waveshapes.

**Track Follow Waveshapes**

The following waveshapes can be seen on an oscilloscope if the servo circuits are working correctly and the servo head is on a servo track. All waveshapes were taken with a X10 grounded probe on the disk drive board (E-A1 or E-B1).

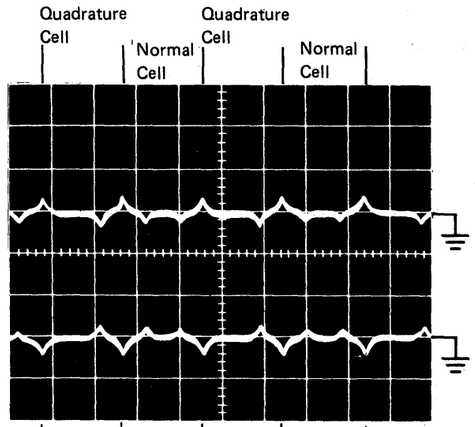
Chan 1 -Osc Early D2B13
Chan 2 -Osc Late D2J02

Voltage: 0.2 V/div
Time: 0.5 µs/div
Trigger: External + D2S10 -Sector
DC Input



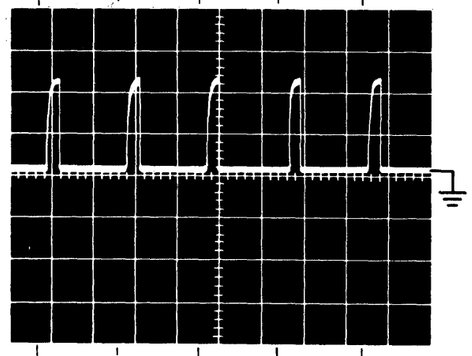
Chan 1 Servo Preamp Output F2D05
Chan 2 Servo Preamp Output F2D06

Voltage: 10 mV/div
Time: 1 µs/div
Trigger: External + D2S10 -Sector
AC Input



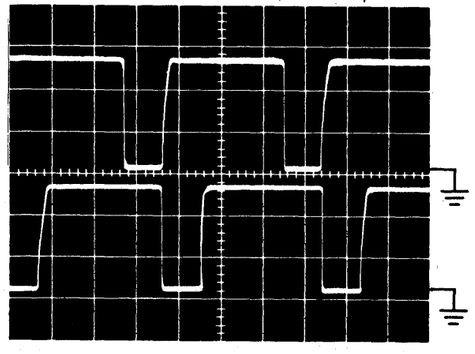

Chan 1 +Servo Clock SS F2B04

Voltage: 0.2 V/div
Time: 1 µs/div
Trigger: External + D2S10 -Sector
DC Input



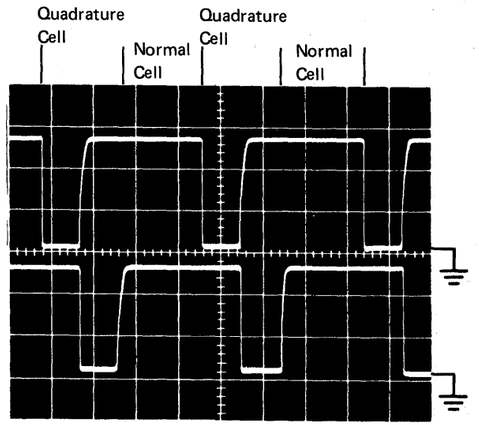
Chan 1 -Select Demod N1 F2B10
Chan 2 -Select Demod N2 F2D09

Voltage: 0.2 V/div
Time: 1 µs/div
Trigger: External + D2S10 -Sector
DC Input



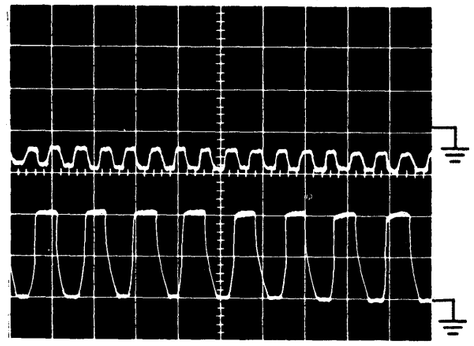
Chan 1 -Select Demod Q1 F2B08
Chan 2 -Select Demod Q2 F2B09

Voltage: 0.2 V/div
Time: 1 µs/div
Trigger: External + D2S10 -Sector
DC Input



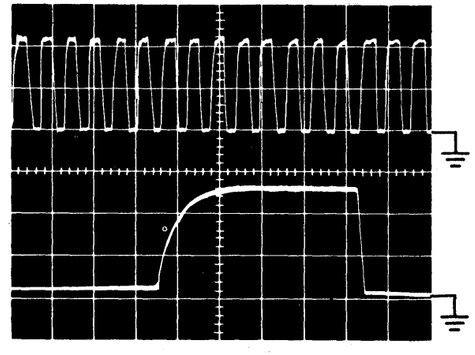

Chan 1 2F Write Clock E2G12
Chan 2 1F Write Clock B2U12

Voltage: 0.2 V/div
Time: 0.1 µs/div
Trigger: External + D2S10 -Sector
DC Input



Chan 1 1F Write Clock B2U12
Chan 2 -Counter 4 D2M06

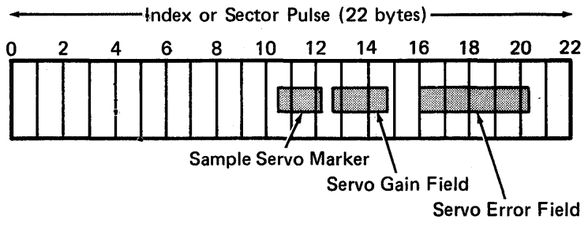
Voltage: 0.2 V/div
Time: 0.2 µs/div
Trigger: External + D2S10 -Sector
DC Input



**Sample Servo**

The sample servo field is 12 1/2 bytes long. This field is included in a protected area at the start of each sector and is located on every data surface. The sample servo is written during disk enclosure assembly and is synchronized with the dedicated servo patterns. The sample servo is written in half track patterns and contains three main areas:

- Sample servo marker
- Servo gain field
- Servo error field




*Sample Servo Marker:* This area is used to synchronize the sample servo circuits.

*Servo Gain Field:* This field is used to adjust the automatic gain control (AGC) for sample servo signal deviations. These deviations are caused by head to disk surface deviations, disk surface changes, and the track location on the disk surface. The AGC is necessary so that the servo error field output is increased to the same level for each sector on the disk.

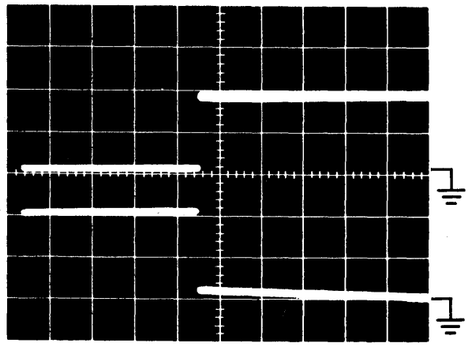
*Servo Error Field:* This field is used by the servo error circuits to determine the position of the data head relative to the center of the data track. The servo error field contains two tracks written with opposite patterns so that when the data head is centered, there is no output. If the data head is not centered, the track patterns will cause an output. This output is collected in a capacitor which generates a voltage that is used to offset the servo head enough to center the data head.

**Sample Servo Waveshapes**

The following sample servo waveshapes can be seen on an oscilloscope if the sample servo circuits are working correctly and the data heads are on track. All waveshapes were taken with a X10 grounded probe on the disk drive board (E-A1 or E-B1).

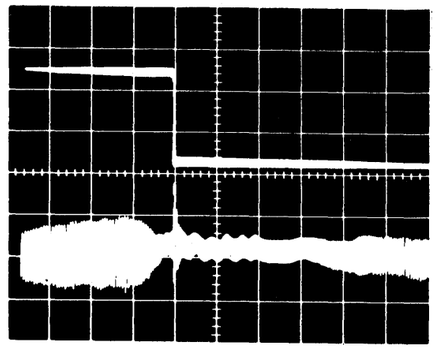
Chan 1 -Sector D2S10
Chan 2 +Enable Sample Servo D2J10

Voltage: 0.2 V/div
Time: 5 μs/div
Trigger: Internal
DC Input



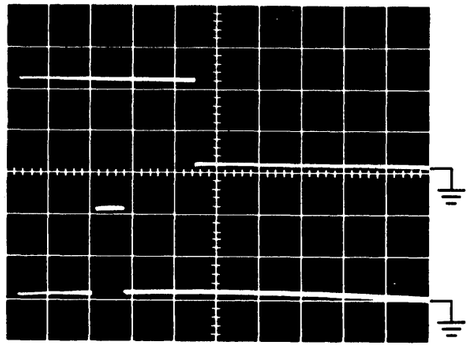
Chan 1 +Enable Mark Detect D2U13
Chan 2 Data PES E2B13

Voltage: 0.2 V/div Chan 1
5 mV/div Chan 2
Time: 1 μs/div
Trigger: Internal
DC Input



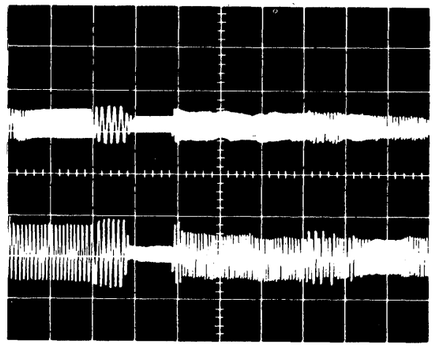
Chan 1 +Enable Sample Servo D2J10
Chan 2 + Enable Mark Detect D2U13

Voltage: 0.2 V/div
Time: 5 μs/div
Trigger: Internal
DC Input



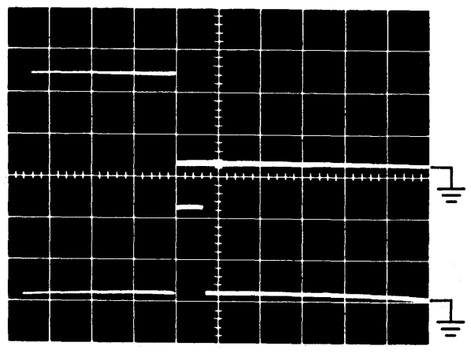
Chan 1 Actuator I/O Line A B2D05
Chan 2 Buffered Analog Data A E2B03

Voltage: 5 mV/div
Time: 1 μs/div
Trigger: External + D2S10 -Sector
AC Input



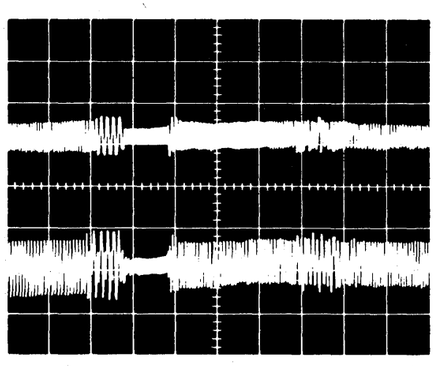
Chan 1 +Enable Mark Detect D2U13
Chan 2 +Shift Reg Clock D2J09

Voltage: 0.2 V/div
Time: 1 μs/div
Trigger: Internal
DC Input

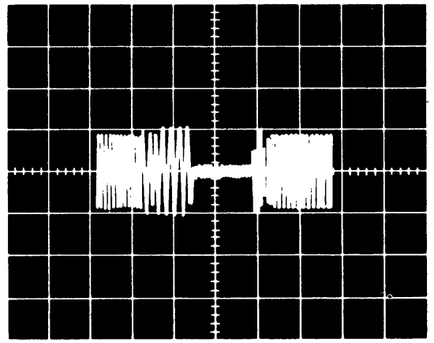


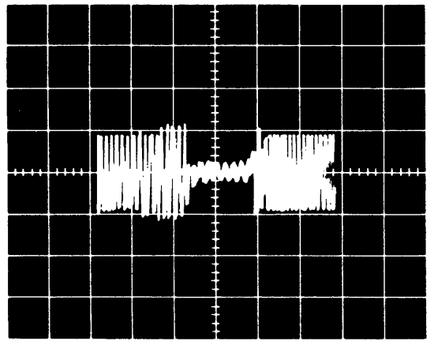
Chan 1 Actuator I/O Line B B2D06
Chan 2 Buffered Analog Data B E2D02

Voltage: 5 mV/div
Time: 1 μs/div
Trigger: External + D2S10 -Sector
AC Input



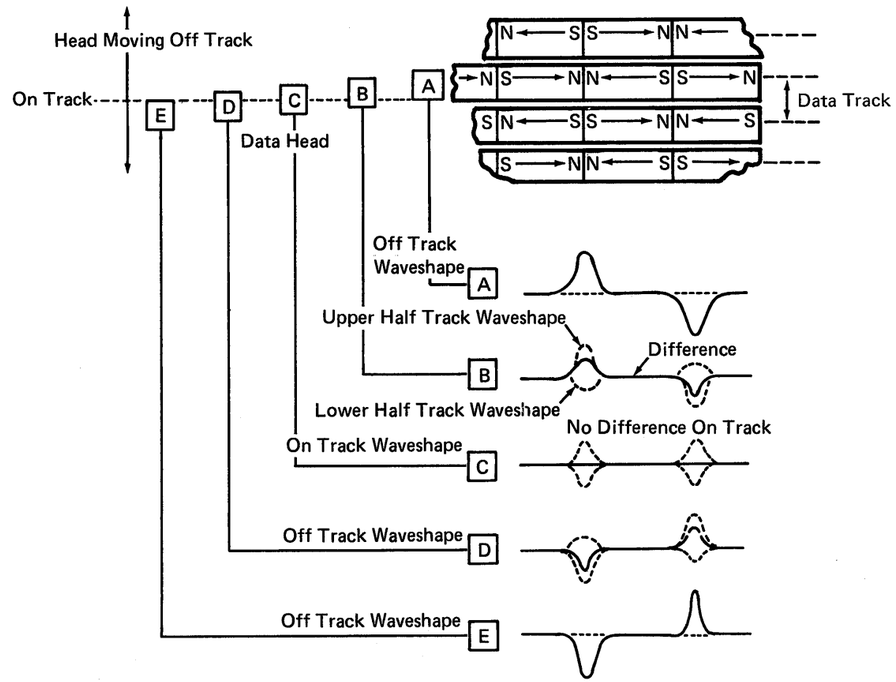
The oscilloscope traces show the output of two data heads on the same disk enclosure. The upper trace shows the signal from a data head in position C. The lower trace shows the signal from a data head that is slightly off track after much use and has a large error signal. Both of the heads operated without errors.





Chan 1 High-Level Diff Analog Signal B2D13

Voltage: 50 mV/div
Time: 2 μs/div
Trigger: External + D2U13 + Enable Mark Detect



## Access Operations

The access operations are:

- Recalibrate
- Control seek
- Automatic seek

The recalibrate and control seek operations are started by loading the command byte of the common adapter file control block (FCB) for the desired disk drive.

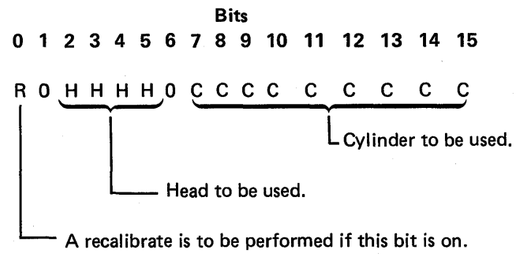
The command byte for an access operation is:

| | Bits 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|
| Seek | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Recalibrate | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

The automatic seek is selected by setting bit 12 of the command byte to 0 for read and write operations. The common adapter uses the head byte and cylinder number from the FCB to generate the seek control word. The common adapter checks the cylinder number to verify that a valid cylinder is requested, but does not check the head byte. The selection of a head that is not valid results in a seek incomplete or a no record found error.

### Starting the Operation

The common adapter starts the access operation by sending the seek control word to the disk drive during two consecutive control bus cycles. The seek control word contains 2 bytes, as described below:




The seek required address byte (bits 8 through 15) is moved to the disk drive first on the control bus by setting a tag code of 010 (A) and (B). The common adapter sets the 'control sample' latch (C), causing the disk drive to read the tag code (D) and then the control bus (E). The disk drive moves the control bus byte into the desired address register and sets the 'control sample received' latch (F). The common adapter then resets the 'control sample' latch (G), causing the disk drive to reset the 'control sample received' latch (H).

The seek control byte (bits 0 through 7) is then moved to the disk drive on the control bus by setting a tag code of 001. The control bus cycle is the same as before, but the disk drive moves the control bus byte, bits 0 through 5 into the head address register and bit 7 into the desired address register. A recalibrate operation needs only that the seek control byte, with a value of hexadecimal 80, be sent to the disk drive.

The access operation is started by the disk drive, and the seek complete bit is reset so that the disk drive will ignore any additional access commands, although the disk drive will respond correctly to the control bus cycle. An interrupt will be set by the disk drive when the seek or recalibrate is completed (Q). The access time is the time from the resetting of the 'control sample received' latch to the setting of the 'interrupt' latch.

When the interrupt is received, the common adapter performs a sense to the interrupting disk drive. The common adapter sets a tag code of 100 (J), and after a 100-nanosecond delay, sets the 'control sample' latch (L). The disk drive reads the tag code (M), sets the sense data on the control bus (K), and sets the 'control sample received' latch (P). When the common adapter has read the control bus (N), it resets the 'control sample' latch (R), which causes the disk drive to reset the 'interrupt' (T) and 'control sample received' (S) latches. The disk drive also resets the control bus after a slight delay.

**Tag 100**

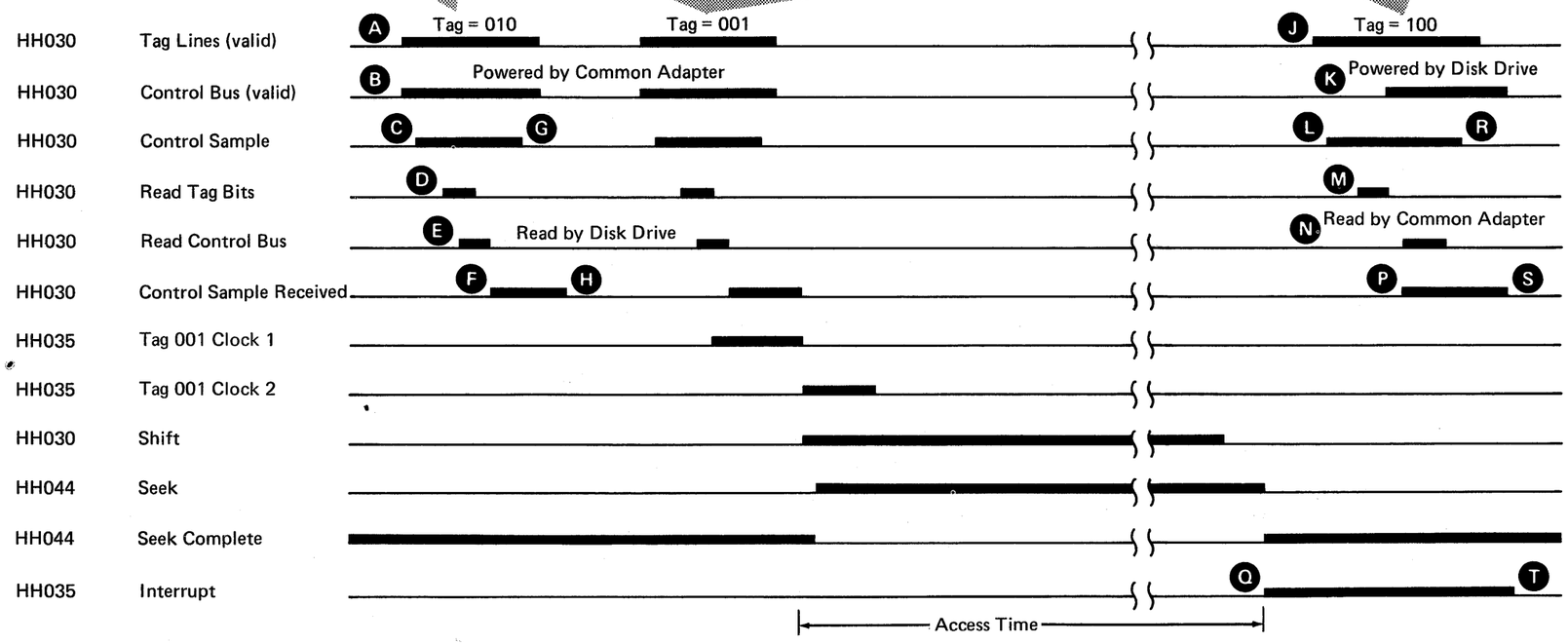
| Control Bus Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| Bit Meaning | Not used | Brake applied | Track unavailable | Command error | Data unsafe | Seek in-complete | Home | Not ready |
| Status When On (1) | | | | | | | | |

**Tag 010**

| Control Bus Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| Cylinder Address | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Status When On (1) | | | | | | | | |

**Tag 001**

| Control Bus Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| Bit Meaning | Recalibrate | Not used | Head select 8 | Head select 4 | Head select 2 | Head select 1 | Not used | Cylinder address 256 |
| Status When On (1) | | | | | | | | |



## Access Operations (continued)

### Performing the Operation

A tag code of 001 starts the seek operation. The track unavailable bit is set if the desired address register indicates a track larger than 359. The track unavailable bit is ORed with the command error bit to inhibit the '-set seek' line. This prevents a seek under error conditions.

The desired address is compared with the current address (stored in the absolute address accumulator) by the subtractor. The output of the subtractor is the difference between the two addresses and also the number of tracks that the actuator must go across. Any output other than all zeros activates the '-shift' line, which indicates actuator movement is needed. The '-shift' line activates the '-set seek' line, which starts the seek operation. The '-carry 256' line from the subtractor sets the '+out' latch, which controls the direction of the seek.

The access time between tracks is kept at a minimum by placing a high current on the actuator coil for the longest possible time during the access operation. The current is decreased to stop the actuator over the correct track.

If the output of the subtractor is more than 127 tracks, the full coil current is applied. If the output of the subtractor is less than 127 tracks, or when a long seek has less than 127 tracks left to go across, the coil current is controlled by the desired velocity output of the access velocity generator. The output of the subtractor and the '+half track' and '+quarter track' lines address a read-only storage (ROS) module. Each quarter track change in actuator location generates a ROS module output. These digital outputs represent the maximum voltage to use on the actuator coil for each specific seek length. The ROS outputs drive a digital-to-analog converter so that, as the subtractor output decreases from 127 to 0, the desired velocity output decreases.

Because of mechanical and electrical tolerances and conditions of the environment, the desired velocity output is controlled by the profile gain voltage. The profile gain voltage is generated by a seek calibration sequence. See *Seek Calibration Sequence* later in this section.

When the cylinder indicated by the desired address is reached, the '+seek' line is reset and after a delay to permit the selected data head to be centered over the data track by the sample servo, the seek incomplete bit is reset and an interrupt is set.

If a failure causes the actuator to go to a crash stop, phase lock oscillator synchronization is lost. This resets the '+seek' line and sets the 'not ready' line.

If the failure is less severe, or if the seek is longer than 2 seconds, a time-out resets the '+seek' line and sets the 'not ready' line. An interrupt is set for both failures.

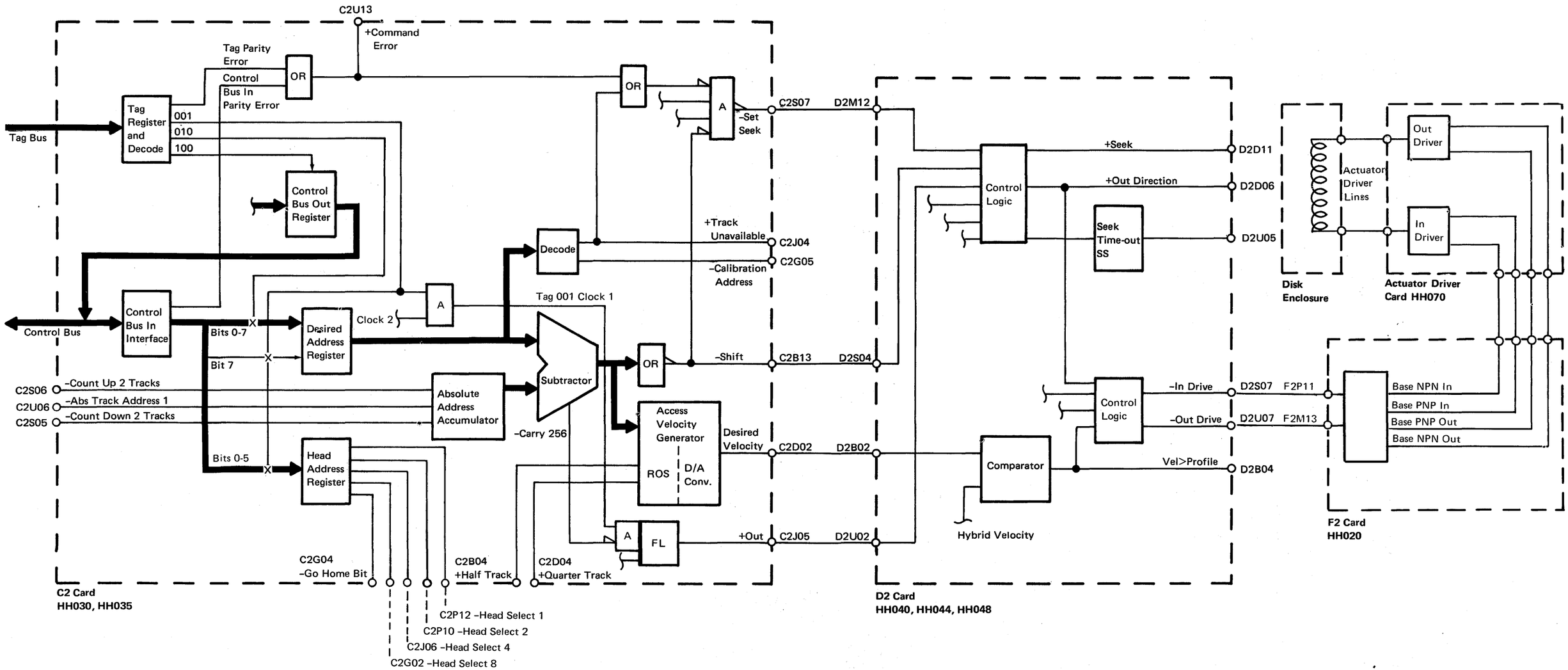
### Seek Calibration Sequence

The common adapter executes a seek calibration sequence after a power on reset. The seek calibration sequence performs a 128-track seek with head 1 selected following a recalibrate, or after a correct disk drive power on sequence, which causes the '+home' line to be active. When the seek command is sent to the disk, the '+seek' and '-calibration address' lines are set. The '-calibration address' line resets the profile gain counter. The profile gain counter generates the profile gain voltage.

During the seek, the real actuator velocity is compared to the desired velocity. If the real velocity is equal to the desired velocity (the 'vel > profile' line is active) before the actuator goes across 64 tracks, the profile gain counter is increased by 1. The profile gain counter is then increased by 1 for every two tracks gone across until 64 tracks have been gone across. The value in the profile gain counter generates a profile gain voltage which controls the access velocity generator.

The seek calibrate sequence also generates the 'handover velocity' signal, which is used to control the actuator on the last quarter track of a seek operation.

The disk enclosure has a dummy coil that is made of the same material as the actuator coil. If the actuator coil resistance changes, this dummy coil will change a comparable amount. The dummy coil is used by the analog circuits to adjust for the actuator coil resistance change.


C2U13
+Command Error
Tag Parity Error
Control Bus In Parity Error
OR
Tag Bus
Tag Register and Decode
001
010
100
Control Bus Out Register
Control Bus
Control Bus In Interface
Bits 0-7
Bit 7
Desired Address Register
Clock 2
A
Tag 001 Clock 1
Decode
OR
A
-Set Seek
C2S07
D2M12
+Track Unavailable
C2J04
-Calibration Address
C2G05
Subtractor
-Carry 256
OR
-Shift
C2B13
D2S04
C2S06 -Count Up 2 Tracks
C2U06 -Abs Track Address 1
C2S05 -Count Down 2 Tracks
Absolute Address Accumulator
Bits 0-5
Head Address Register
Access Velocity Generator
ROS
D/A Conv.
Desired Velocity
C2D02
D2B02
A
FL
+Out
C2J05
D2U02
C2G04 -Go Home Bit
C2B04 +Half Track
C2D04 +Quarter Track
C2P12 -Head Select 1
C2P10 -Head Select 2
C2J06 -Head Select 4
C2G02 -Head Select 8
C2 Card
HH030, HH035
Control Logic
+Seek
+Out Direction
Seek Time-out SS
D2D11
D2D06
D2U05
Comparator
Hybrid Velocity
Control Logic
-In Drive
-Out Drive
Vel>Profile
D2S07 F2P11
D2U07 F2M13
D2B04
D2 Card
HH040, HH044, HH048
Actuator Driver Lines
Out Driver
In Driver
Disk Enclosure
Actuator Driver Card HH070
Base NPN In
Base PNP In
Base PNP Out
Base NPN Out
F2 Card
HH020

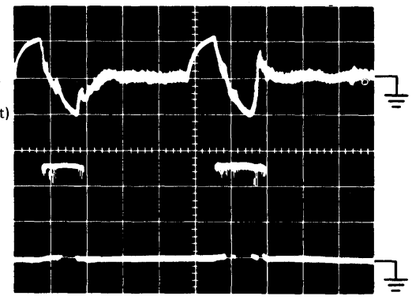
## Seek Waveshapes

The following waveshapes can be seen using the disk exerciser test, option 2, ±seek with a five-track (0-5) seek using the scope loop option.

All waveshapes were taken with a X10 probe on the disk drive board (E-A1 or E-B1).

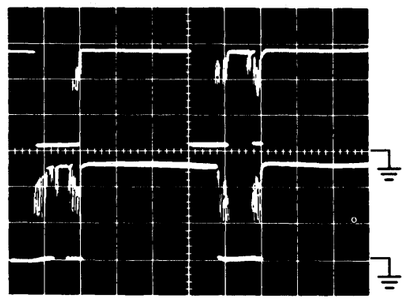
Chan 1 Coil Current Signal F2S10
Chan 2 Vel > Profile D2B04

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



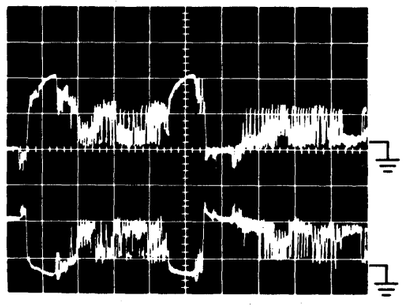
Chan 1 In Drive D2S07
Chan 2 Out Drive D2U07

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



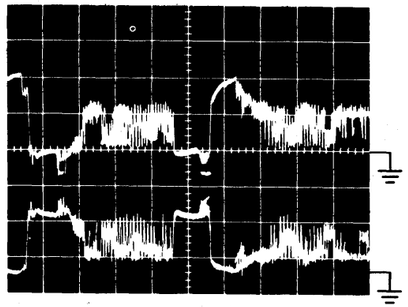
Chan 1 Base NPN In F2U04
Chan 2 Base PNP In F2U07

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C1J05 (+Out)
Mode: Alternate



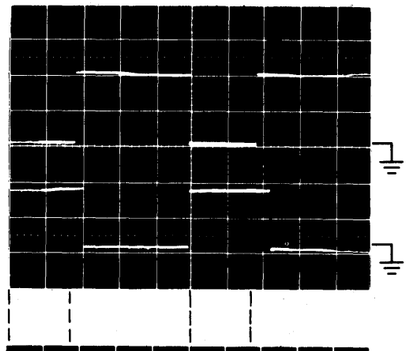
Chan 1 Base NPN Out F2U02
Chan 2 Base PNP Out F2M12

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



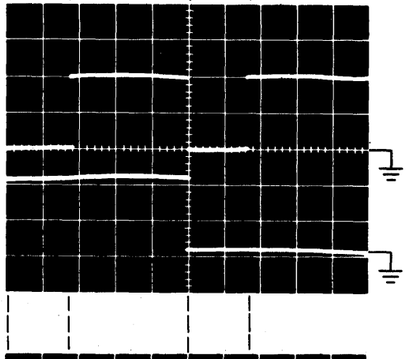
Chan 1 -Shift C2B13
Chan 2 +Seek D2D11

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: Internal
Mode: Alternate



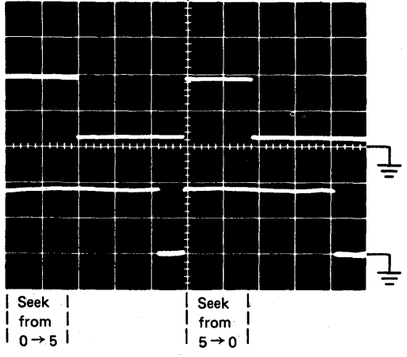
Chan 1 -Shift C2B13
Chan 2 +Out C2J05

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: Internal
Mode: Alternate



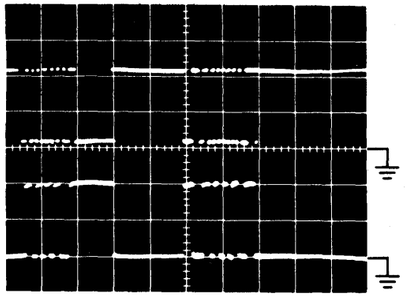
Chan 1 +Seek D2D11
Chan 2 -Seek Complete D2J13

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



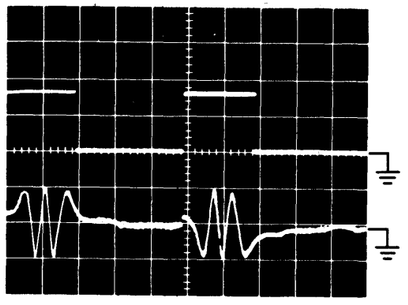

Chan 1 Quarter Track D2B10
Chan 2 Half Track D2D10

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



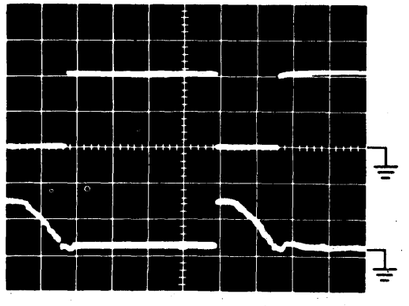
Chan 1 +Seek D2D11
Chan 2 Hybrid Position
Error Signal F2J11

Voltage: 0.2 V/div
Time: 1 ms/div
Trigger: External + C2J05 (+Out)
Mode: Alternate



Chan 1 -Shift C2B13
Chan 2 Desired Velocity C2D02

Voltage: 0.2 V/div Chan 1
20 ms/div Chan 2
Time: 1 ms/div
Trigger: Internal
Mode: Alternate



## Reading and Writing Data

The reading and writing of data is controlled by the logic on the data channel card on the disk drive card gate (E-A1B2 or E-B1B2). The data lines between the data channel card and the common adapter are contained in the dedicated cable (E-A1A5 or E-B1A5). The data channel card contains the circuits to decode the data read from the disk, coding data to be written on the disk, data safety, sample servo output, and the voltage controlled oscillator for the read clock.

### Head Selection

The head used for reading or writing is selected by the head address register, which was loaded during the last seek operation. A seek operation is needed to switch heads.

The output of the head address register is decoded and sent to the disk enclosure on five lines in the disk enclosure cable (E-A1A2 or E-B1A2). Three of these lines select chips which, with the other two lines, select the correct data head. Each chip contains the following circuits:

- Four write drivers
- Four read preamplifiers
- Head selection logic
- A common read output amplifier
- Read/write selection circuits
- Safety circuits

The disk enclosure does not reject head selections that are not valid, but indicates an error condition if a write is attempted on a head that is not valid. Head 11 is selected when a data unsafe condition occurs.

Each data head has a center tap, and these center tap lines are all connected together. Read mode is selected by grounding the center tap line to generate an output signal. Write mode is selected by putting a plus voltage on the center tap line to supply current for the write operation. The center tap circuits on the data channel card check the current through the center tap line and limit the current to a safe value.

The read and write signals are received from and sent to the disk enclosure on the 'actuator I/O line A' and 'actuator I/O line B' lines in the disk enclosure cable (E-A1A2 or E-B1A2).

To read or write data, the common adapter first activates the 'data select' line to cause the disk drive to gate the 'write clock', 'read clock', and 'read data' lines to the common adapter. When the common adapter activates the 'read' line, the disk drive puts the data it is reading from the disk on the 'read data' line. When the common adapter activates the 'write' line, the disk drive takes the data from the 'write data' line and writes it on the disk. The read data is synchronized with the read clock and the write data is synchronized with the write clock. Selecting both read and write at the same time causes a data unsafe condition.

**Writing Data**

The 'write data' line in the dedicated cable enters a 4-bit shift register on the data channel card and is moved through the register by a clock generated by the 1F write clock. The 4 bits in the shift register are sent to a precompensation encoder where the bits are encoded to a modified frequency modulation (MFM) signal. This signal is sent through the interface amplifier and then to the selected data head.

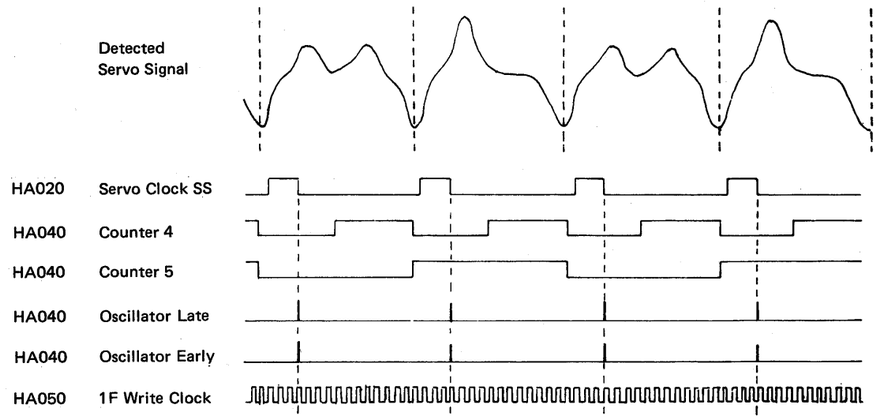
The precompensation encoder is used to prevent timing errors that could occur because of the high density of data on the disk surface. In principle, bit 3 of the shift register is the bit being written, bit 4 is the bit just written, and bits 1 and 2 are the bits yet to be written. The timing logic supplies the precompensation encoder with three clock pulses, all at the 2F frequency. These pulses are 2F early, 2F on time, and 2F late. The '2F early' pulse is 9 nanoseconds before the '2F on time' pulse and the '2F late' pulse is 9 nanoseconds after the '2F on time' pulse.

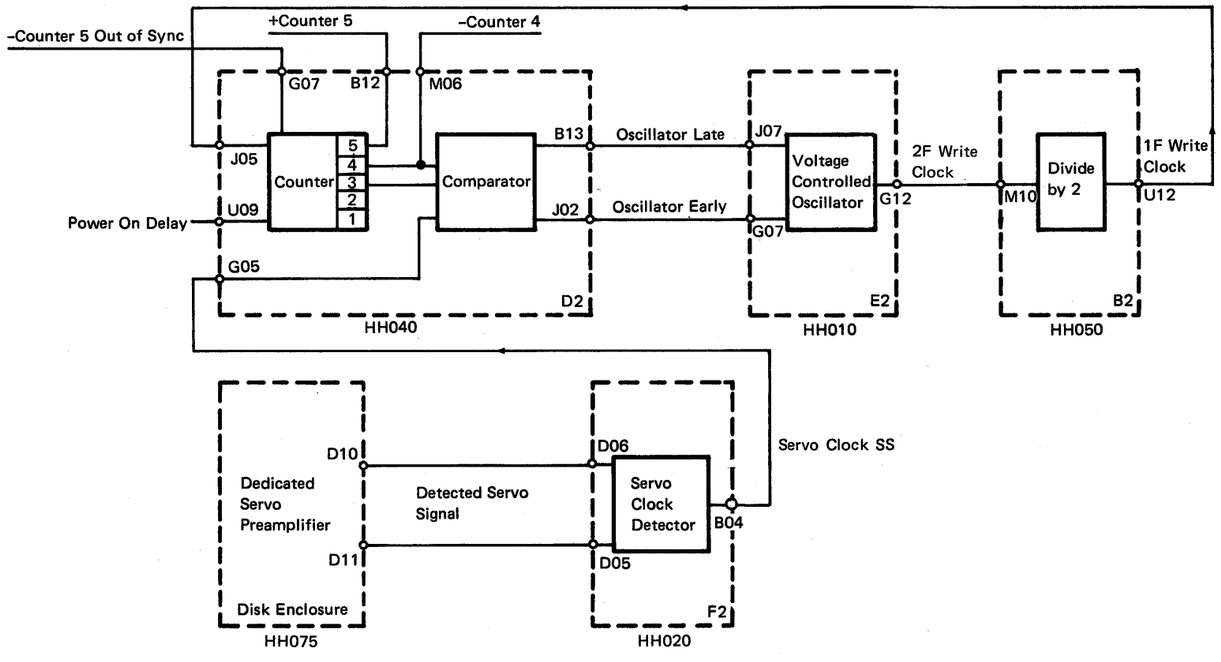
The precompensation encoder tests the 4 bits in the shift register and, from the 16 possible combinations, determines if bit 3 should be written to the disk on time, early, or late and uses the correct 2F clock pulse to write the bit. The write data is also sent to the read circuits for diagnostic purposes.

*Write Clock*

The 2F write clock is generated on the servo 1 card (E-A1E2 or E-B1E2) by a voltage controlled oscillator. This output is divided by 2 on the data channel card to supply the '1F write clock' signal which is used to write data on the disk.

The voltage controlled oscillator is controlled by the '-osc early' and '-osc late' lines from the logic 2 card (E-A1D2 or E-B1D2). Because one servo pulse is received for every 2 data bytes, the '1F write clock' signal is connected to a divide-by-16 counter so it can be compared to the servo pulses. If the write clock is too quick, a pulse is sent on the 'osc early' line; if the write clock is too slow, a pulse is sent on the 'osc late' line. This keeps the voltage controlled oscillator synchronized with the servo clock pulses from the disk.



### Reading Data

Any time a data head is selected and write mode is not active, the data head is reading data from the disk. The data read is amplified inside the disk enclosure and sent to the interface amplifier on the actuator I/O lines. There are two outputs from the interface amplifier. The first output, buffered analog data A and buffered analog data B, supplies sample servo input to the servo 1 card. The second output is increased by an amplifier again on the data channel card and decoded as data and control signals for the read clock.

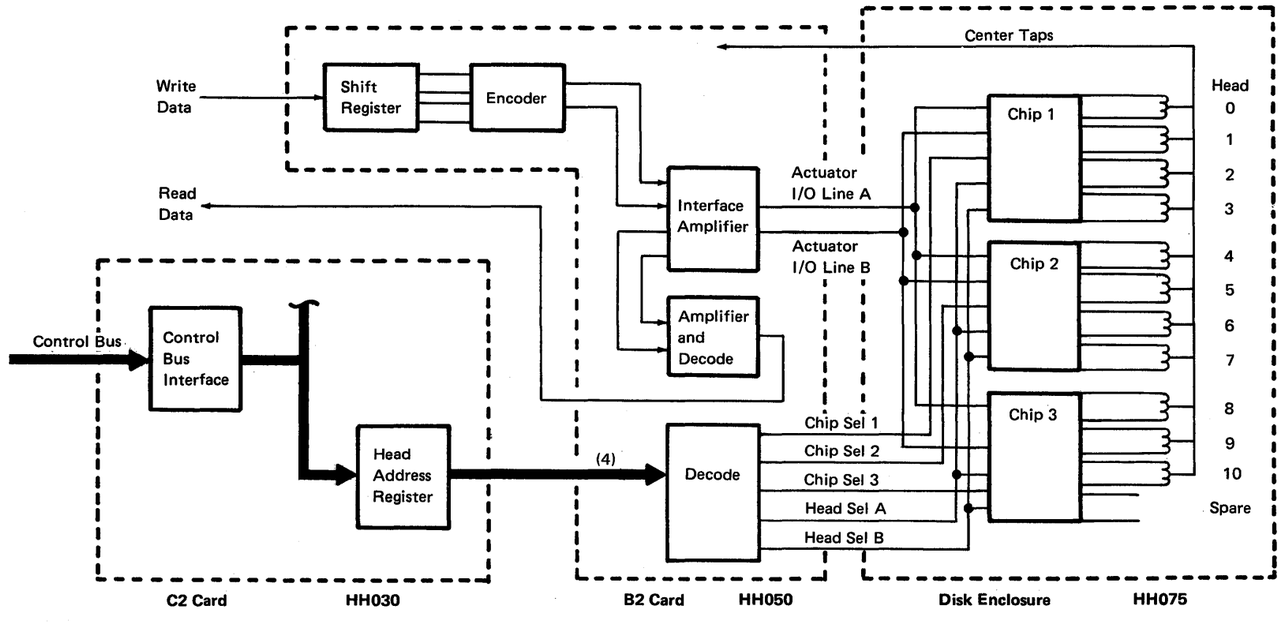
*Read Clock*

The read clock is generated by a voltage controlled oscillator on the data channel card. This is not the same voltage controlled oscillator that generates the write clock. This voltage controlled oscillator is synchronized, in frequency and phase, to the data read from the disk.

There is a phase decoder that senses differences in phase between the oscillator and the data, and increases or decreases the control voltage to the oscillator as needed. This oscillator is stopped by the '+servo inhibit VCO' line during the sample servo area of the sector.

Quick changes in operation can cause the voltage controlled oscillator to go out of synchronization. The 'fast sync' line is used to synchronize the oscillator. When the 'fast sync' line is active, the oscillator is stopped. When the 'fast sync' line is reset, the voltage controlled oscillator is started again in synchronization with the read data.

The voltage controlled oscillator output is used by the data separator circuits (MFM decoder) to separate the read data from the signals received from the disk enclosure. The data is then sent to the common adapter in synchronization with the read clock.




| Head Select Code | Head Selected | Output Lines Selected | | | | |
|---|---|---|---|---|---|---|
| | | Chip Select | | | Head Select | |
| | | 1 | 2 | 3 | A | B |
| 0000 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0001 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0010 | 2 | 1 | 0 | 0 | 0 | 1 |
| 0011 | 3 | 1 | 0 | 0 | 1 | 1 |
| 0100 | 4 | 0 | 1 | 0 | 0 | 0 |
| 0101 | 5 | 0 | 1 | 0 | 1 | 0 |
| 0110 | 6 | 0 | 1 | 0 | 0 | 1 |
| 0111 | 7 | 0 | 1 | 0 | 1 | 1 |
| 1000 | 8 | 0 | 0 | 1 | 0 | 0 |
| 1001 | 9 | 0 | 0 | 1 | 1 | 0 |
| 1010 | 10 | 0 | 0 | 1 | 0 | 1 |
| 1011 | 11* | 0 | 0 | 1 | 1 | 1 |

* There is no physical head 11, but this position is selected when a data unsafe condition occurs.

**Write Waveshapes**

The following write waveshapes were taken with a X10 grounded probe on the disk drive board (E-A1 or E-B1).

*Write ID*

Using the disk exerciser, set up the following command:

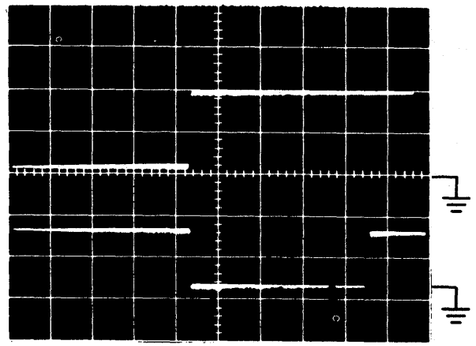
- Select write ID command.
- Select drive A or B.
- Select 0 for number of sectors.
- Select 0 for sector number.
- Select hex 167 for cylinder number.
- Select 0 for head number.
- Select data field 1.
- Set up a data pattern of hex 00000167 at location 000.
- Select the scope loop option to execute the command.

The following waveshapes will be observed if the write circuits are working correctly and there are no servo problems.

*Note:* Before an ID field is written, the ID should be read to verify that the sector is not defective. For the example above, if the ID field that is read is not hex 00000167, it should be restored to its original value, or a different sector should be selected. As an example, if 4 is selected for the head number, the expected ID field is hex 00001167.

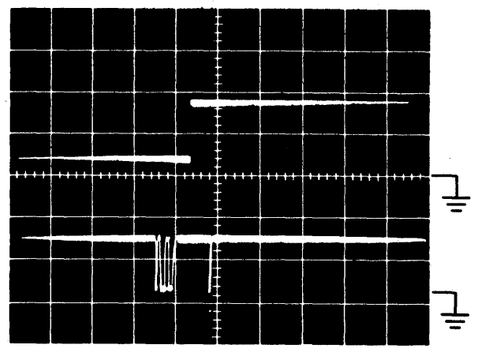
Chan 1 -Index D2U13
Chan 2 -Write A5D11

Voltage: 0.2 V/div
Time: 5 µs/div
Trigger: Internal
DC Input



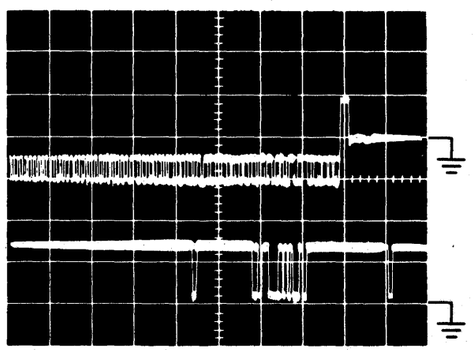
Chan 1 -Write A5D11
Chan 2 -Write Data A5B10

Voltage: 0.2 V/div
Time: 5 µs/div
Trigger: Internal
DC Input



Chan 1 Actuator I/O Line A B2D05
Chan 2 -Write Data A5B10

Voltage: 50 mV/div Chan 1
0.2 V/div Chan 2
Time: 2 µs/div
Trigger: External + A5D05 -Fast Sync
AC Input Chan 1
DC Input Chan 2



*Write Data*

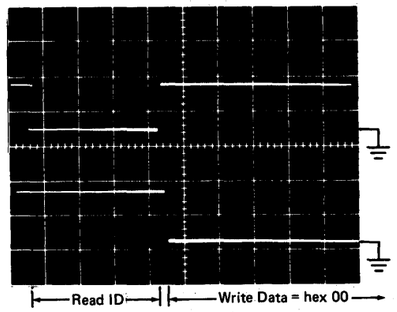
Using the disk exerciser, set up the following command:

- Select write data command.
- Select drive A or B.
- Select 0 for number of records.
- Select 0 for record number.
- Select hex 167 for cylinder number.
- Select 0 for head number.
- Select data field 1.
- Set up a data pattern of hex 0000 in data field 1.
- Select the scope loop option to execute the command.

The following waveshapes will be observed if the write circuits are working correctly and there are no servo problems.

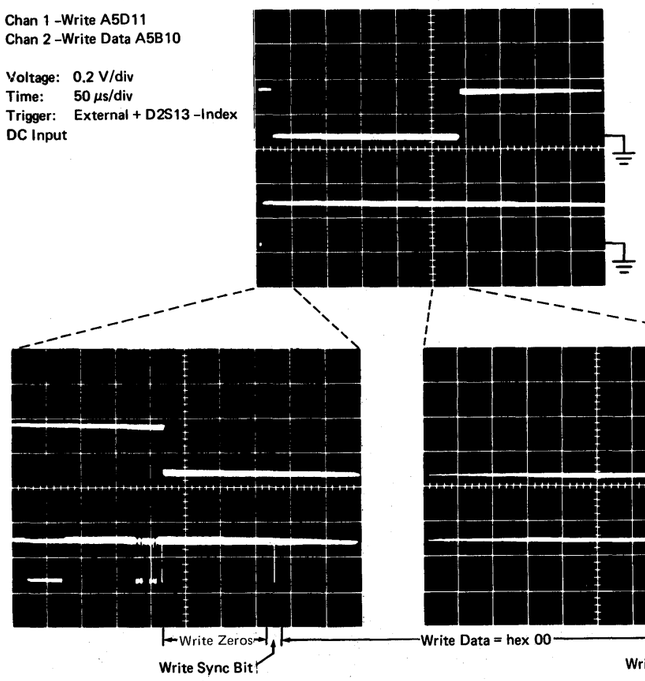
Chan 1 -Read A5D09
Chan 2 -Write A5D11

Voltage: 0.2 V/div
Time: 5 μs/div
Trigger: External + D2S13 -Index
DC Input




Chan 1 -Write A5D11
Chan 2 -Write Data A5B10

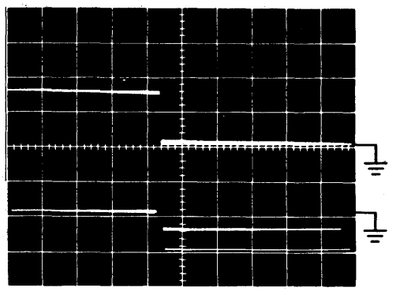
Voltage: 0.2 V/div
Time: 50 μs/div
Trigger: External + D2S13 -Index
DC Input




With X10 magnification (time), the above can be seen.

Chan 1 -Write A5D11
Chan 2 Actuator I/O Line B2D05

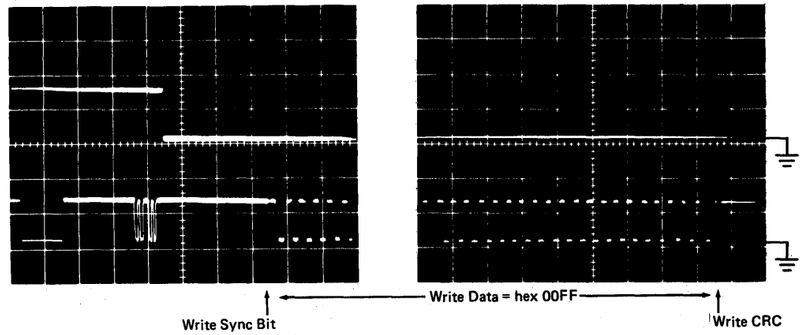
Voltage: 0.2 V/div Chan 1
50 mV/div Chan 2
Time: 5 μs/div
Trigger: External + D2S13 -Index
DC Input Chan 1
AC Input Chan 2



Set up the same write command with a data pattern of hex 00FF to observe the following:

Chan 1 -Write A5D11
Chan 2 -Write Data A5B10

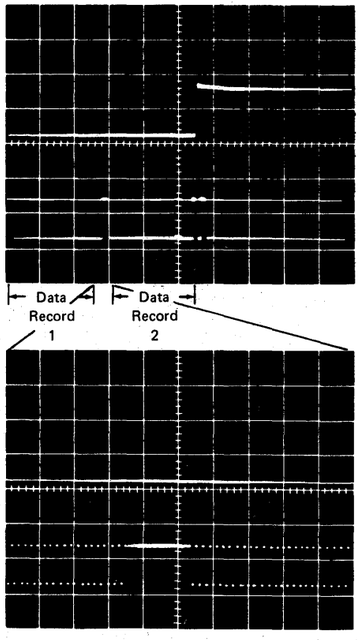
Voltage: 0.2 V/div
Time: 50 μs/div with X10 magnification
Trigger: External + D2S13 -Index




Set up the same write command but select 1 for the number of records and set up a data pattern of hex 00FF to observe the following:

Chan 1 -Write A5D11
Chan 2 -Write Data A5B10

Voltage: 0.2 V/div
Time: 100 μs/div
Trigger: External + D2S13 -Index
DC Input




With X10 magnification, the writing of data can be observed. The waveshape here shows the zeros being written after data record 1 and before data record 2.

**Read Waveshapes**

The following read waveshapes were taken with a X10 grounded probe on the disk drive board (E-A1 or E-B1), unless otherwise indicated.

*Read ID*

Using the disk exerciser, set up the following command:

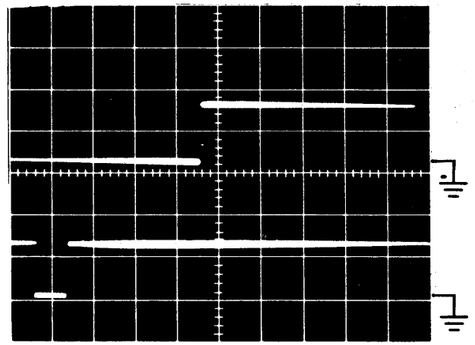
- Select read ID command.
- Select drive A or B.
- Select 0 for number of sectors.
- Select 0 for sector number.
- Select hex 167 for cylinder number.
- Select 0 for head number.
- Select the scope loop option to execute the command.

The following waveforms will be observed if the read circuits are working correctly and there are no servo problems.

*Note:* The '+read data' line waveshape shows an ID field of hex 00000167. If the sector is defective, the waveshape will be different.

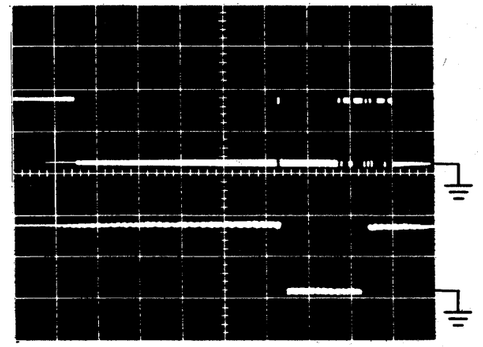
Chan 1 -Read A5D09
Chan 2 -Fast Sync A5D05

Voltage: 0.2 V/div
Time: 5 μs/div
Trigger: Internal
DC Input



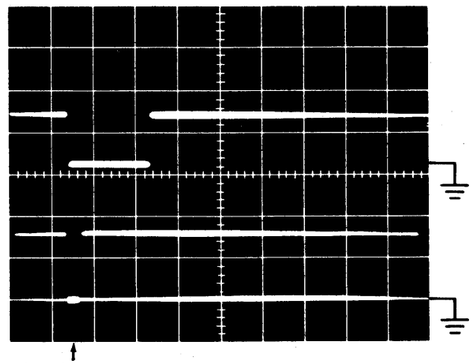
Chan 1 +Read Data A5D09
Chan 2 -Sync Bit Found A-A2C2U09

Voltage: 0.2 V/div
Time: 2 μs/div
Trigger: External + A5D09 -Read
DC Input



Chan 1 -Fast Sync A5D05
Chan 2 Read Clock A5D10

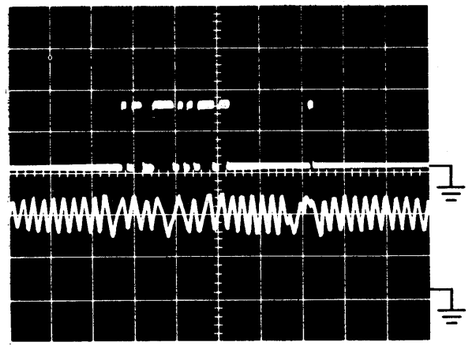
Voltage: 0.2 V/div
Time: 5 μs/div
Trigger: External + A5D09 -Read
DC Input



Fast sync causes read clock to stop and synchronize to the zeros written on the disk.

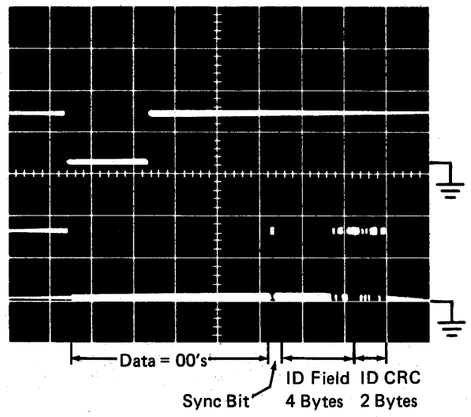
Chan 1 +Read Data A5D09
Chan 2 High Level Diff Analog Signal B2D13

Voltage: 0.2 V/div
Time: 1 μs/div
Trigger: External + A-A2C2U09 -Sync Bit Found
DC Input



Chan 1 -Fast Sync A5D05
Chan 2 +Read Data A5B08

Voltage: 0.2 V/div
Time: 2 μs/div
Trigger: External + A5D09 -Read
DC Input



*Read Data*

Using the disk exerciser, set up the following commands:

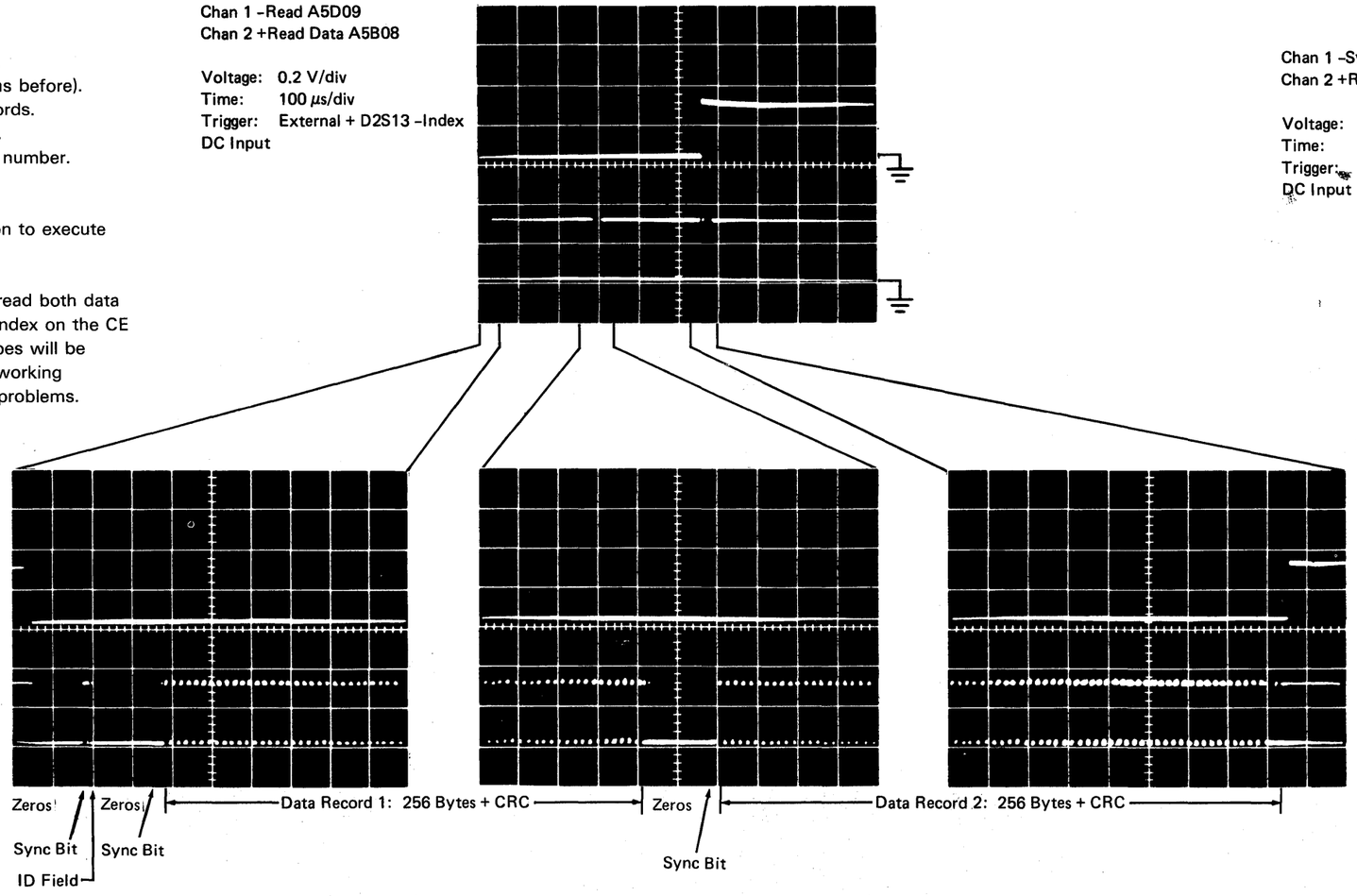
- To write a known data pattern:
  - Select write data command.
  - Select drive A or B.
  - Select 1 for number of records.
  - Select 0 for record number.
  - Select hex 167 for cylinder number.
  - Select 0 for head number.
  - Select data field 1.
  - Set up a data pattern of hex 00FF in data field 1.
  - Execute the command table.

- To read the data:
  - Select read data command.
  - Select drive A or B (same as before).
  - Select 1 for number of records.
  - Select 0 for record number.
  - Select hex 167 for cylinder number.
  - Select 0 for head number.
  - Select data field 1.
  - Select the scope loop option to execute the command.

This will cause the disk drive to read both data fields from the first sector after index on the CE cylinder. The following waveshapes will be observed if the read circuits are working correctly and there are no servo problems.

Chan 1 -Read A5D09
Chan 2 +Read Data A5B08

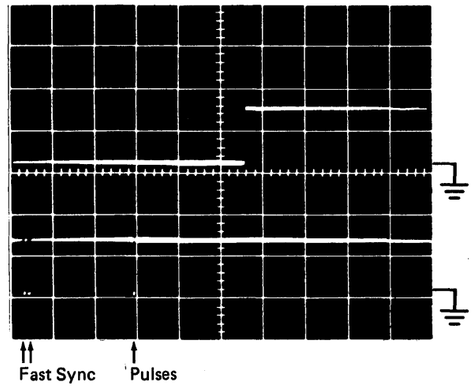
Voltage: 0.2 V/div
Time: 100 µs/div
Trigger: External + D2S13 -Index
DC Input




With X10 magnification, the ID field and hex 00FF data pattern can be seen.

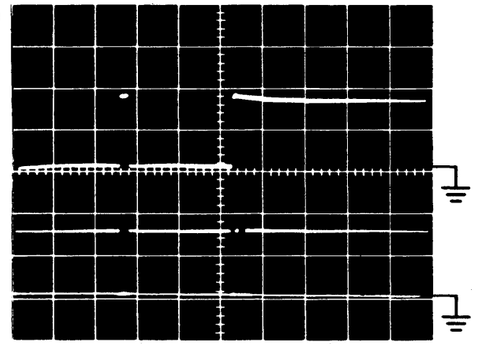
Chan 1 -Read A5D09
Chan 2 -Fast Sync A5D05

Voltage: 0.2 V/div
Time: 100 µs/div
Trigger: External + D2S13 -Index
DC Input




Chan 1 -Sync Bit Found A-A2C2U09
Chan 2 +Read Data A5B08

Voltage: 0.2 V/div
Time: 100 µs/div
Trigger: External + D2S13 -Index
DC Input



## Power On and Power Off Operations

### Power Sequence

The Power switch (on the operator panel) and contactor K1 activate the DC power supplies and the AC power to the fans. When the DC power lines are all in tolerance, the power supply activates the 'power good' line. The 'power good' line causes the disk drive to activate the brake coil (remove the brake), and reset the 'brake applied' line to the power supply. Resetting the 'brake applied' line causes contactor K2 to pick, which supplies AC power to the disk drive motor. If two disk drives are installed, the 'brake applied' line must be reset on both drives before AC power is supplied to the drives. The disk drive automatically performs a recalibrate operation when it becomes ready.

On power off, the 'power good' line is reset, which causes the disk drive to reset the brake coil driver, and the brake is applied. The data heads are located over the landing zone by a spring.

A. AC power is supplied to the fans and DC power is supplied to the disk drive when the Power switch is set to I (operator panel). When all voltages are within tolerance, the 'power good' latch is set.

B. When the disk drive receives the 'power good' signal, the disk drive resets the 'brake applied' latch, releasing the brake. AC power is then supplied to the disk drive motor.

C. The 'power on delay' line becomes active for about 20 seconds to permit the disk speed to become a stable 3,125 rpm.

D. When the 'power on delay' line times out, counter 4 is synchronized and the 'kick SS' line is set. If counter 4 does not synchronize, the brake is applied and AC voltage to the disk drive motor is removed.

E. The 'kick SS' line puts maximum current to the actuator coil for 10 milliseconds to move the data heads onto the data areas of the disk.

F. Counter 5 is synchronized during the first actuator movement.

G. The 'seek' latch is set at the same time the 'kick SS' line is set, and is reset when the recalibrate operation is complete. During the recalibrate operation, the heads move into the data area, then move behind home again, and finally settle on track 0. The 'seek' latch resets when the heads stop, and is set again when the heads move into the data area or behind home.

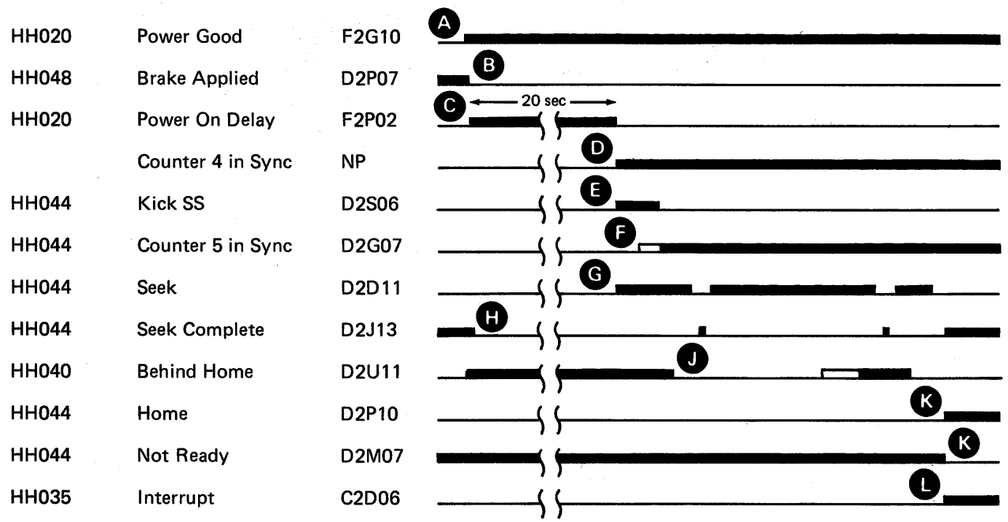
H. The 'seek complete' latch is reset at the start of the recalibrate operation and is set again when the recalibrate operation is complete.

J. The 'behind home' latch is reset whenever the actuator arm is over the data areas of the disk surface.

K. When the recalibrate operation is complete, the 'home' latch is set and the 'not ready' latch is reset.

L. When the 'not ready' latch is reset, the disk drive 'interrupt' line is set to signal the common adapter that the power-on sequence is complete.

Following the power-on sequence the common adapter will perform a seek calibration sequence. For more information, see *Seek Calibration Sequence* earlier in this section.



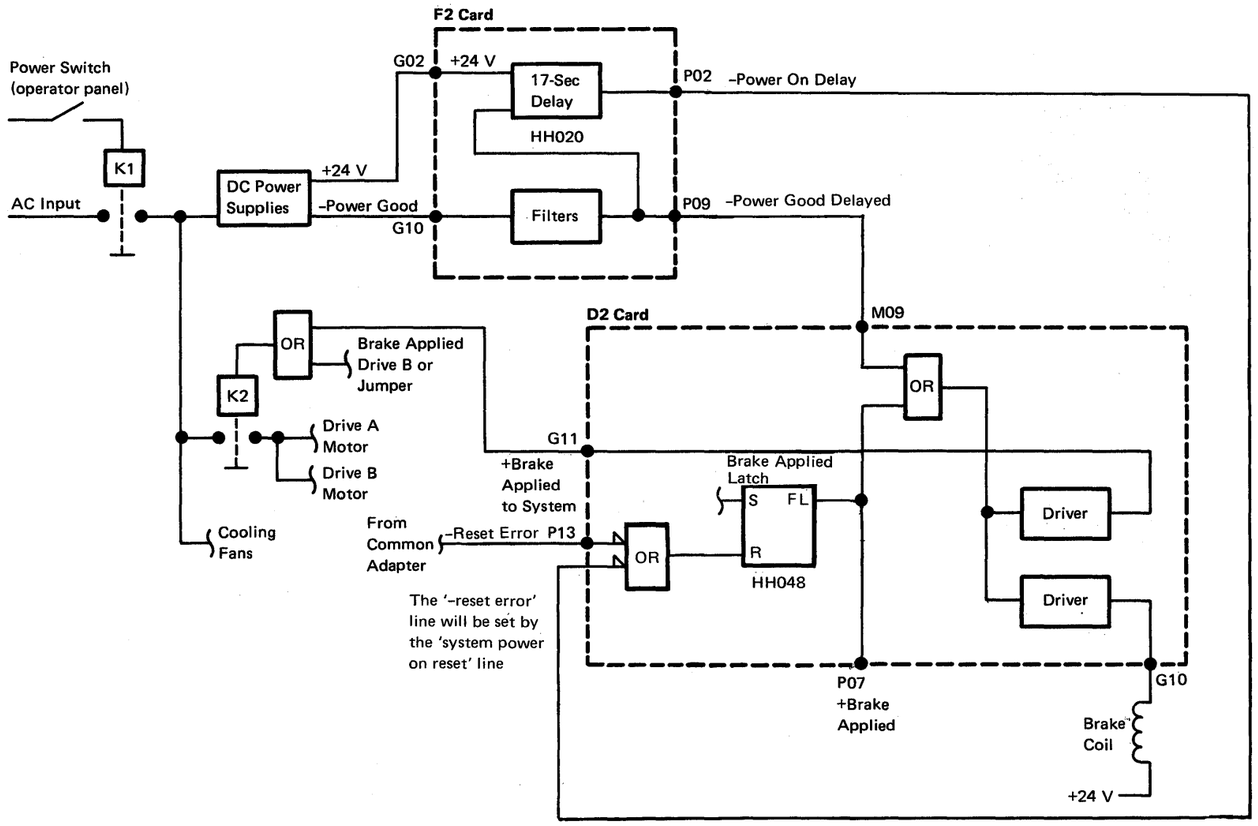
**Emergency Power Off**

If any AC or DC voltage goes out of its tolerance, the power supply resets the 'power good' line and all AC and DC voltages are removed from the disk drive.

If the disk drive senses specific errors, the heads are moved to the inner stop, and the 'brake applied' line is set, which resets the brake coil driver, and the brake is applied. The 'brake applied' line also causes the AC power to be removed from the disk drive motors. The DC voltages are not removed because of these errors. The 'brake applied' line is set if:

- The disk is not at the correct speed when the 'power on delay' line is reset.
- Servo signals are not received from the disk.
- The write clock controls cannot synchronize to the servo signals.

For more information, see *Disk Drive Sensed Errors* later in this section.



## Control Storage Initial Program Load

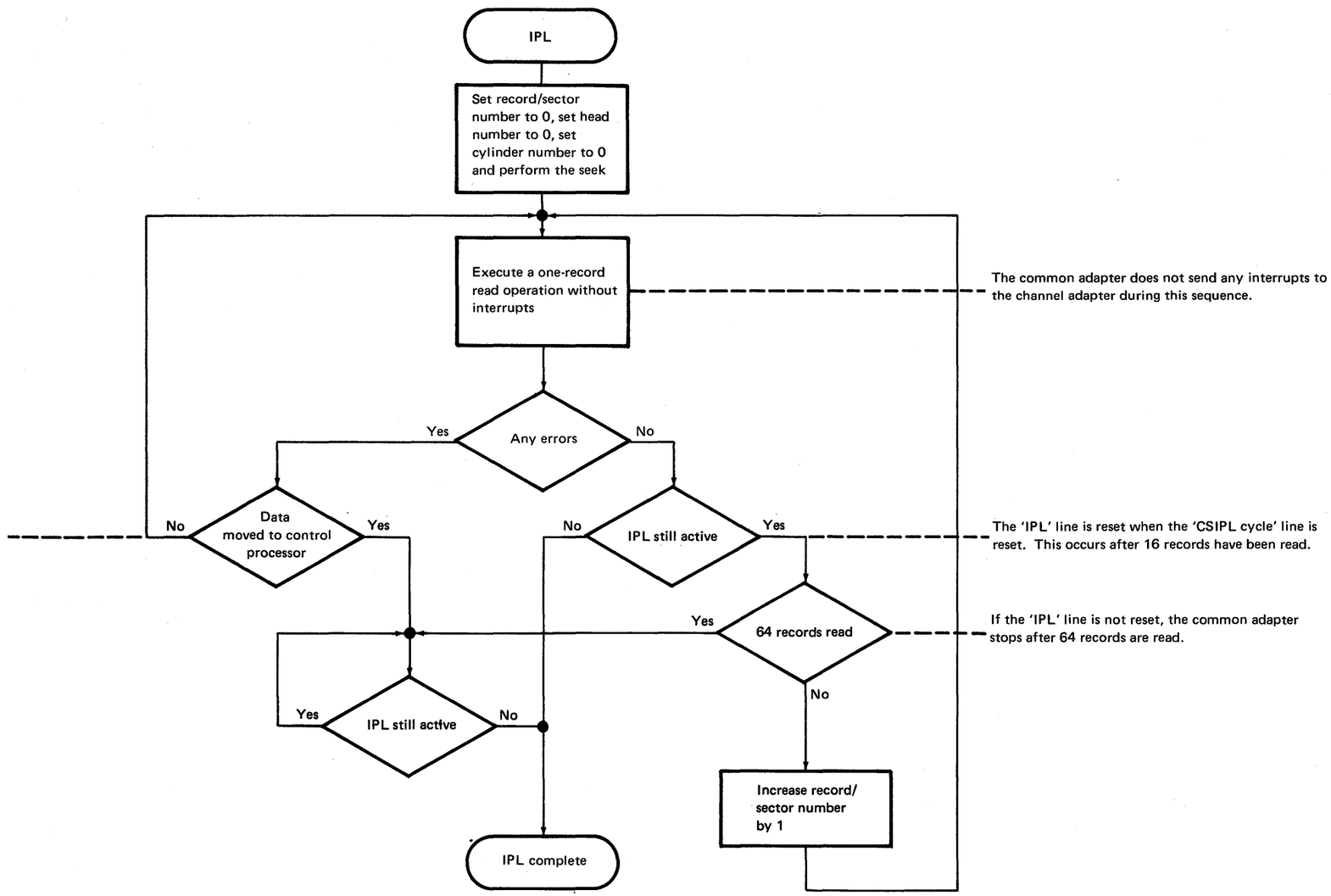
The control storage initial program load is started when the Load switch is pressed while the disk CSIPL is selected. This resets any disk operation now executing and executes a recalibrate operation. Sixteen records (0 through 15) from head 0 cylinder 0 are read into control storage in the processing unit. Reading is started at the index pulse and the data records are read sequentially (one record is read each revolution of the disk) until the 'CSIPL cycle' line is reset. The data is counted by the control processor, which resets the 'CSIPL cycle' line when 4,096 bytes are received.

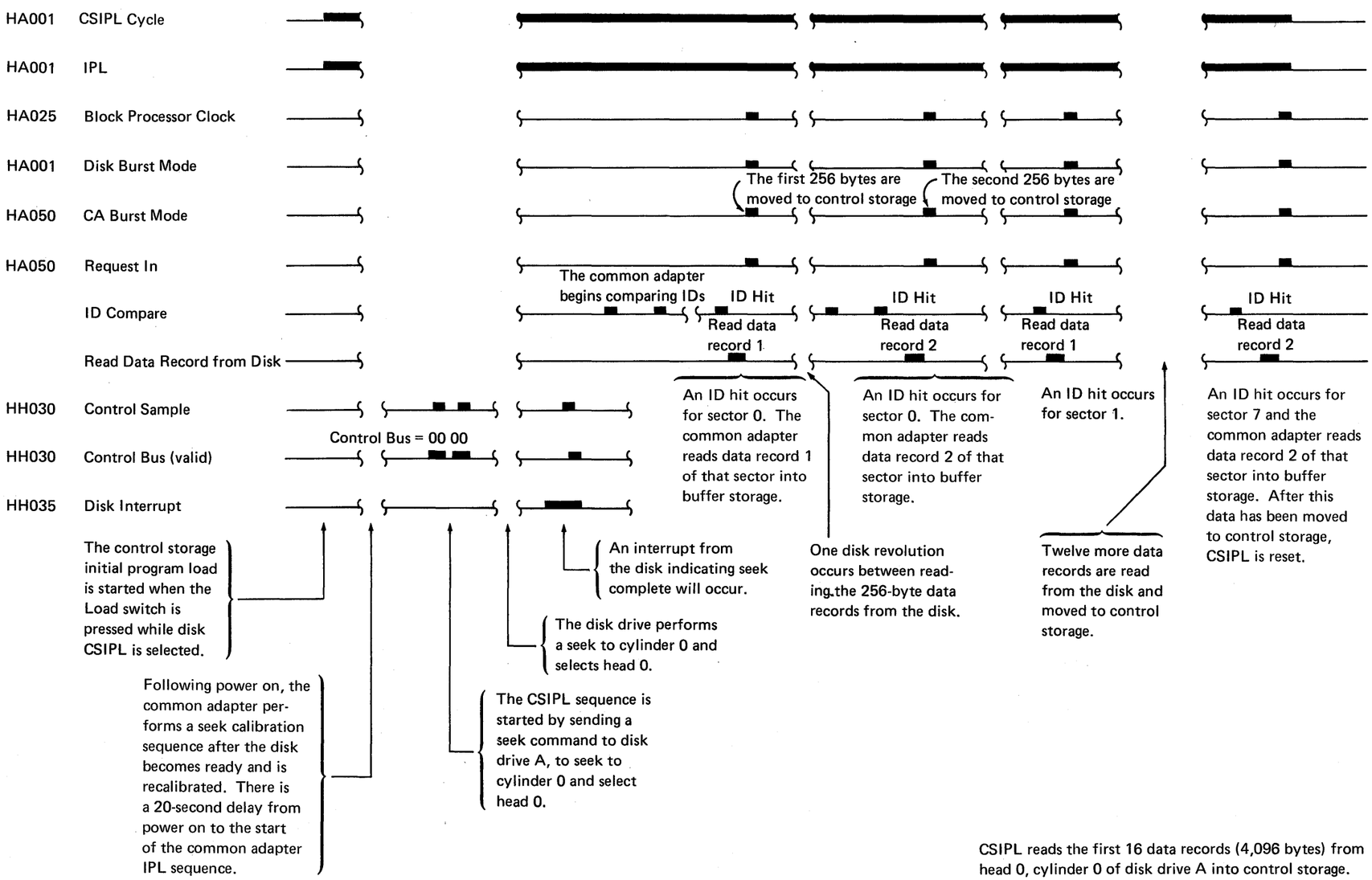
If an error occurs during IPL and the data has not been moved to the control processor, the common adapter retries the operation. If data has been moved, recovery is not possible.

### Common Adapter Initial Program Load Operation

The channel adapter decodes the CSIPL lines and sets the '-IPL' line to the common adapter. This causes the common adapter to execute the initial program load operation that corresponds to the CSIPL described above. The flowchart shows the common adapter functions that are performed.

**Common Adapter Initial Program Load Sequence**






CSIPL reads the first 16 data records (4,096 bytes) from head 0, cylinder 0 of disk drive A into control storage.

## Command Bus In

The command bus in is used by the channel adapter during data cycle steal and during I/O operations. The meaning of the command bus in during data cycle steal is:

Bit 0 Increase the cycle steal data storage address.

- Bit 0=0 Do not increase the storage address.
- Bit 0=1 Increase the storage address by 1.

Bit 1 Direction of the data cycle steal.

- Bit 1=0 Move data from the disk attachment to the control processor.
- Bit 1=1 Move data from the control processor to the disk attachment.

Bit 4 Main or control storage select for the data cycle steal.

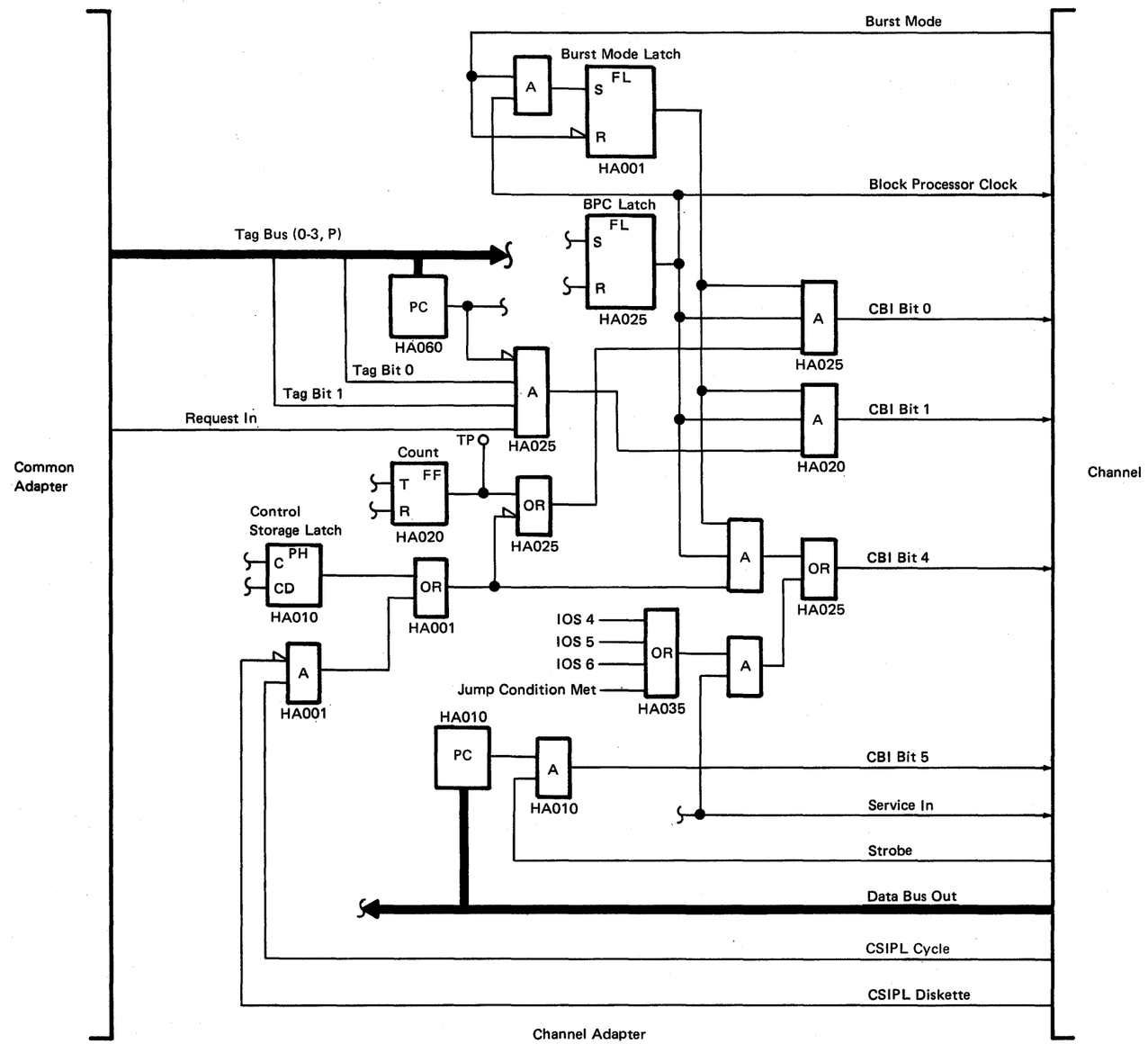
- Bit 4=0 Main storage.
- Bit 4=1 Control storage.

The meaning of the command bus in during I/O operations is:

Bit 4 Jump condition met for jump on I/O commands, and do not check for parity on the data bus in for I/O sense and I/O control sense commands.

Bit 5 Data bus out parity check.

Command bus in bits 2 and 3 are not powered by the disk attachment, and both should be zero for 62PC disk cycle steal and I/O operations.



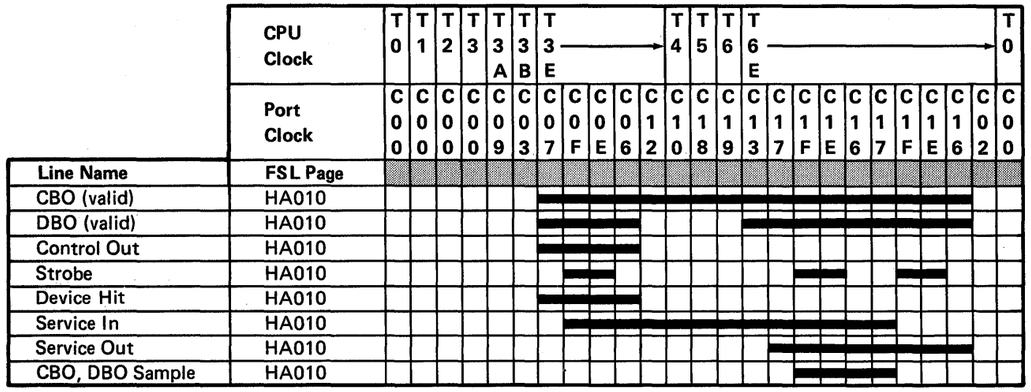
This page intentionally left blank.

## COMMANDS

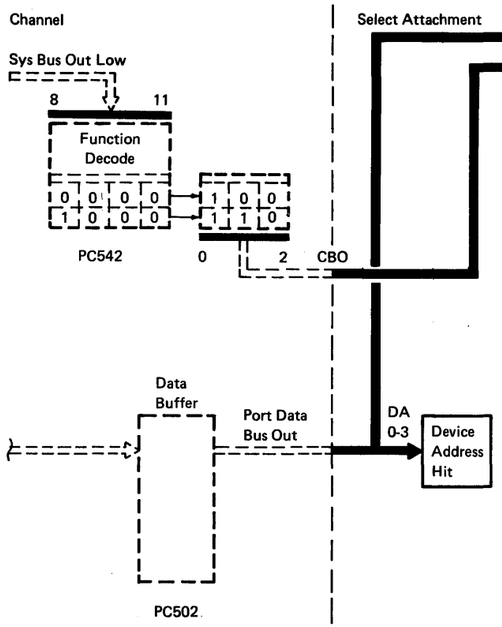
### I/O Load Command—I/O Control Load Command

These commands send diagnostic, interrupt, and operation control information from the control processor to the attachment. The commands are used to set up and start disk operations, and to move interrupt control and diagnostic control information to the channel adapter. Combinations of I/O load and I/O control load commands are used to perform common adapter commands.

WTCH and WTCL I/O storage commands are decoded as IOL commands by the channel adapter and are also used to load the file control block.




*Note:* If the 'request out' line is active at the beginning of the command, the 'device hit' line does not become active until the 'request out' line is not active. The 'service in' line becomes active during the first 'strobe' pulse after the 'device hit' line is active.




Select Attachment

Send Data Byte to Attachment

DB0 0-7

Mod 4-7

| 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 |
|---|---|---|---|---|---|---|---|---|---|
| 1 0 0 I/O Load | 0 0 0 0 Load File Control Block High Byte | File control block high byte | | | | | | | |
| | 0 0 0 1 Load File Control Block Low Byte* | File control block low byte | | | | | | | |
| | 0 0 1 1 Reset Interrupt Request (level 4) | | | | | | | | |
| | 0 1 0 1 Load Tag Register and Transfer to Common Adapter* | Tag byte | | | | | | | |
| | 0 1 1 0 Enable Interrupt Request (level 4) | | | | | | | | |
| | 1 0 0 0 (same as 0000) | File control block high byte | | | | | | | |
| | 1 0 0 1 (same as 0001)* | File control block low byte | | | | | | | |
| | 1 0 1 0 Disable Interrupt Request (level 4) | | | | | | | | |
| | 1 1 1 1 Reset Error Check Latches | | | | | | | | |
| 1 1 0 I/O Control Load | 0 0 0 0 Load Tag Register | Tag byte | | | | | | | |
| | 0 0 0 1 Load Control Latches | Set/reset diagnostic mode | Set/reset disk I/O working | Set/reset control storage select | | Reset channel adapter | | | |
| | 0 1 0 0 Reset Channel Busy Latch | | | | | | | | |
| | 1 0 0 0 Load Channel Adapter Diagnostic Control Latches | Set/reset diagnostic request in | Set/reset diagnostic acknowledge request out | Set/reset diagnostic strobe in | Set/reset diagnostic common adapter burst mode | Set/reset channel adapter DBO even parity | Set/reset diagnostic clock | | |
| | 1 0 1 1 Set Data Pending Latch | | | | | | | | |
| | 1 1 0 0 Reset Data Pending Latch | | | | | | | | |

Bits 0 and 4 on at the same time activate the 'power on reset' line to the common adapter.

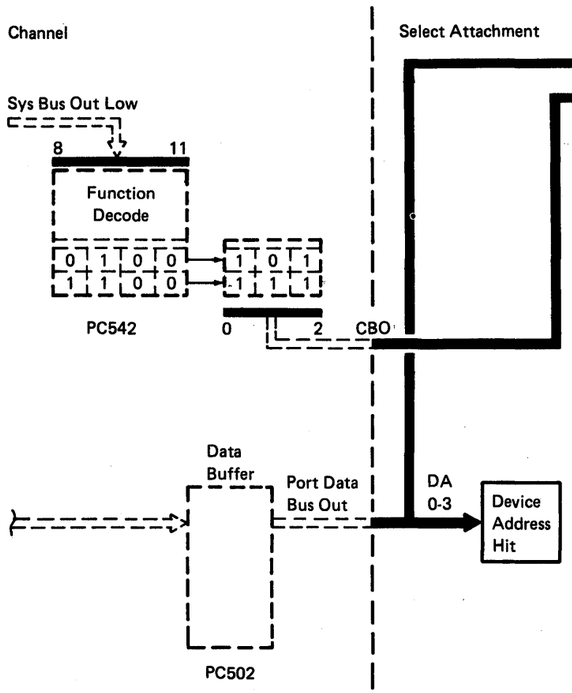
*See *Common Adapter Commands* later in this section.

## I/O Sense Command—I/O Control Sense Command

These commands are used to move 1 byte of control, error, or diagnostic information from the attachment to the control processor. Commands that sense information from the common adapter must be preceded by an IOCL 0 to load the channel adapter tag register. Sense information from the common adapter or the channel adapter is sent to the control processor on the DBI.

RDCH and RDCL I/O storage commands are decoded as IOS commands by the channel adapter and are also used to sense the file control block.

| CPU Clock | T0 | T1 | T2 | T3 | T3A | T3B | T3E → | | | | | | | T4 | T5 | T6 | T6E → | | | | | | T0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Port Clock | C000 | C000 | C000 | C000 | C009 | C003 | C007 | C00F | C00E | C006 | C007 | C00F | C00E | C006 | C012 | C110 | C118 | C119 | C113 | C117 | C11F | C11E | C116 | C002 | C000 |
| **Line Name** | **FSL Page** | | | | | | | | | | | | | | | | | | | | | | | | |
| CBO (valid) | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| DBO (valid) | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| Control Out | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| Strobe | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| Device Hit | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| Service In | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| Service Out | HA010 | | | | | | | | | | | | | | | | | | | | | | | | |
| DBI (valid) | HA070 | | | | | | | | | | | | | | | | | | | | | | | | |




| 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 |
|---|---|---|---|---|---|---|---|---|---|
| 1 0 1 I/O Sense | 0 0 0 1 Sense FCB High Byte* | File control block high byte | | | | | | | |
| | 0 0 1 0 Sense FCB Low Byte* | File control block low byte | | | | | | | |
| | 0 1 0 0 Diagnostic Sense 2 | IL4 data pending latch | Channel busy latch | | Storage select latch | IL4 request pending latch | IL4 enable latch | FCB load retry latch | |
| | 0 1 0 1 Channel Adapter Check Status | Adapter check | Overrun check | DBO parity check | Tag parity check | Common adapter data bus parity check | Interface error | | |
| | 0 1 1 0 Diagnostic Sense 1 | Request out | Acknowledge request in | Strobe out | | Block processor clock | Diagnostic mode latch | | |
| | 1 0 0 1 (same as 0001)* | File control block high byte | | | | | | | |
| | 1 0 1 0 (same as 0010)* | File control block low byte | | | | | | | |
| 1 1 1 I/O Control Sense | 0 0 0 0 Sense Tag Register | Tag byte | | | | | | | |
| | 1 0 0 0 Sense Data In Buffer High Byte | Data buffer high byte | | | | | | | |
| | 1 0 0 1 Sense Data In Buffer Low Byte | Data buffer low byte | | | | | | | |

DBI

Send Data Byte to Channel

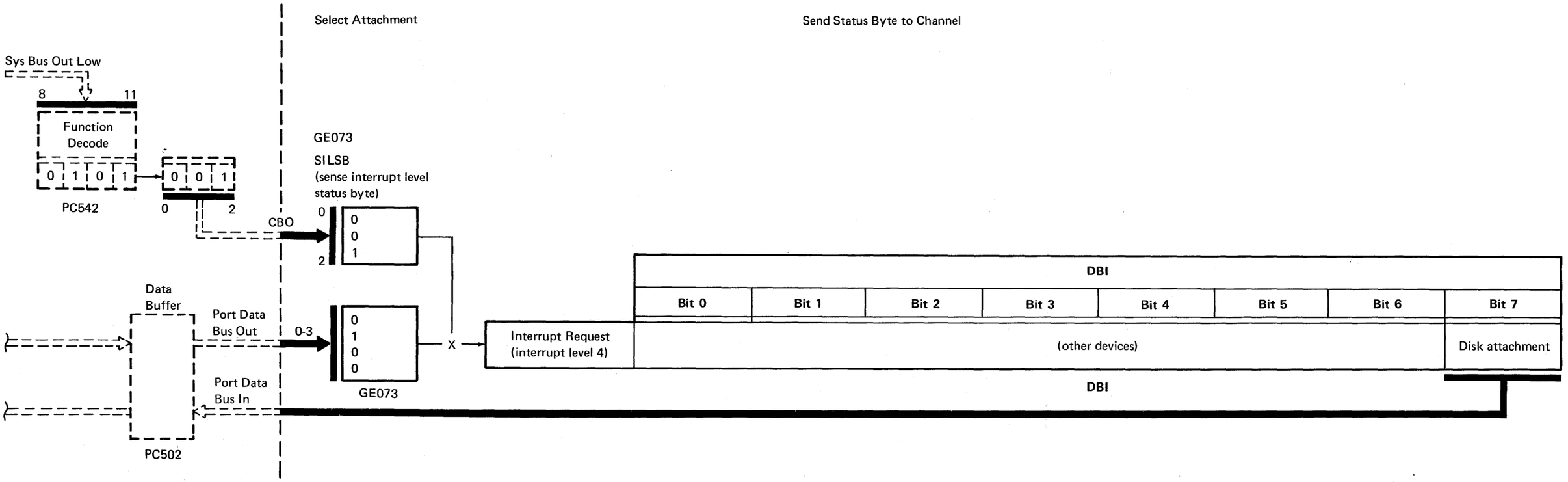
*See *Common Adapter Commands* later in this section

## Sense Interrupt Level Status Byte Command

This command tests all devices on the system to determine which device is requesting an interrupt for a specific interrupt level. If the SILSB is for interrupt level 4, and if the disk attachment is requesting an interrupt, DBI bit 7 will be active. Other data bus bits are set if other I/O devices are requesting an interrupt.

| | CPU Clock | T0 | T1 | T2 | T3 | T3A | T3B | T3E → | T4 | T5 | T6 | T6E → | T0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Port Clock | C000 | C000 | C000 | C009 | C003 | C007 | C00F C00E C006 C007 C00F C00E C006 C007 C00F C00E C006 C012 | C110 | C118 | C119 | C113 C117 C11F C11E C116 C102 | C000 |
| Line Name | FSL Page | | | | | | | | | | | | |
| CBO (valid) | HA010 | | | | | | | | | | | | |
| DBO (valid) | HA010 | | | | | | | | | | | | |
| Control Out | HA010 | | | | | | | | | | | | |
| Strobe | HA010 | | | | | | | | | | | | |
| Interrupt Hit* | HA010 | | | | | | | | | | | | |
| Service In | HA010 | | | | | | | | | | | | |
| Service Out | HA010 | | | | | | | | | | | | |
| Multidevice Response | HA010 | | | | | | | | | | | | |
| DBI (bit 7 valid)* | HA070 | | | | | | | | | | | | |
| CBI (bit 4 valid)* | HA025 | | | | | | | | | | | | |

*Active only if interrupt level 4 is set in the disk attachment.

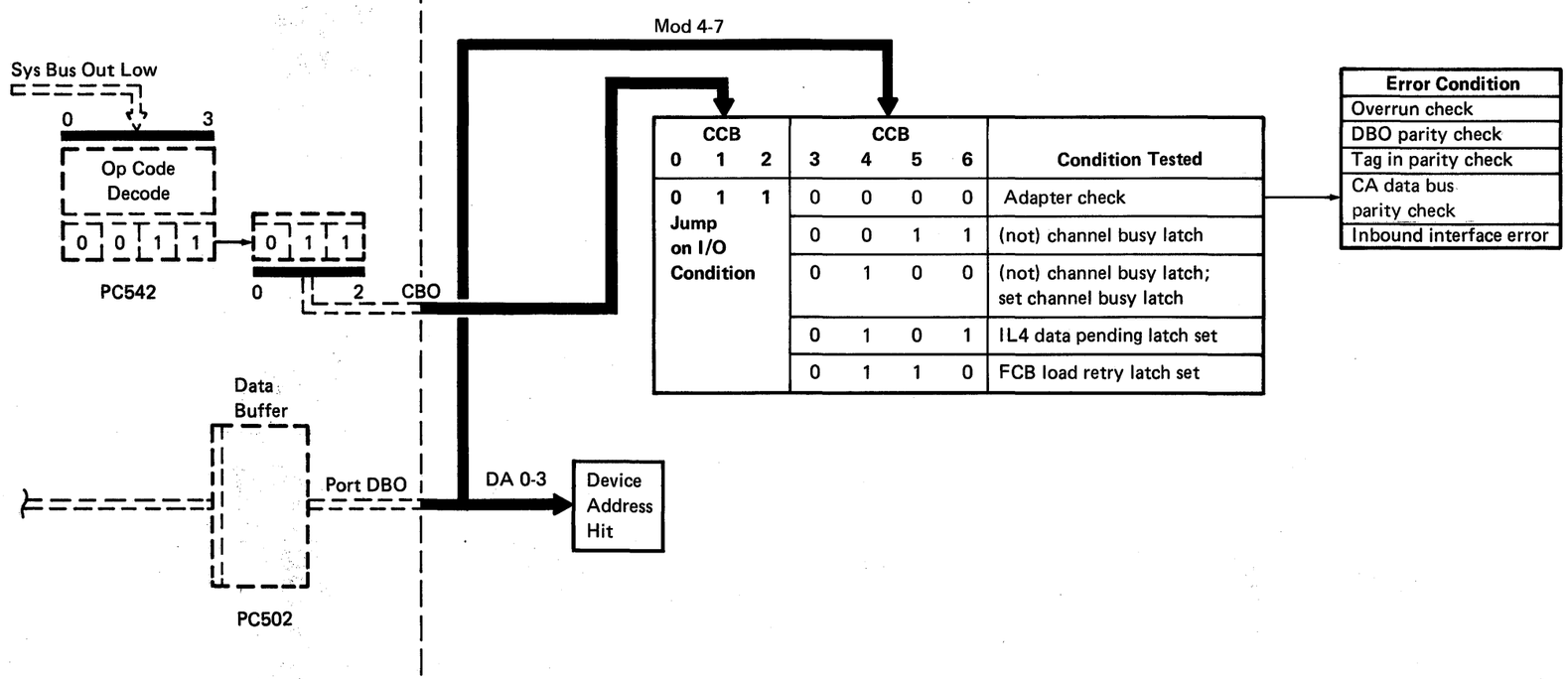


## Jump on I/O Condition Command

This command tests the channel adapter for status and error conditions. If the condition tested for is present, command bus in bit 4 is set to the control processor. If the condition is not present, bit 4 is not set.

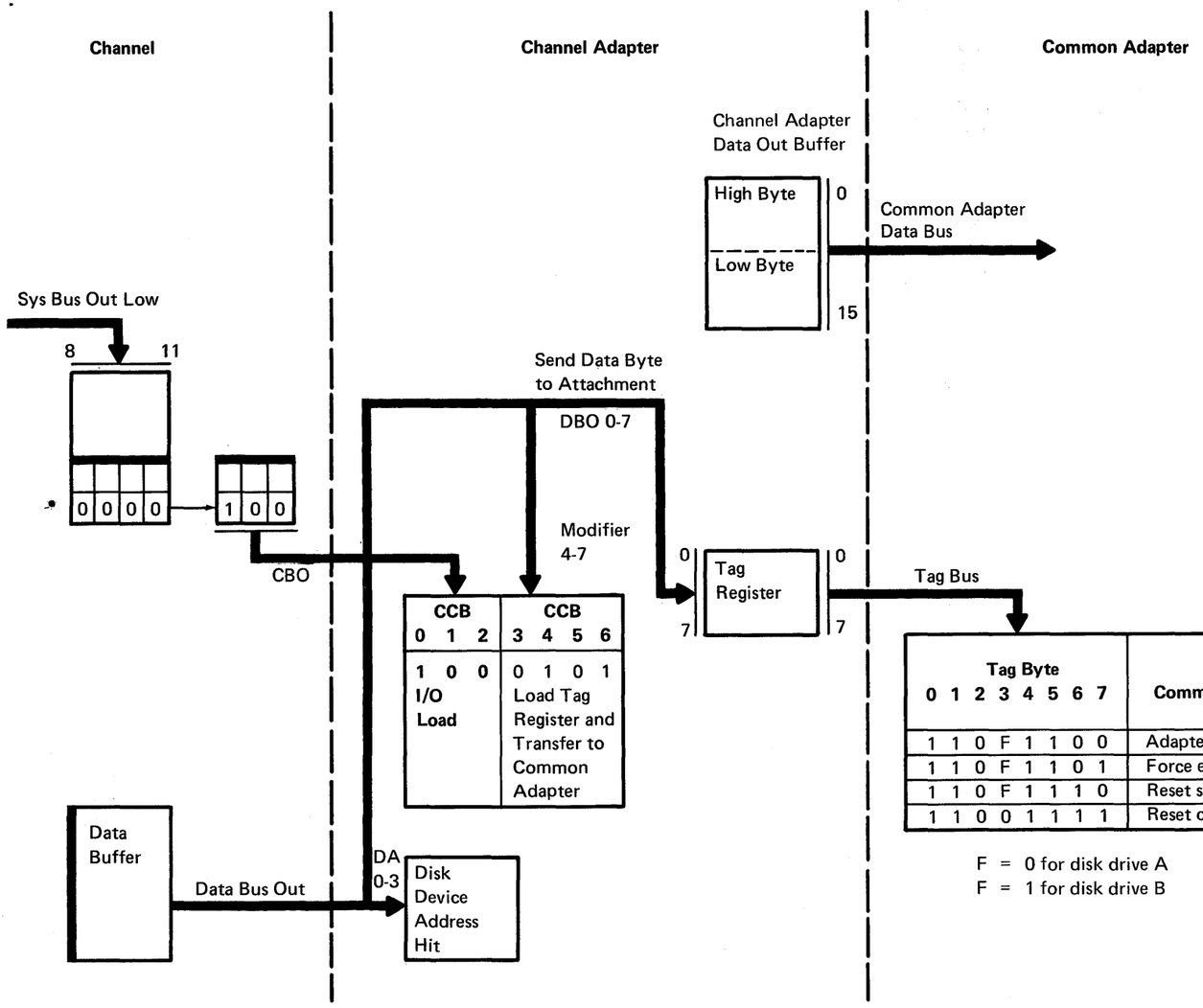
| CPU Clock | T0 | T1 | T2 | T3 | T3A | T3B | T3→E | T4 | T5 | T6 | T6→E | T0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Port Clock | C000 | C000 | C000 | C009 | C003 | C007 | C00F→C006 | C00F→C012 | C010 | C018 | C019→C016 | C002 | C000 |
| **Line Name** / **FSL Page** | | | | | | | | | | | | | |
| CBO (valid) | HA010 | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | |
| DBO (valid) | HA010 | | | | | | ■ | ■ | | | | | |
| Control Out | HA010 | | | | | | ■ | ■ | | | | | |
| Strobe | HA010 | | | | | | ■ | ■ | | | ■ | | |
| Device Hit | HA010 | | | | | | ■ | ■ | | | | | |
| Service In | HA010 | | | | | | | ■ | ■ | ■ | | | |
| Service Out | HA010 | | | | | | | | | | ■ | ■ | |
| CBI (bit 4 valid)* | HA025 | | | | | | | ■ | ■ | ■ | | | |

*Active only if the jump condition is met.



## Common Adapter Commands

The common adapter commands load and sense the common adapter file control block and diagnostic words, and perform control functions such as start and reset. The tag register and the data out buffer in the channel adapter must be loaded before a common adapter load command can be done. The IOL 1 or 9 command moves the data out buffer to the common adapter for loading the file control block or diagnostic words. The tag register in the channel adapter must be loaded before a common adapter sense command is executed. The IOS 1 or 9 command senses the high byte of a file control block or diagnostic word, and the IOS 2 or A command senses the low byte. The IOL 5 command performs common adapter control functions. For more information, see *Loading and Sensing the Common Adapter File Control Block* earlier in this section.

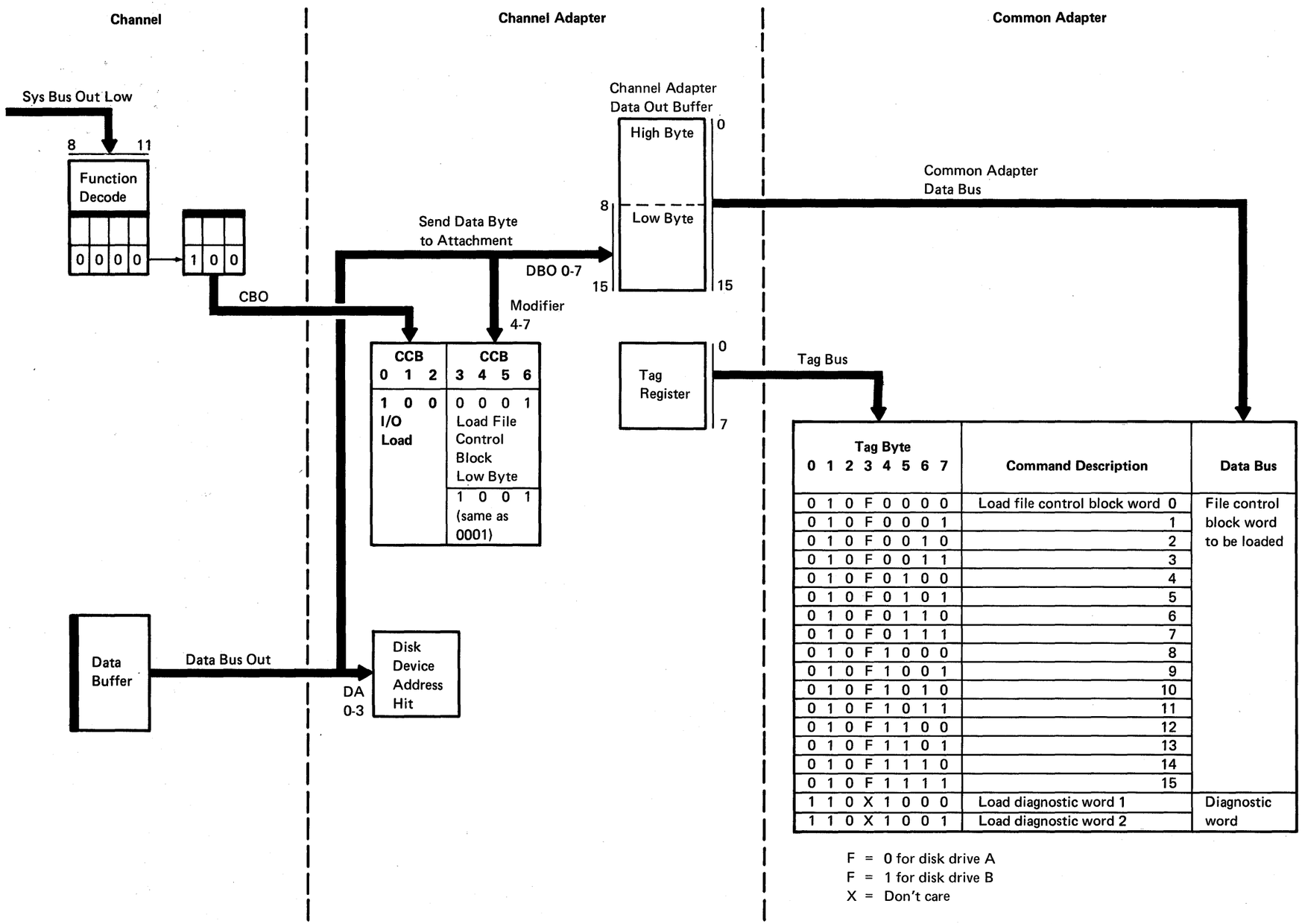



| Tag Byte 0 1 2 3 4 5 6 7 | Command Description |
|---|---|
| 1 1 0 F 1 1 0 0 | Adapter start |
| 1 1 0 F 1 1 0 1 | Force end of operation |
| 1 1 0 F 1 1 1 0 | Reset selected disk |
| 1 1 0 0 1 1 1 1 | Reset common adapter |

F = 0 for disk drive A
F = 1 for disk drive B

## Common Adapter Commands (continued)

For an IOL 1 or 9 command or an IOL 5 command, the channel adapter sets the 'request out' line to the common adapter active when the 'service in' line becomes not active. When the 'acknowledge request out' line is set active by the common adapter, the channel adapter puts the data on the common adapter data bus and sets the 'strobe out' line active for 200 nanoseconds. The 'request out' line then becomes not active, followed by the 'acknowledge request out' line.

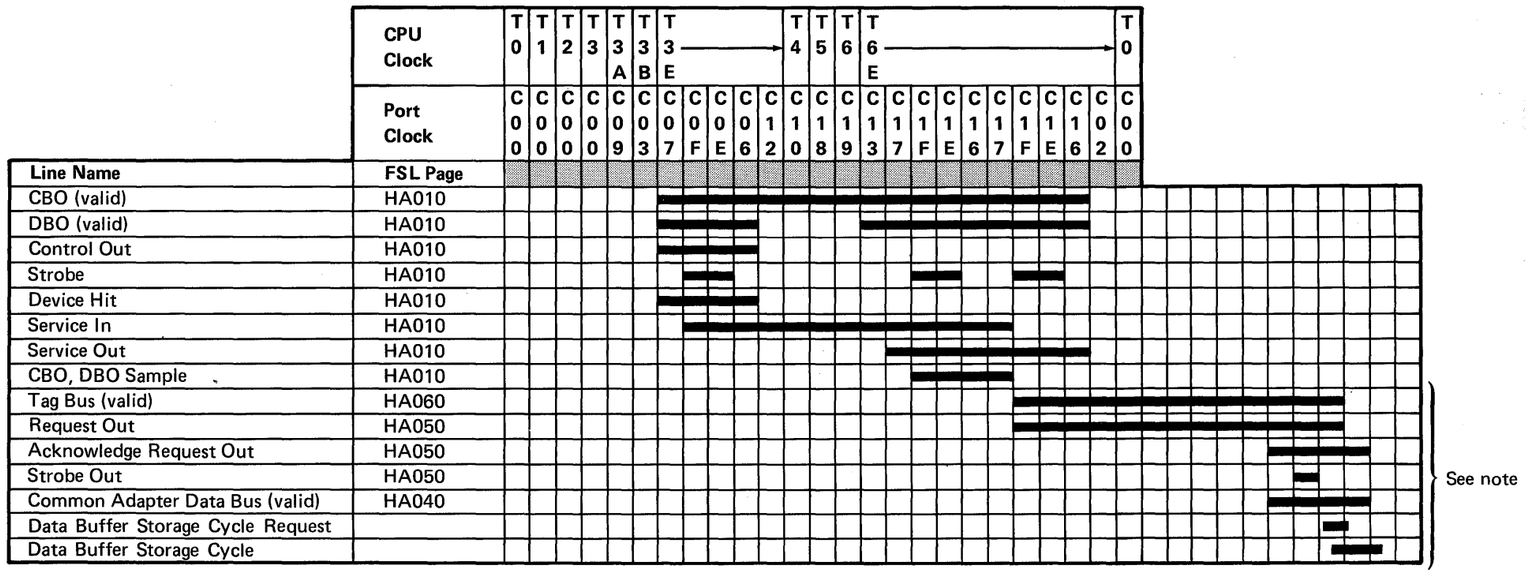



| Tag Byte 0 1 2 3 4 5 6 7 | Command Description | Data Bus |
|---|---|---|
| 0 1 0 F 0 0 0 0 | Load file control block word 0 | File control block word to be loaded |
| 0 1 0 F 0 0 0 1 | 1 | |
| 0 1 0 F 0 0 1 0 | 2 | |
| 0 1 0 F 0 0 1 1 | 3 | |
| 0 1 0 F 0 1 0 0 | 4 | |
| 0 1 0 F 0 1 0 1 | 5 | |
| 0 1 0 F 0 1 1 0 | 6 | |
| 0 1 0 F 0 1 1 1 | 7 | |
| 0 1 0 F 1 0 0 0 | 8 | |
| 0 1 0 F 1 0 0 1 | 9 | |
| 0 1 0 F 1 0 1 0 | 10 | |
| 0 1 0 F 1 0 1 1 | 11 | |
| 0 1 0 F 1 1 0 0 | 12 | |
| 0 1 0 F 1 1 0 1 | 13 | |
| 0 1 0 F 1 1 1 0 | 14 | |
| 0 1 0 F 1 1 1 1 | 15 | |
| 1 1 0 X 1 0 0 0 | Load diagnostic word 1 | Diagnostic word |
| 1 1 0 X 1 0 0 1 | Load diagnostic word 2 | |

F = 0 for disk drive A
F = 1 for disk drive B
X = Don't care

**Load FCB Command**

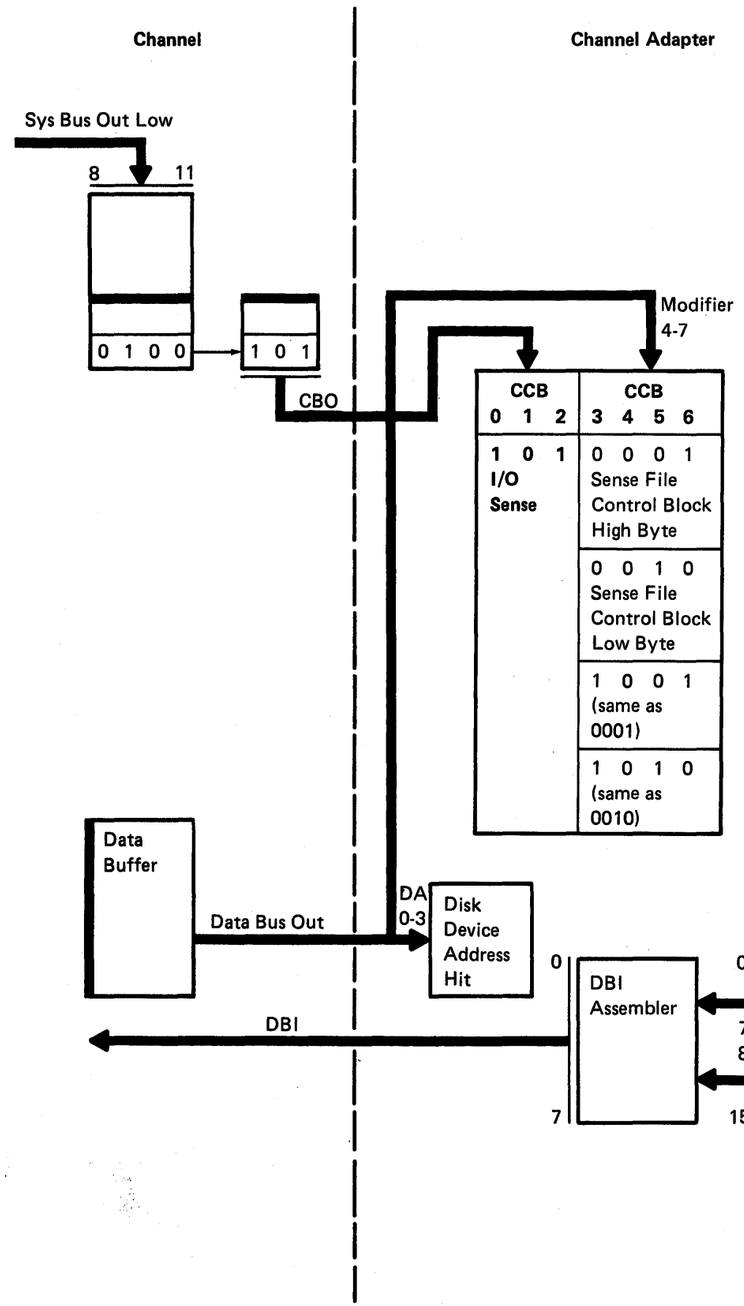
| State Count Changed by the Following Conditions | Clock | State Count | State |
|---|---|---|---|
| | | 000 | Idle, waiting for a command |
| The 'request out' line is active. | — | 00X<br>001 | Time delay |
| The 'strobe out' line is active and the 'tag bus' lines = write FCB. | — | X01<br>101 | Waiting for the next T4 |
| Wait for T4. | T4 | 1X1<br>111 | Requesting an FCB cycle |
| FCB cycle is active on the channel interface. | T8 | 11X<br>110 | Performing a write FCB cycle |
| FCB cycle is complete. | T8 | X10<br>010 | Waiting |
| End of load FCB command. Data transmission is complete. | T9 | 0X0<br>000 | End of command |




*Note:* These seven lines show timings that are in the common adapter and are not referenced to the CPU clocks or port clocks.

## Common Adapter Commands (continued)

For an IOS sense file control block or diagnostic word command, the channel adapter sets the 'request out' line to the common adapter when the 'control out' line becomes active. While the 'request out' line is active, the 'device hit' line is turned off. When the common adapter has found the requested data, it puts the data on the common adapter data bus and the 'acknowledge request out' line is set. This is followed by a 200-nanosecond 'strobe in' pulse which resets the 'request out' line, which turns on the 'device hit' line, so that the 'service in' line is set on the next 'strobe' pulse and the sense command is completed normally.

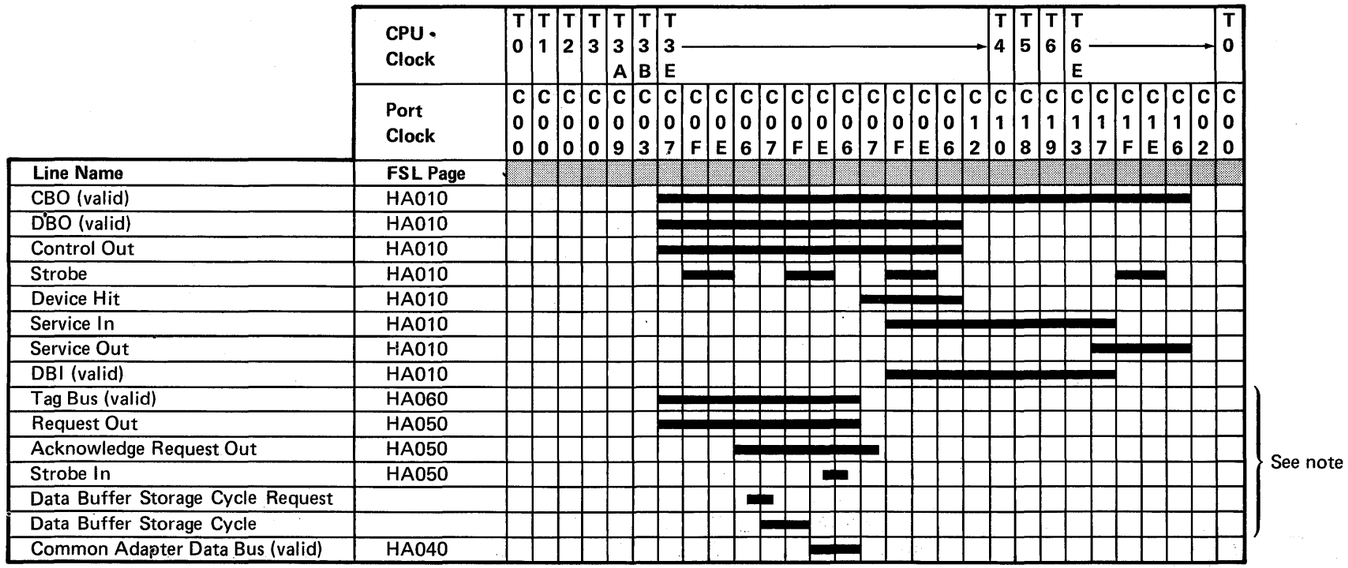



| Tag Byte 0 1 2 3 4 5 6 7 | Command Description | Data Bus |
|---|---|---|
| 0 0 0 F 0 0 0 0 | Sense file control block word 0 | File control block word to be sensed |
| 0 0 0 F 0 0 0 1 | 1 | |
| 0 0 0 F 0 0 1 0 | 2 | |
| 0 0 0 F 0 0 1 1 | 3 | |
| 0 0 0 F 0 1 0 0 | 4 | |
| 0 0 0 F 0 1 0 1 | 5 | |
| 0 0 0 F 0 1 1 0 | 6 | |
| 0 0 0 F 0 1 1 1 | 7 | |
| 0 0 0 F 1 0 0 0 | 8 | |
| 0 0 0 F 1 0 0 1 | 9 | |
| 0 0 0 F 1 0 1 0 | 10 | |
| 0 0 0 F 1 0 1 1 | 11 | |
| 0 0 0 F 1 1 0 0 | 12 | |
| 0 0 0 F 1 1 0 1 | 13 | |
| 0 0 0 F 1 1 1 0 | 14 | |
| 0 0 0 F 1 1 1 1 | 15 | |
| 1 0 0 X 1 0 0 0 | Sense diagnostic word 1 | Diagnostic word |
| 1 0 0 X 1 0 0 1 | Sense diagnostic word 2 | |

F = 0 for disk drive A
F = 1 for disk drive B
X = Don't care

**Sense FCB Command**

| State Count Changed by the Following Conditions | Clock | State Count | State |
|---|---|---|---|
| | | 000 | Idle, waiting for a command |
| The 'request out' line is active. | — | 00X<br>001 | Time delay |
| The 'acknowledge request out' line is active and the 'tag bus' lines = read FCB. | T4 | X01<br>101 | Requesting an FCB cycle |
| FCB cycle is active on the channel interface. | T8 | 1X1<br>111 | Performing a read FCB cycle |
| FCB cycle is complete. | TA | X11<br>011 | Sending a 'strobe in' pulse to the system |
| The 'strobe in' line is active. | — | 01X<br>010 | Waiting for the command to end |
| End of sense FCB command. Data transmission is complete. | T9 | 0X0<br>000 | End of command |




*Note:* These six lines show timings that are in the common adapter and are not referenced to the CPU clocks or port clocks.

**Common Adapter Command Descriptions**

*Load and Sense File Control Block Commands:* These commands permit one of the 16 file control block words in the common adapter to be loaded or sensed.

*Load and Sense Diagnostic Word 1 Commands:* These commands load or sense one word of data to or from the common adapter channel data buffer register.

*Load and Sense Diagnostic Word 2 Commands:* These commands are used to operate the common adapter and disk interface at an instruction cycle rate for diagnostic purposes.

*Adapter Start Command:* This command sets a bit in the poll register to instruct the common adapter to execute the operation indicated by the file control block of the selected disk. Any error bits in the file control block are reset before the operation is started.

*Force End Operation Command:* This command causes the common adapter to terminate any operation now executing on the selected disk drive, and to send an end operation interrupt to the control processor after the status is set in the FCB.

*Reset Selected Disk Command:* This command pulses the 'reset error' line to the selected disk drive, stops any operation now executing to that disk drive, and sets the status in the FCB. The common adapter sends a recalibrate command to any disk drive that is not ready.

*Reset Common Adapter Command:* This command resets hardware error conditions, the buffer address registers, and the poll register. This command also stops any operation now executing and pulses the 'reset error' line to all disk drives. This command does not reset the FCBs or the data buffer. The common adapter sends a recalibrate command to any disk drive that is not ready.

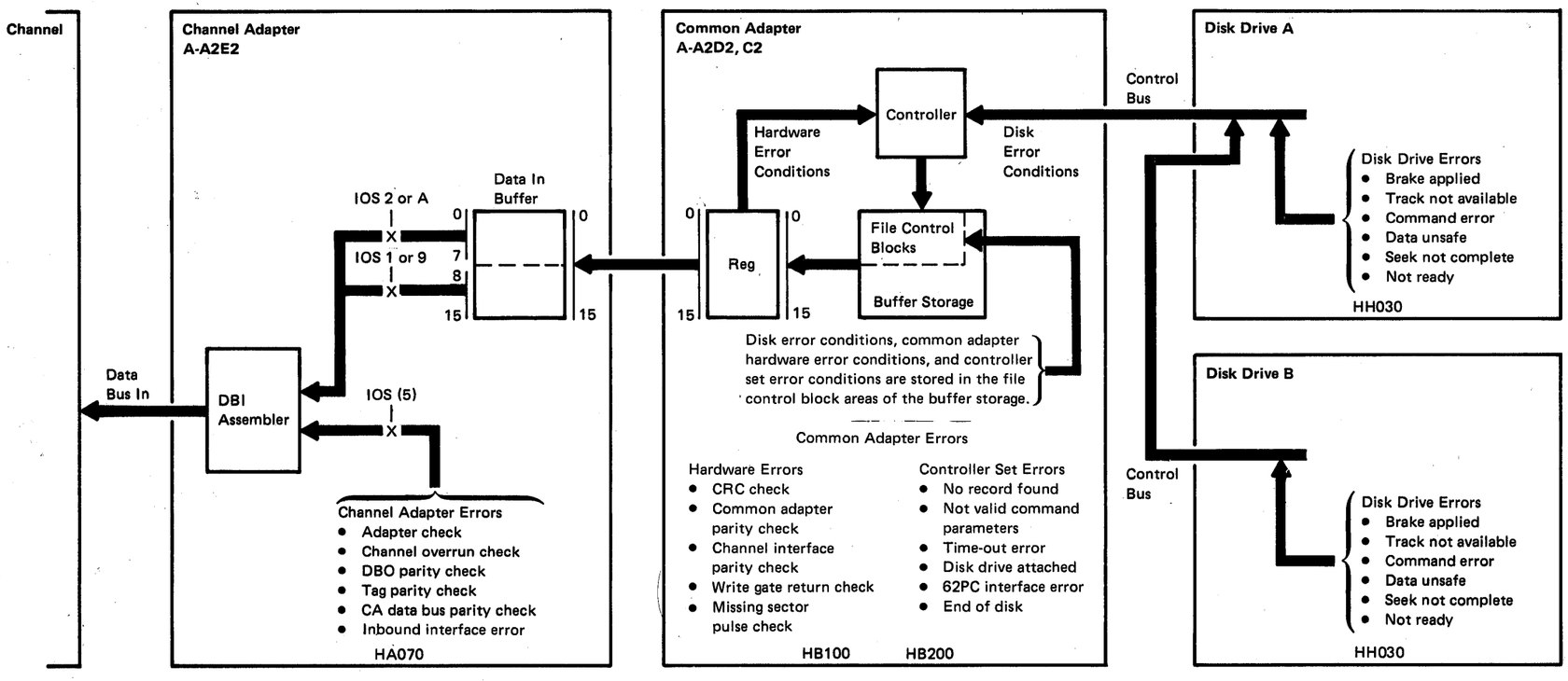
## ERROR CONDITIONS

The following sense bytes indicate errors:

- Sense byte 0 indicates errors the channel adapter senses.
- Sense byte 3 indicates errors the disk drive senses.
- Sense bytes 1, 4, and 5 indicate errors the common adapter senses.

Errors sensed by the disk drive are sent to the common adapter on the control bus during a disk sense (tag code 100). The common adapter controller moves the sense byte received from the disk into the FCB area of buffer storage. The controller also stores common adapter sensed errors, both hardware and microcode, in the FCB area of buffer storage. All error conditions in the FCB area are moved to the control processor using IOS commands to the channel adapter.

*Note:* The common adapter error checking circuits are internal to common adapter cards and cannot be probed.



## Channel Adapter Sensed Errors

*Adapter Check (Byte 0, Bit 0):* This check indicates that one of the following hardware error conditions was set by the channel adapter:

- Channel overrun check
- Parallel DBO parity check
- Tag parity check
- CA data bus parity check
- Inbound interface error

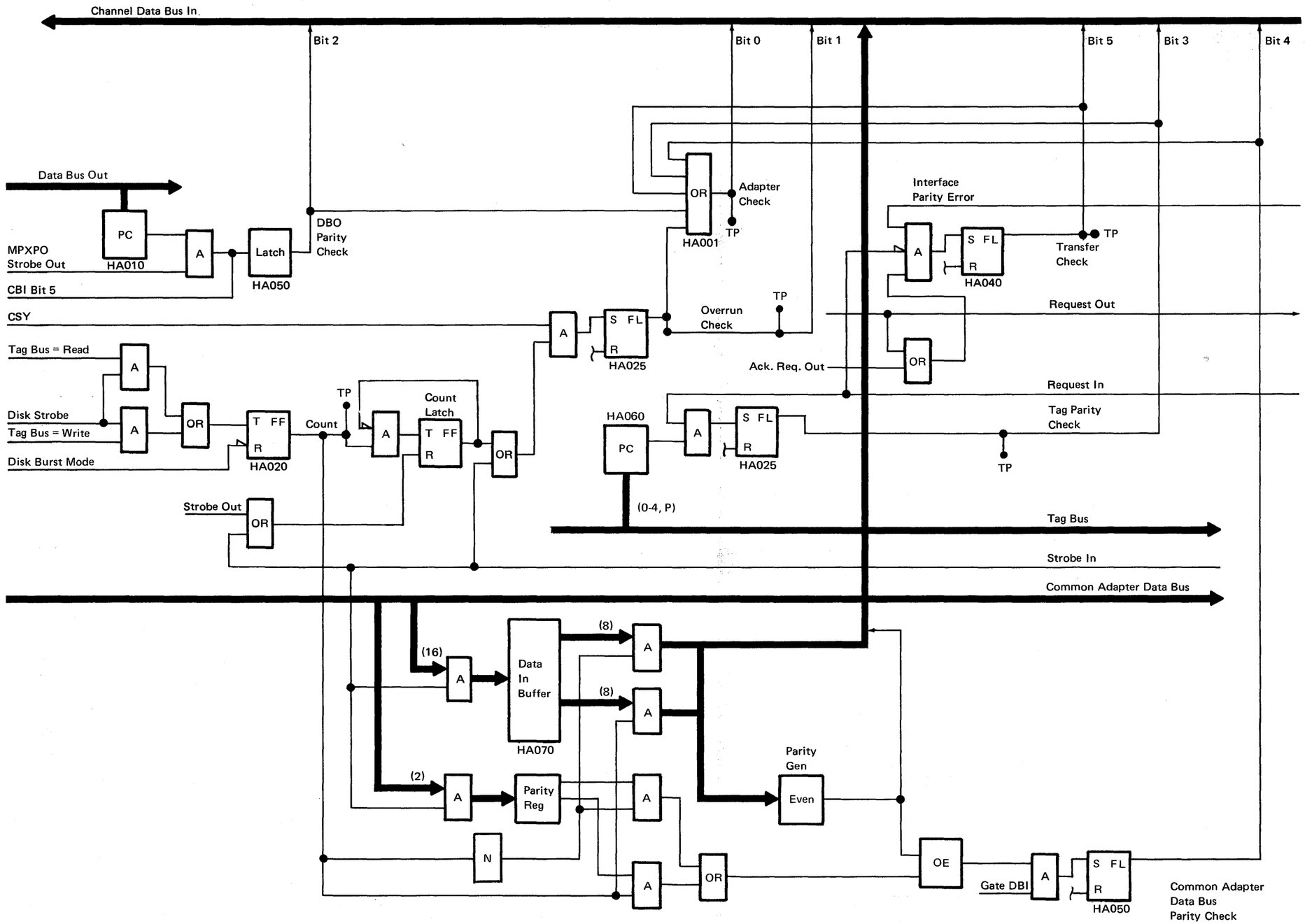
*Channel Overrun Check (Byte 0, Bit 1):* This check indicates that the data rate was not maintained during a cycle steal operation. The channel overrun check latch is set if the 'strobe in' pulse does not occur or is still on when a 'CSY' pulse occurs during a cycle steal to the control processor, or if the 'strobe out' pulse does not occur before a 'CSY' pulse during a cycle steal from the control processor.

*Parallel DBO Parity Check (Byte 0, Bit 2):* This check indicates that bad parity was sensed on the DBO during an I/O instruction. When bad DBO parity is sensed, CBI bit 5 is set when the 'MPXPO strobe out' line goes active and the error latch is set in the channel adapter.

*Tag Parity Check (Byte 0, Bit 3):* This check indicates that the channel adapter sensed bad parity on tag bus lines 0 through 3 and the parity line from the common adapter.

*CA Data Bus Parity Check (Byte 0, Bit 4):* This check indicates that the channel adapter sensed bad parity on the data bus from the common adapter.

*Inbound Interface Error (Byte 0, Bit 5):* This check indicates that bad parity was sensed by the common adapter on either the tag bus or the data bus between the channel adapter and the common adapter during an operation started by the control processor.



## Disk Drive Sensed Errors

*Brake Applied (Byte 3, Bit 1):* This check indicates that the brake coil is no longer activated and that AC power to the drive motor should be removed. The brake applied check is set if:

- The disk is stopped or not turning at the correct speed
- Servo signals are not being received from the disk enclosure
- The write clock controls cannot synchronize the write clock to the servo signals

*Track Unavailable (Byte 3, Bit 2):* This check indicates that an attempt was made to seek beyond cylinder 359. This bit is also set by bit 13 (end of disk) of the FCB error sense word.

*Command Error (Byte 3, Bit 3):* This check is set when the disk drive senses a parity error on the control bus or tag bus lines. This check causes the disk to set the 'interrupt' line to the common adapter, which causes the common adapter to sense the disk drive and store the sense information in the FCB. The sense resets the error, except for the tag parity check which is reset by the 'reset error' line.

*Data Unsafe (Byte 3, Bit 4):* This check indicates that a condition was sensed that can lead to possible lost data. The condition causes the disk to set the 'interrupt' line to the common adapter. Write current is inhibited and the data head lines are de-activated. The disk drive ignores all commands except sense commands. The following conditions set the data unsafe bit:

- Write or read selected and multiple chip selection error
- Write selected and no write data
- Write and read selected
- Write selected and not ready
- Write selected and a head short circuit to ground indicated
- Write selected and data servo unsafe
- Write selected and a head selected during an index or sector pulse
- Write selected and not on track
- Write not selected and write current

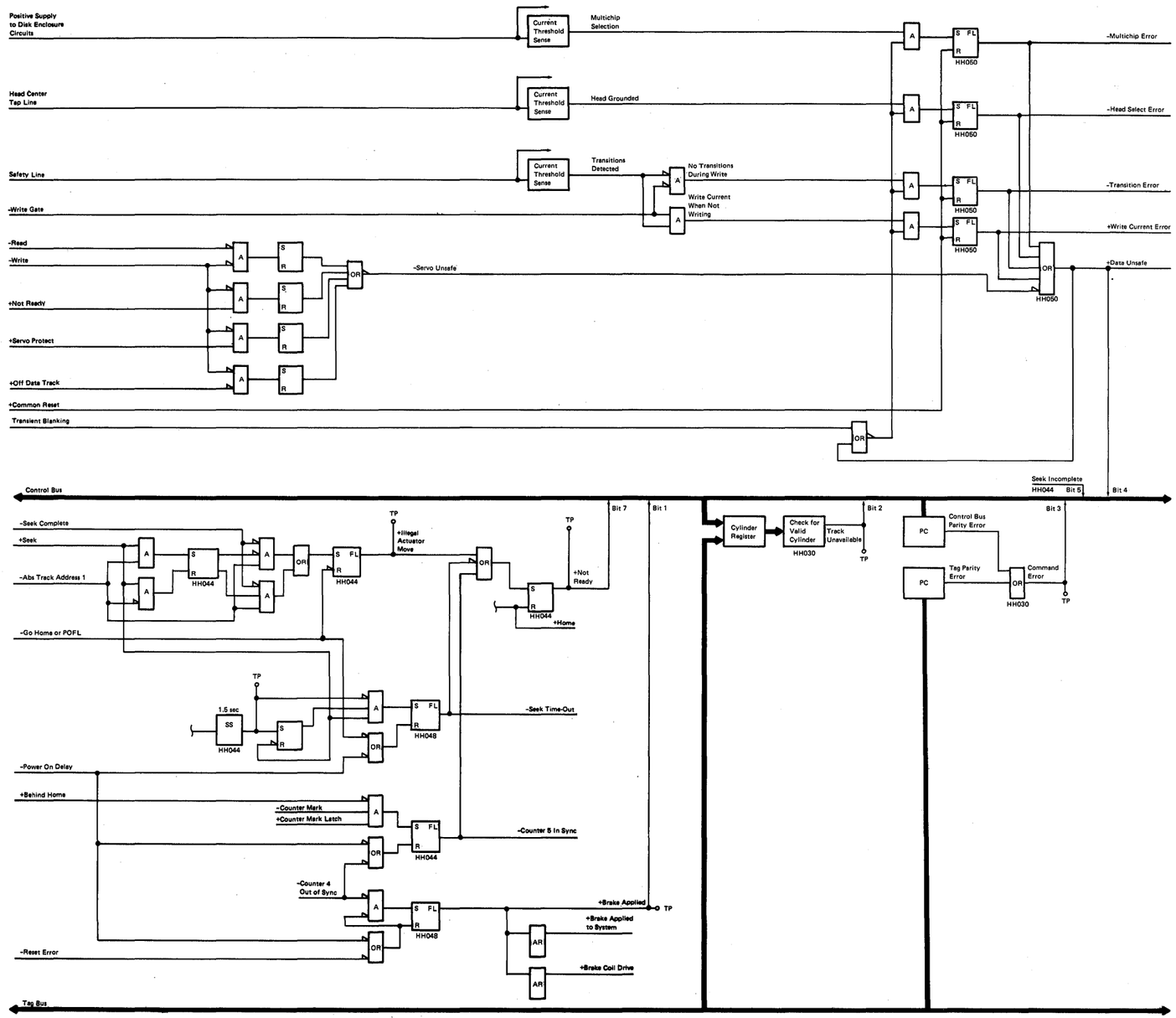
The 'reset error' line is used to reset this error.

*Seek Incomplete (Byte 3, Bit 5):* This check indicates that a seek operation was not completed or was completed too slowly. Normally, this condition is set when a seek is started and reset when the seek is completed.

*Disk Not Ready (Byte 3, Bit 7):* This check indicates one of the following:

- A seek time-out occurred, indicating a seek was not complete after 1.5 seconds.
- The 'illegal actuator move' line is active, which indicates that actuator movement occurred without an access command or that an attempt to write occurred during an access command.
- The conditions that set brake applied also set disk not ready.

A 'not ready' condition causes the disk drive to set the 'interrupt' line to the common adapter. The not ready error is reset by the 'reset error' line and a recalibrate operation is needed. The brake applied condition requires a power-on cycle.



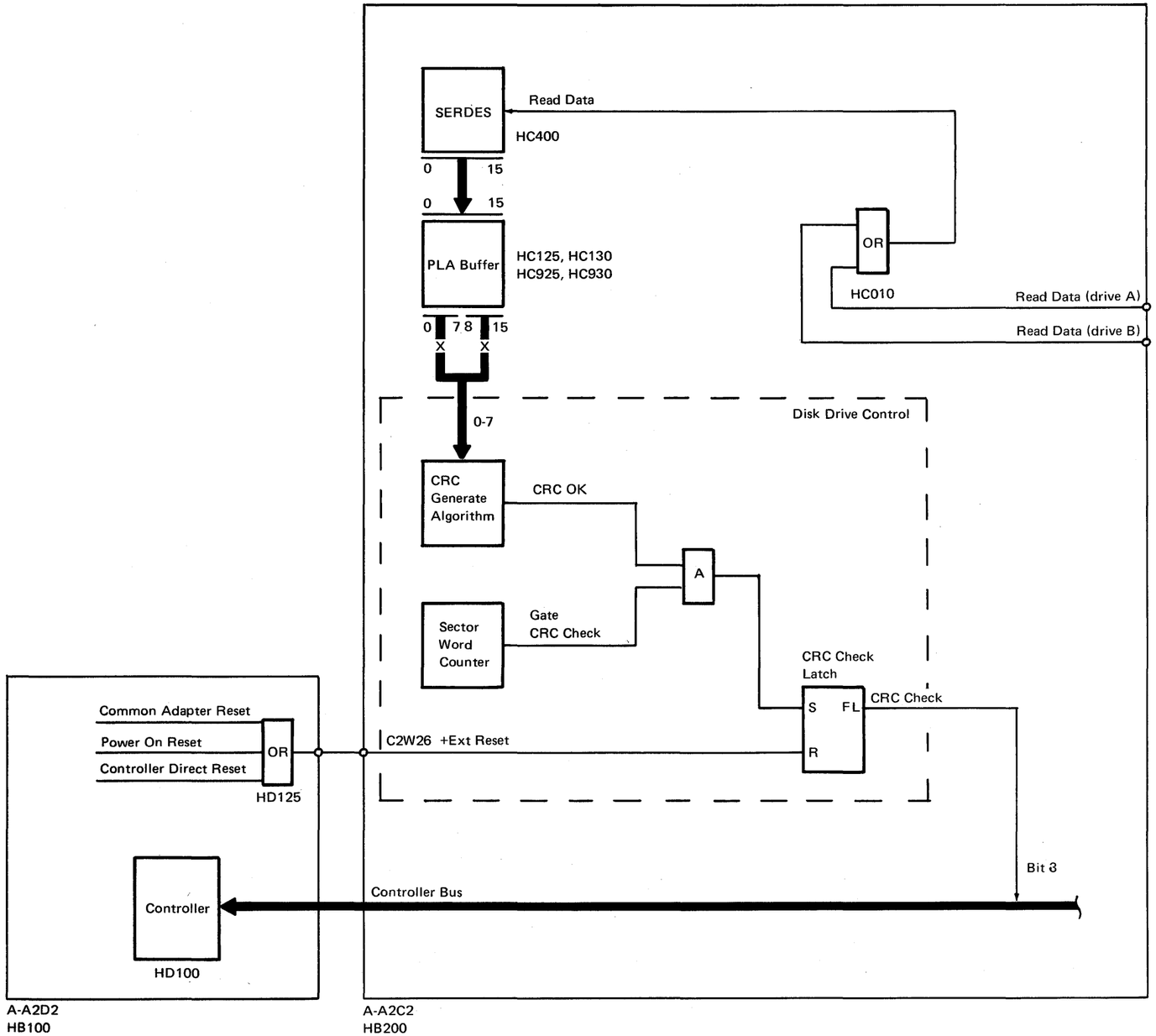
## Common Adapter Sensed Errors

*Any Error (Byte 1, Bit 5):* This check indicates that the common adapter sensed an error during an operation. Any of the following disk drive errors or common adapter errors set this check:

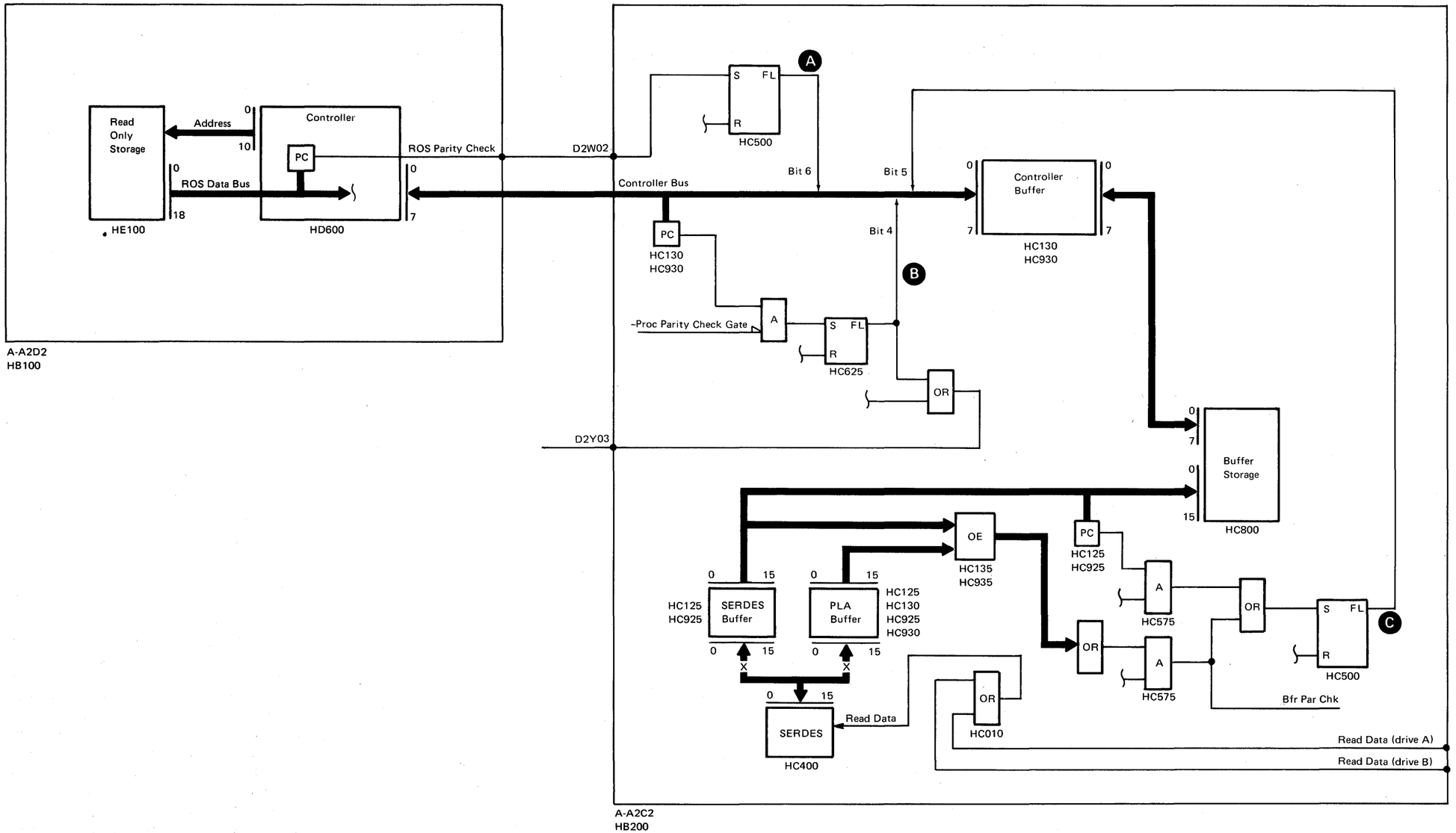
- Cyclic redundancy check (CRC)
- Common adapter parity check
- Channel interface parity check
- Write gate return check
- No record found
- Not valid command parameters
- Missing sector pulse
- Time-out error
- Disk drive not attached
- 62PC interface error
- Brake applied*
- Track unavailable*
- Command error*
- Data unsafe*
- Seek incomplete*
- Disk not ready*

*See *Disk Drive Sensed Errors* earlier in this section.

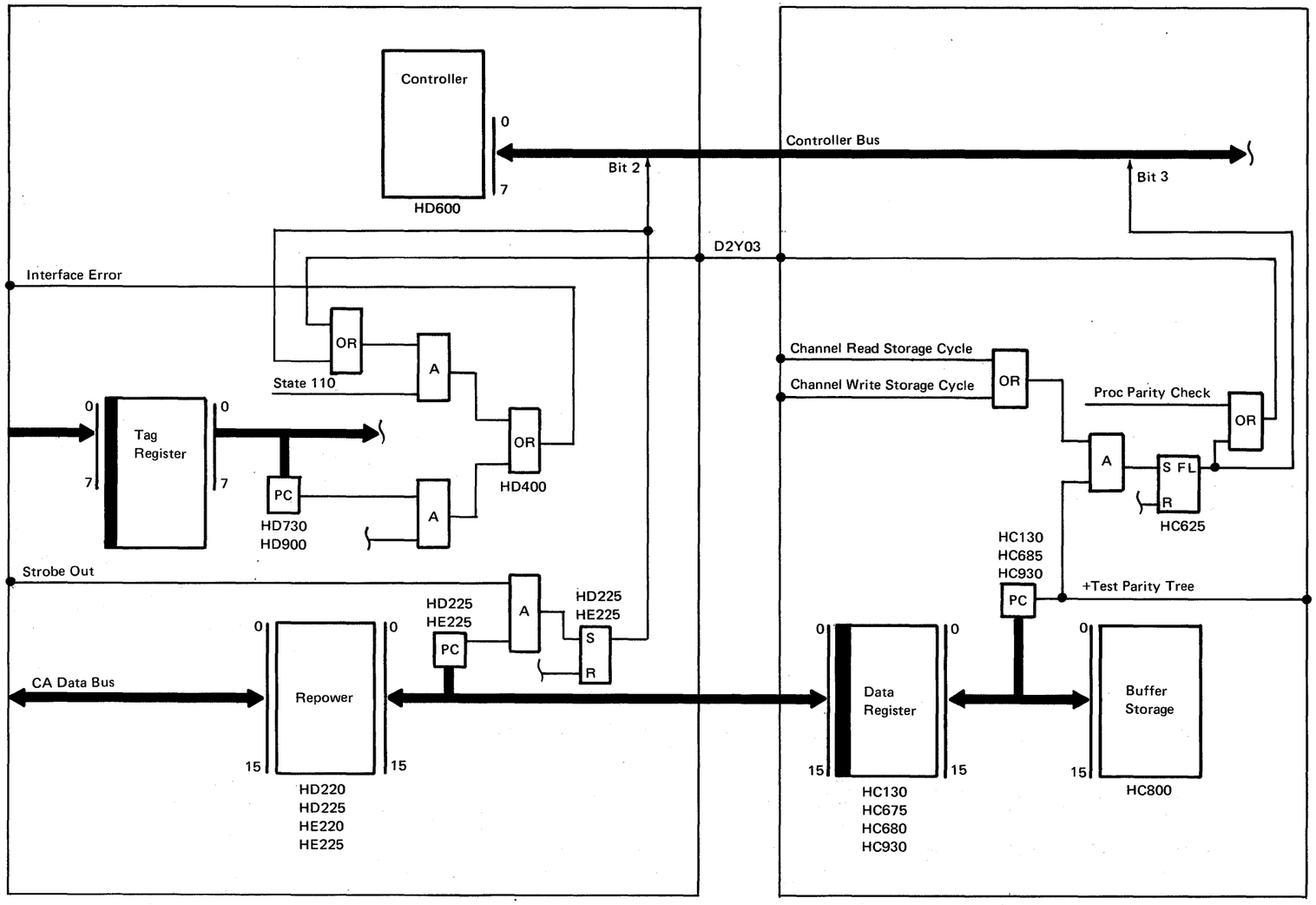
*Cyclic Redundancy Check (Byte 4, Bit 0):* This check indicates that after the correct sector was located, the data CRC character generated from the disk did not compare with the CRC character field. A CRC error condition can be indicated during a read or scan operation. If a CRC check occurs while the common adapter is searching for a specific ID, the CRC error is not recorded. If a CRC check occurs on the ID the common adapter is searching for, a no record found error is recorded. The controller senses the check condition using the controller bus, and sets the CRC check bit in the file control block. The following figure summarizes the logic for a cyclic redundancy check.



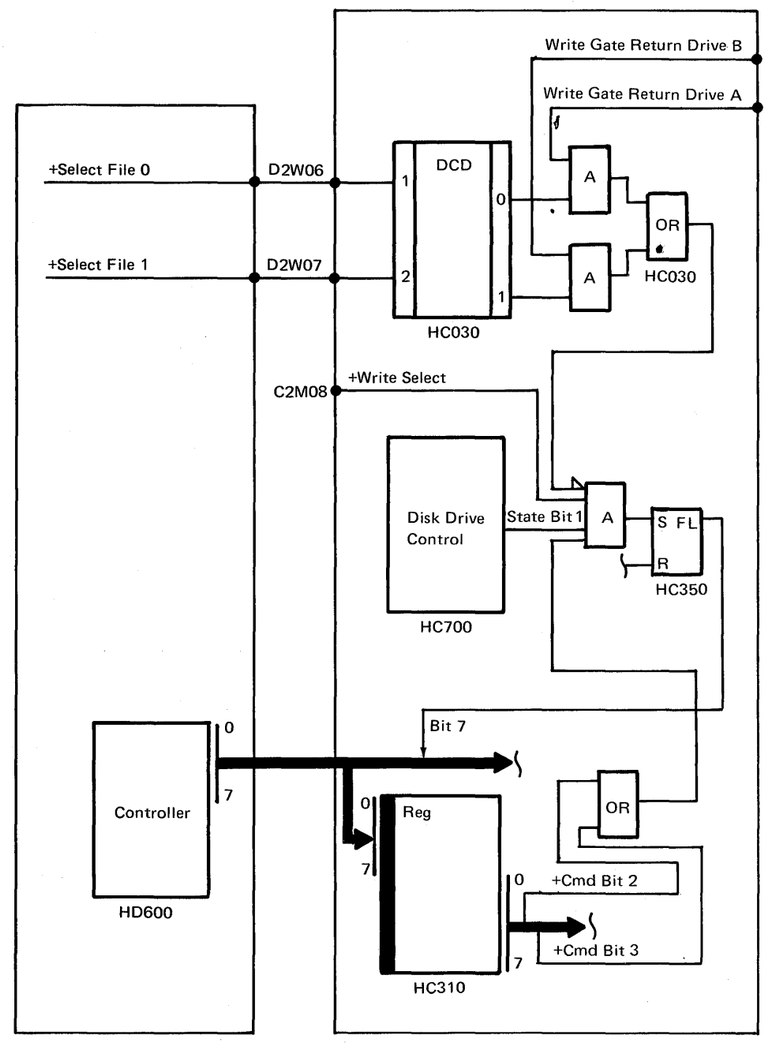
*Common Adapter Parity Check (Byte 4, Bit 1):* This check indicates that the common adapter has sensed bad parity in its internal circuits. The check can be set by bad ROS parity (A), by bad parity on the controller bus (B), or by bad parity or data compare failure on the SERDES to buffer storage bus (C). The controller senses these check conditions using the controller bus and sets the common adapter parity check bit in the file control block. The following figure summarizes the logic for a common adapter parity check.



*Channel Interface Parity Check (Byte 4, Bit 2):* This check indicates that the common adapter has sensed bad parity on either the data bus or the tag bus between the channel adapter and the common adapter. The controller sets the channel interface parity check bit in the file control block when it detects the error. The following figure summarizes the logic for a channel interface parity check.




*Write Gate Return Check (Byte 4, Bit 3):* This check indicates that the common adapter attempted to write to the disk drive but no write current was sensed. The check is set if write gate return is not received after the common adapter has set write to a drive. The controller senses the error and sets the write gate return check bit in the file control block. The following figure summarizes the logic for a write gate return check.

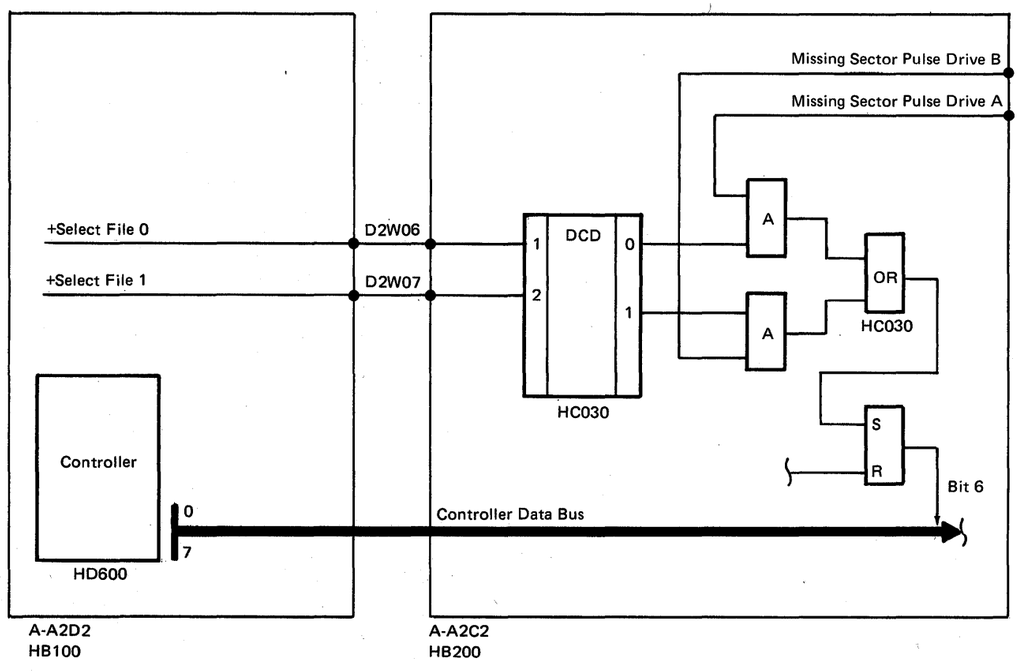


*No Record Found (Byte 4, Bit 4):* This check indicates that the sector being addressed was not found after two index pulses (one revolution of the disk). On an operation using more than one sector, if an ID hit has occurred on the first sector, this error is set if any following sector is not found two sector pulses after the last ID hit. This error is set by the common adapter controller.

*Not Valid Command Parameters (Byte 4, Bit 5):* This check indicates that the command that is loaded in the FCB has parameters that are not valid. The common adapter controller checks for the following FCB errors:

- FCB word 0 (bits 0 through 7), word 2 (bits 0 through 6), and word 3 (bits 0 through 2) must be zero.
- The command byte must be valid (FCB word 0, low byte).
- A sector count (FCB word 1, high byte) larger than 32 for a read ID or write ID operation; larger than 0 for a read diagnostic operation.
- A record number (FCB word 3, low byte) larger than 63 for a read, write, or scan command.
- A sector number (FCB word 3, low byte) larger than 32 for a read ID, write ID, or read diagnostic operation.

*Missing Sector Pulse (Byte 4, Bit 6):* This check indicates that the 'missing sector pulse' line from the disk drive was active during a read ID, write ID, or read diagnostic command. This line is ignored during a read data, write data, or scan command unless it occurs at the selected sector, which causes a no record found error. The controller uses the controller bus to sense the error condition and sets the missing sector pulse bit in the common adapter file control block. The following figure summarizes the logic for a missing sector pulse error.




*Time-out Error (Byte 4, Bit 7):* This check is set by the following common adapter hardware conditions:

- Seek complete interrupt not received: The common adapter has not received a seek complete interrupt within 200 milliseconds of the start of a seek command or within 5.1 seconds of the start of a recalibrate command.
- Read/write hardware time-out: The common adapter controller determined that the read/write counters have not advanced during a read or write operation.
- The common adapter controller executed commands are not valid: An attempt was made to execute not used locations of the common adapter controller ROS.

*Disk Drive Not Attached (Byte 5, Bit 0):* This check indicates that the command received by the common adapter specifies a 62PC disk drive that is not attached to the common adapter. This check is determined by the configuration jumpers on the common adapter erface card.

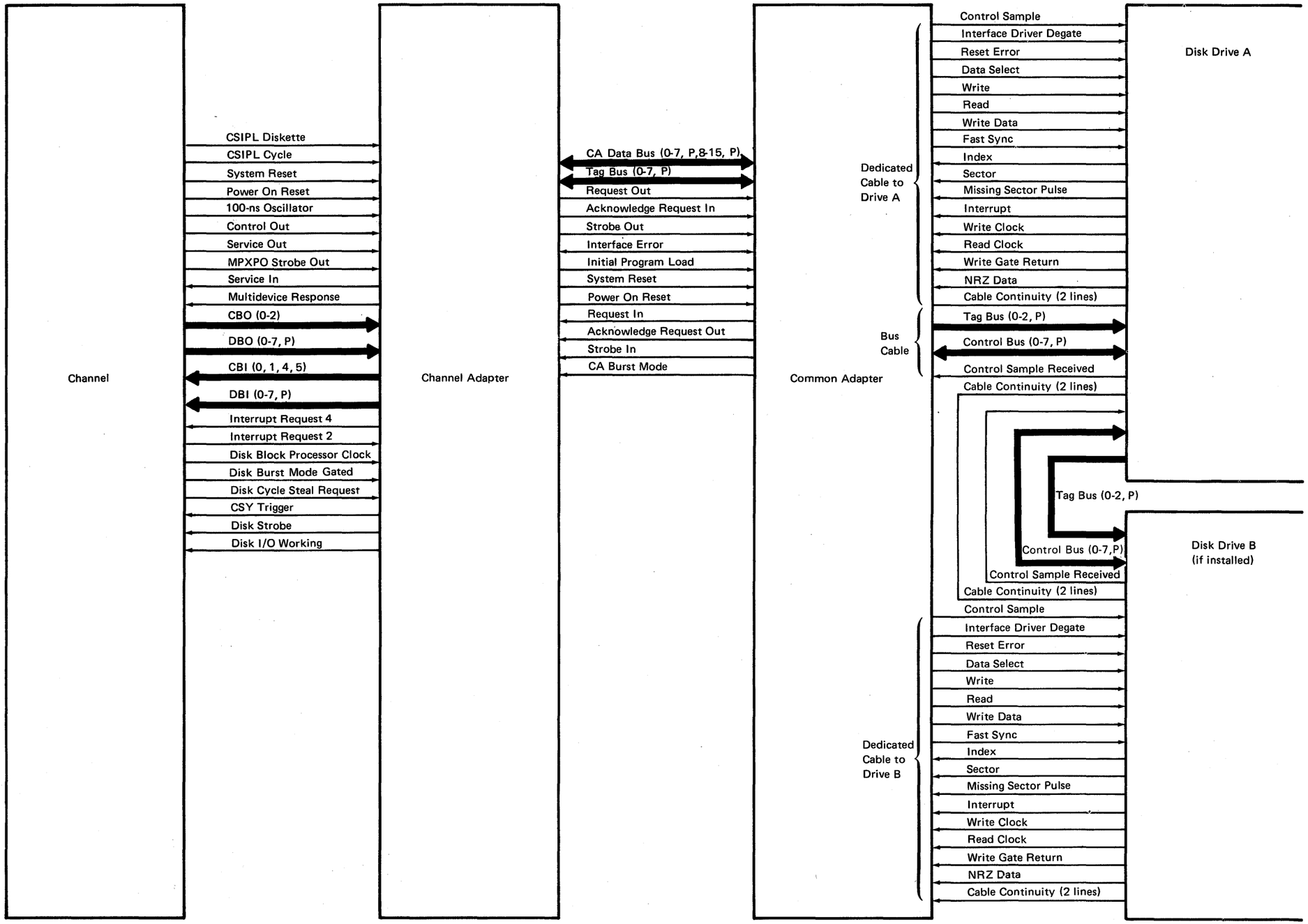
*62PC Interface Error (Byte 5, Bit 7):* This check indicates that an error occurred in the interface to the disk. The following conditions set this error:

- Cable continuity lines open.
- The 'control sample received' line not set in response to the 'control sample' line.
- A parity check on the control bus during an incoming data movement.
- The interrupt line from the disk not reset or did not remain reset after a sense.

This page intentionally left blank.

## INPUT/OUTPUT LINES

The following figure shows the lines between the channel and the channel adapter, the lines between the channel adapter and the common adapter, and the lines between the common adapter and the 62PC disk drive(s). Descriptions of these lines follow the figure.



## Lines Between the Channel and the Channel Adapter

There are 43 lines between the channel and the channel adapter. These lines are used for control and to move information.

*CSIPL Diskette:* This line is used to select the diskette or disk for CSIPL. The disk is selected when the line is at an up level.

*CSIPL Cycle:* This line is used to request a CSIPL.

*System Reset:* This line is used to reset all latches in the channel adapter; it is also sent to the common adapter.

*Power On Reset:* This line is used to reset all latches and to clear buffer storage in the common adapter after a power on operation.

*100-ns Oscillator:* This line is used to generate the 'strobe out' pulse.

*Control Out:* This line is set by the channel to start an I/O command, and to indicate that the CBO and DBO are valid.

*Service Out:* This line is set by the channel in response to a 'service in' pulse, and to indicate that the DBO is valid for the second part of the command.

*MPXPO Strobe Out:* This line is used as a timing pulse for I/O commands.

*Service In:* This line is set by the channel adapter in response to a 'control out' pulse after receiving the CBO and DBO, if the disk was the I/O device that was addressed. This line is reset in response to a 'service out' pulse, after the DBO is received.

*Multidevice Response:* This line is set by the channel adapter in response to a 'control out' pulse.

*CBO (0-2):* These lines specify the I/O command. See *Commands* earlier in this section.

*DBO (0-7, P):* These lines are used for control information during a 'control out' pulse, specifying the device address and command modifier. They are used during a 'service out' pulse to move control information to the I/O device. During cycle steal they are used to move data to the I/O device.

*CBI (0, 1, 4, 5):* These lines indicate the following when active:

- Bit 0: This line is used during burst mode cycle steal to increment the main storage or control storage address register.
- Bit 1: This line is used during burst mode cycle steal to indicate the direction that data is moved.
- Bit 4: This line is used during burst mode cycle steal to select control storage. It is used to indicate a jump condition met during a jump on I/O command to the disk, and to inhibit parity checking on the DBI during an SILSB command or a disk status sense command. See *Commands* earlier in this section.
- Bit 5: This line is used to indicate bad parity on the DBO.

For more information, see *Command Bus In* earlier in this section.

*DBI (0-7, P):* These lines are used to move control and status information to the control processor during I/O commands. They are used to move data into storage during cycle steal.

*Interrupt Request 4:* This line is used by the channel adapter to indicate an interrupt request.

*Interrupt Request 2:* This line causes the channel adapter to delay the start of a cycle steal until IL2 is reset.

*Disk Block Processor Clock:* This line is used by the channel adapter to request burst mode for data movement.

*Disk Burst Mode Gated:* This line is set by the channel in response to block processor clock.

*Disk Cycle Steal Request:* This line is used by the channel adapter to request the start of a burst mode data movement.

*CSY (Channel Storage Cycle) Trigger:* This line is used by the channel adapter during burst mode cycle steal to set an overrun error, if the data movement across the channel adapter interface is too slow.

*Disk Strobe:* This line is used by the channel adapter for internal control and to gate and latch internal data during burst mode data movement.

*Disk I/O Working:* This line, set by programming, indicates the disk is operating.

## Lines Between the Channel Adapter and the Common Adapter

There are 38 lines between the channel adapter and the common adapter. These lines permit communications between the channel adapter and the common adapter.

### CA Data Bus (18 Lines, Bidirectional)

These lines represent two 8-bit bytes, each with a parity bit (odd), that are used to move data and control information.

### Tag Bus (9 Lines)

These lines represent one 8-bit byte, with a parity bit (odd), that is used to describe the information on the CA data bus. This bus is gated by the channel adapter when the 'request out' line is active. Bits 0 through 3 and the parity bit are bidirectional and are gated by the common adapter when the 'request in' line is active.

*Request Out:* This line is set by the channel adapter to signal the common adapter that the channel adapter is sending a common adapter command on the tag bus and data on the CA data bus. The common adapter determines the source, type, and direction of the data to be moved by use of the tag bus. A 'request out' line has priority over a 'request in' line.

*Acknowledge Request In:* This line is a response from the channel adapter to a 'request in' line when the data movement is from common adapter to channel adapter. It indicates the data movement requested by the common adapter can be completed.

*Strobe Out:* This line is the signal from the channel adapter that is used as a load pulse for moving data from the channel adapter to the common adapter.

*Interface Error:* This line is a bidirectional line that signals not valid parity has occurred on the CA data bus or tag bus lines. This error is signaled by the common adapter during a 'request out' pulse and by the channel adapter during a 'request in' pulse.

*Initial Program Load:* This line is set by the channel adapter when a disk CSIPL is started.

*System Reset:* This line resets the common adapter to a not busy, cleared, no-error condition. The reset goes to each disk drive, resetting any errors that had been set. The reset ends any read or write operation now executing.

*Power On Reset:* This line initializes the common adapter to a cleared condition.

*Request In:* This line is set by the common adapter to signal the channel adapter of a need to move data.

*Acknowledge Request Out:* This line is used by the common adapter to respond to a 'request out' line.

*Strobe In:* This line is the signal from the common adapter that is used as a load pulse for moving data from the common adapter to the channel adapter.

*CA Burst Mode:* This line indicates that a block of data is ready to be moved into main or control storage, through the channel adapter.

## Lines Between the Common Adapter and the 62PC Disk Drive

The lines between the common adapter and the 62PC disk drive are in two cables: the dedicated cable and the bus cable. Each disk drive has its own dedicated cable and the bus cable goes to the first drive and then from the first drive to the second drive, if a second drive is present. The bus cable must be terminated in the last drive by a terminator card.

### Dedicated Cable

The dedicated cable contains the lines that control disk drive operations and signal the common adapter of the needs of the disk drive.

*Control Sample:* This line is used with the 62PC tag lines to start a control bus load or sense cycle. See *Tag Bus and Control Bus* later in this section for the meanings of the lines during the 'control sample' pulse.

*Interface Driver Degate:* This line disables the 62PC drivers for the 'control bus' and 'control sample received' lines, and holds them at an up level.

*Reset Error:* This line is used to reset the data unsafe or command error sense bits.

*Data Select:* This line enables the 62PC disk drive to use the 'read' and 'write' lines on the dedicated cable. When this line is not active, the 'read clock', 'write clock', and 'NRZ data' lines are held down.

*Write:* This line turns on write current source to permit write data to be written by the selected head. A data unsafe error turns off the write current.

*Read:* This line is used with the 'data select' line to permit data to be decoded from the disk.

*Write Data:* This line is at a down level for a 1-bit and at an up level for a 0-bit. The line changes level only with the write clock.

*Fast Sync:* This line is used to synchronize the data separator before reading the sector ID or before reading data.

*Index:* This line indicates the start of the first physical sector of each cylinder.

*Sector:* This line indicates the start of each sector except the first, which is started by the 'index' pulse.

*Missing Sector Pulse:* This line is active if one or more index or sector pulses is missing.

*Interrupt:* This line is active when the 62PC needs system service.

*Write Clock:* This line is present when the 'data select' line is active and the 62PC disk is ready. The 'write clock' pulses are synchronized to the servo signals received from the servo head.

*Read Clock:* This line is generated from data signals that are received from the selected data head. This signal is present only when the 'data select' line is active, but the read clock timing is valid only when the 'read' line is active.

*Write Gate Return:* This line is used to verify that the write gate was set in the disk drive.

*NRZ Data:* This line is the output of the data separator: 1 is an up level and 0 is a down level. Read data is valid after the end of fast sync.

*Cable Continuity:* These two lines in the dedicated cable are used to check that the cable between the common adapter and the 62PC disk drive is connected correctly.

**Bus Cable**

The bus cable contains the tag bus, the control bus, the 'control sample received' line, and two cable continuity lines.

*Tag Bus and Control Bus*

The tag bus contains three tag lines and a parity line (odd parity). Tag bus bit 0 is the most significant bit and tag bus bit 2 is the least significant bit. The tag bus describes the information on the control bus.

The control bus contains eight control lines and a parity line (odd parity). Control bus bit 0 is the most significant bit and control bus bit 7 is the least significant bit. The control bus moves information between the common adapter and the 62PC disk drive(s). The tag bus and control bus bits are described in the following figure.

*Control Sample Received*

This line becomes active when the 'control sample' line is active and the 62PC disk drive reads the tag bus lines.

*Cable Continuity*

These two lines in the bus cable are used to check that the cable between the common adapter and the 62PC disk drive is connected correctly.

| Tag Bus Bits 0 1 2 | Control Bus Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 0 0 Not Used | | | | | | | | |
| 0 0 1 Seek Control | Recalibrate | Reserved | Head select 8 | Head select 4 | Head select 2 | Head select 1 | Not used | Cylinder address 256 |
| 0 1 0 Required Address | Cylinder address 128 | Cylinder address 64 | Cylinder address 32 | Cylinder address 16 | Cylinder address 8 | Cylinder address 4 | Cylinder address 2 | Cylinder address 1 |
| 0 1 1 Diagnostic Wrap Byte | Cylinder address 128 | Cylinder address 64 | Cylinder address 32 | Cylinder address 16 | Cylinder address 8 | Cylinder address 4 | Cylinder address 2 | Cylinder address 1 |
| 1 0 0 Sense | Reserved (always 1) | Brake applied | Track not available | Command error | Data unsafe | Seek incomplete | Home | Not ready |
| 1 0 1 Diagnostic Sense 1 | On track | Linear region normal and even | Index and sector pulses missing | Out direction | Not out drive | Not in drive | Tag parity error | Velocity profile error |
| 1 1 0 Diagnostic Sense 2 | Behind home | Missing clocks divided by 2 | Missing clocks error latch | Coil current low | Missing servo signal | Off data track | Not missing position error signal | Counter 5 in sync |
| 1 1 1 Diagnostic Sense 3 | Not shift | Not (off track and write) | Inside AGC window | Not AGC freeze | Demodulator pulsing | Not (read and write) | Not (servo protect and write) | Illegal actuator move |

This page intentionally left blank.