CRD-5400

SCSI RAID Controller
OEM Manual

CMD TECHNOLOGY INC.

MAN-105400-000
Rev. 1.1
April 1, 1996

CRD-5400

SCSI RAID Controller
OEM Manual

CMD Technology, Inc.
1 Vanderbilt
Irvine, California 92718
(714) 454-0800

Trademarks and Copyright

CMD, CMD Technology, and CRD-5400 are trademarks of CMD Technology, Inc. All other product and company names are trademarks or registered trademarks of other manufacturers.

CMD reserves the right to make changes to this manual and the equipment described in this manual without notice. CMD has made all reasonable efforts to ensure that the information in this manual is accurate and complete. CMD will not be liable, however, for any technical or editorial errors or omissions made in this manual, or for incidental, special, or consequential damage of whatsoever nature, resulting from the furnishing of this manual, or operation and performance of equipment in connection with this manual.

FCC Notice

This equipment has been tested and found to comply with the limits of a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Any changes or modifications not expressly approved by the manufacturer could void the user's authority to operate the equipment.

Warranty

CMD Technology warrants this product to be free of defects in materials and/or workmanship for a period of 3 years from the date of purchase. If the product proves to be defective within the warranty period, CMD Technology will either repair or replace it. This warranty covers defects incurred in normal use only. Defects, malfunctions, or failures resulting from accidents, misuse, or mishandling are not covered. In the event that this product must be repaired or replaced, please contact CMD for an RMA number.

Important
This warranty does not cover controller failures brought about by excessive heat build up due to inadequate air flow over the controller. Failure to use the fan and bracket assembly that comes with the CRD-5400 or a similar measure to dissipate the heat generated by the controller will void this warranty.

Table of Contents

This page left blank intentionally.

1 System Integration Issues

1.1 Cabinet Integration

The CRD-5400 is designed to fit in any standard enclosure built to accommodate 5.25-inch form factor devices. Figure 1-1 shows the relevant chassis dimensions for mounting in an enclosure.

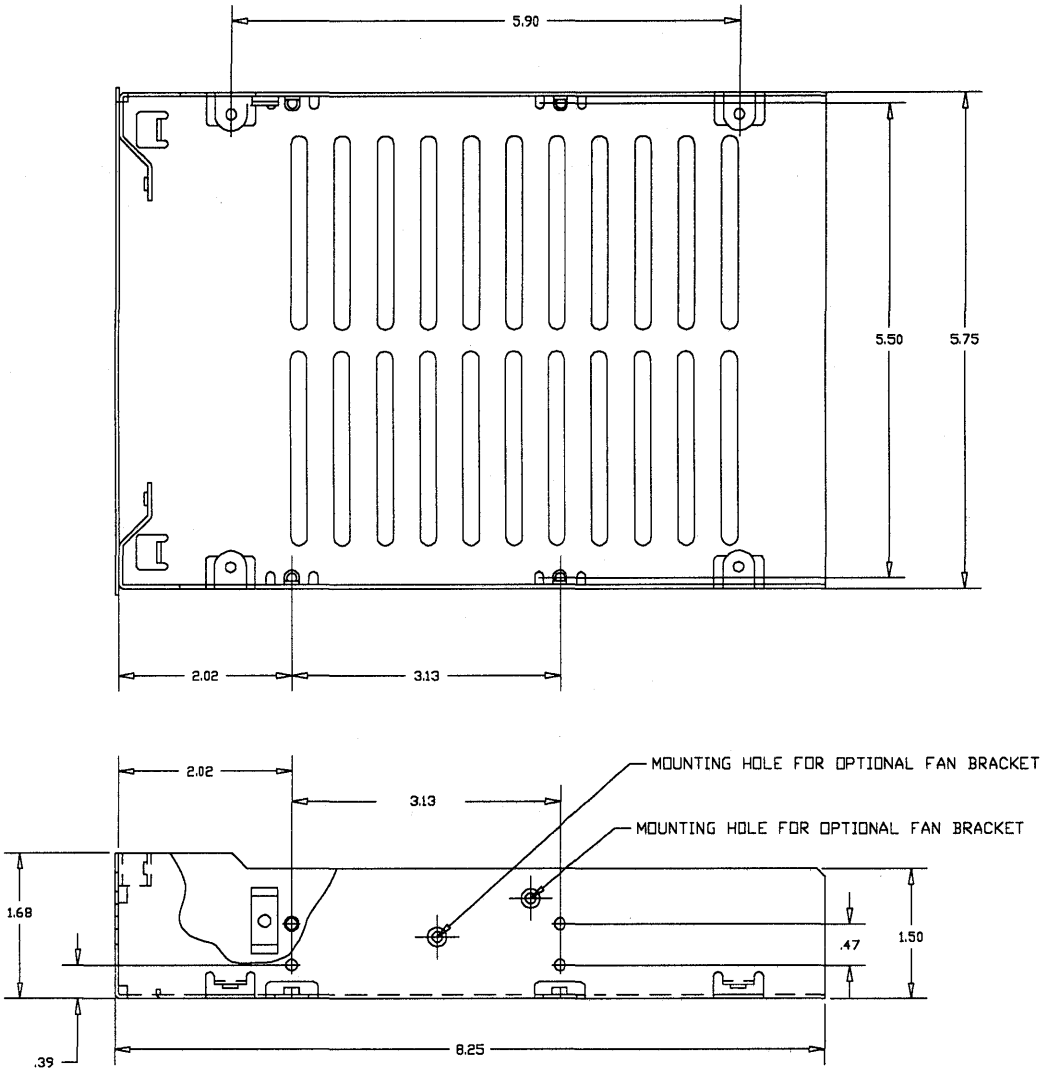


Figure 1-1: Mounting diagram

1.1.1 Mounting Screws

To secure the CRD-5400 within an enclosure, drive screws through the enclosure's rails and into the four holes on the bottom or the four holes drilled on the side of the CRD-5400. Screws driven into the bottom should be no longer than 5/16 inch. Screws driven into the sides should be no longer than 1/4 inch. Longer screws can damage the circuitry inside the controller.

1.1.2 Ventilation

Be sure to leave enough space above the CRD-5400 box to ensure that the controller's fan can provide adequate air flow to the components inside the box.

CMD recommends a minimum vertical clearance of 1.0 inch above the box. No ventilation space is required below the controller.

Make sure the enclosure housing the CRD-5400 has its own ventilation system, preferably a fan and vent system that pulls fresh air through the front and exhausts hot air through the back. The CRD-5400 comes with a fan that attaches to the chassis on a bracket. This fan should be used to supplement the enclosure's ventilation system. It is not designed to provide adequate air flow by itself.

Important
Failure to use the fan and bracket assembly that comes with the CRD-5400 or a similar measure to dissipate the heat generated by the controller will void the warranty.

A two-level temperature sensor will monitor any heat build-up inside the CRD-5400. When the temperature reaches the first level, the alarm will sound every two seconds. In such an event, you should take immediate steps to cool the unit or shut the system down in an orderly fashion. If the temperature climbs to the second level, the alarm will sound twice a second. The CRD-5400 will complete any pending I/O activity and then accept no more instructions from the host.

1.1.3 Silencing the Alarm

The CRD-5400's alarm will sound when the temperature at the controller approaches the operational limit. It will also sound when the power supply voltage drops below 4.80 volts or exceeds 5.25 volts. You may silence the alarm by pressing Ctrl-x on the monitor keyboard. Or by pressing both the up and down arrow buttons on the front panel at the same time.

1.2 Board Connectors

Fig. 1-2 identifies the connectors on the rear of the controller.

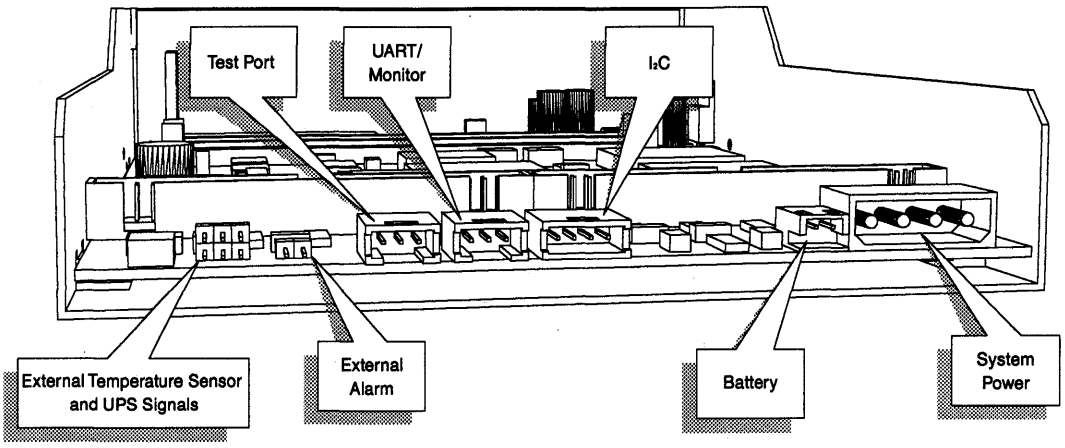
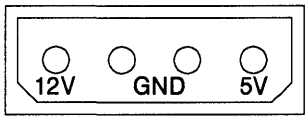


Figure 1-2: Back panel connectors

System Power: Provides 12±10% volts DC (3 amps maximum) to power the battery charger and fan, also provides 4.80 to 5.25 volts DC (4.5 amps maximum) to power the controller's systems and the disk channels' terminator resistors. The controller's systems draw a steady 2.5 amps. The terminator resistors can draw anywhere from zero to 2.0 amps from one instant to the next. The amount of current drawn by the terminator resistors depends on conditions beyond the CRD-5400's control, such as the RAID level of the RAID sets attached to the controller and the character of the data being written to or read from the RAID sets.

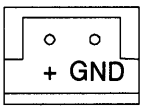
If the voltage on the 5-volt line dips to 4.80 volts or climbs to 5.25 volts, the controller will sound its alarm and record the event in the event log. If the power continues to fall to 4.75 volts or climb to 5.30 volts, the controller will shut down.

The pinout for the system power connector is show below.



Battery: Connects to a 6.6 volt DC, lead-acid gell pack backup battery with a capacity in the range of 4 to 34 amp hours. The CRD-5400 will trickle-charge the battery during normal operation. The recharge rate is set by the peak charge jumper shown in Fig. 1-3. Insert the jumper shunt for a 100 milliamp peak charge current. Remove the jumper shunt for a 10 milliamp peak charge current.

The following diagram shows the pinout of the connector.



Warning
The CRD-5400 will not operate unless it is connected to a battery backup or uninterruptible power supply (UPS). CMD Technology recommends that you make sure that your battery is fully charged before connecting it to the controller. Since the CRD-5400 trickle charges the battery, it can take several hours to fully charge a depleted battery. If power is interrupted to the controller, data is exposed and vulnerable without an adequate battery backup.

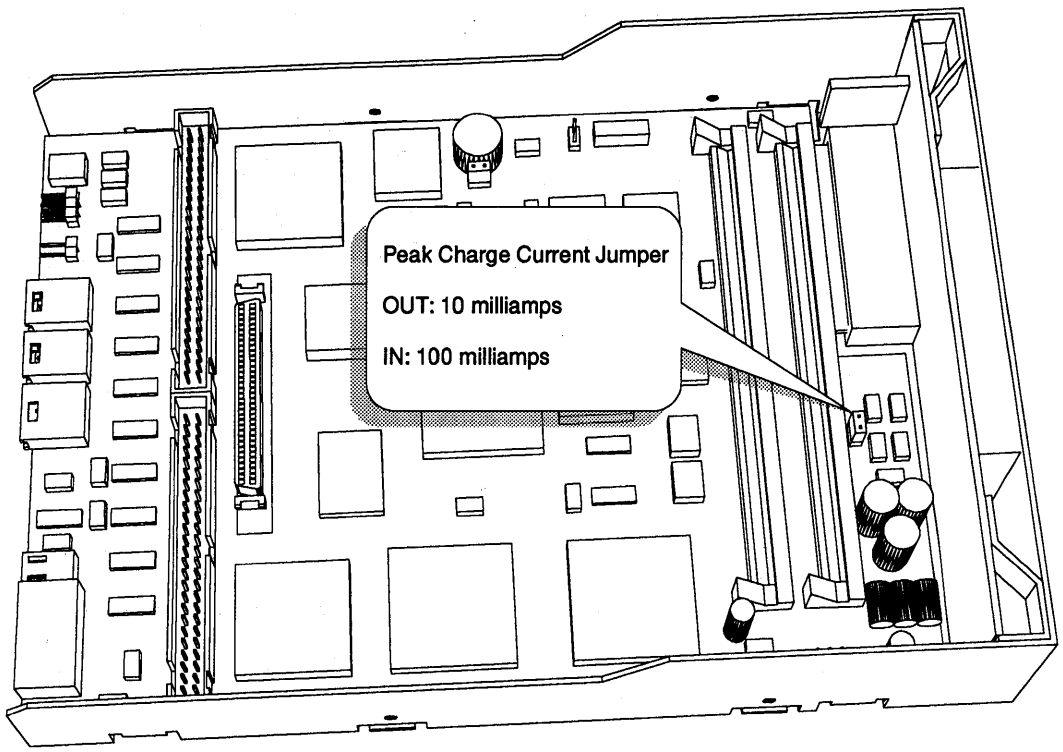
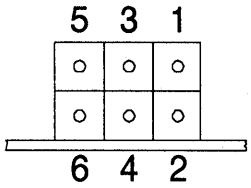


Figure 1-3: Peak charge current jumper

External Temperature Sensor and UPS Signal Connector: This connector does double duty. The odd pins may be used to attach an external temperature sensor to supplement the CRD-5400's internal temperature sensor. The even pins make it possible to bring critical status signals from an uninterruptible power source (UPS) unit into the controller. The following diagram and table show the pinout of the connector.



The top row of pins are for connecting external temperature sensor.

Pin	Description
1	Temperature Sensor: Power
3	Temperature Sensor: Signal Input
5	Temperature Sensor: Ground

Figure 1-4 shows how to connect an external temperature sensor to the CRD-5400. The external temperature sensor must be from National Semiconductor (Part Number LM34DZ).

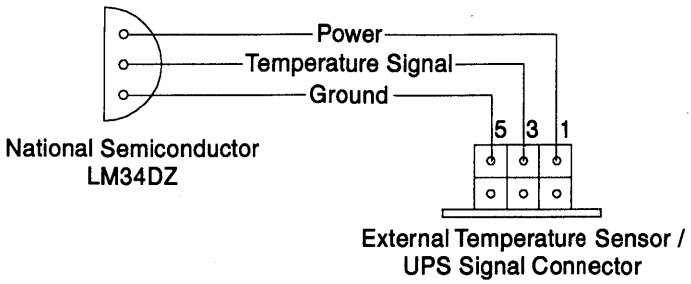
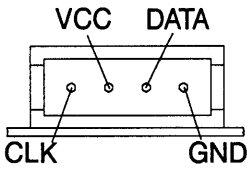


Figure 1-4: External temperature sensor wiring diagram

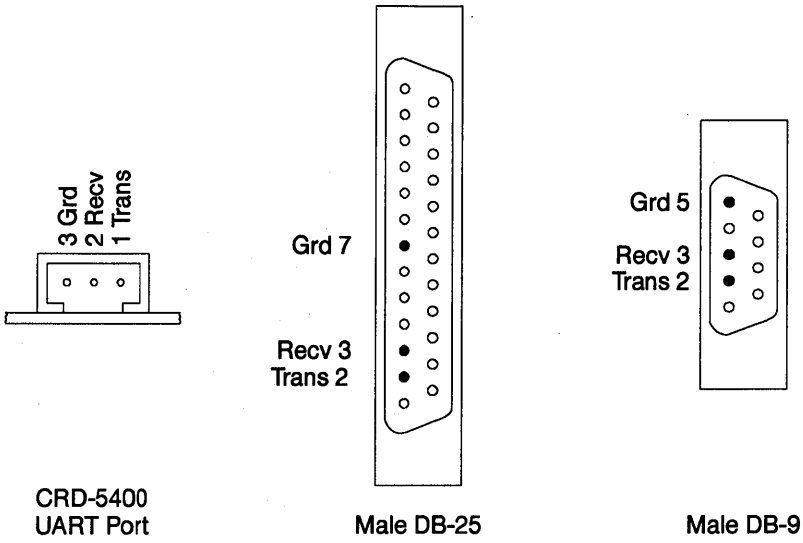
The bottom row of pins are for connecting a UPS. The current CRD-5400 firmware responds to both the AC power failure and UPS "two minute" warning signals by flushing its cache and refusing to accept additional commands from the host.

Pin	Description
2	UPS: low battery "two minute" warning (5 milliamps max., low true)
4	UPS: AC power fail (5 milliamps max., low true)
6	UPS: Ground

Inter-Integrated Circuit (I^2C) Connector: This is used for the environmental control system (ECS), which will be implemented in a future firmware release. The pinout of the connector is shown in the following diagram.



UART Connector: This is used to connect an external monitor to the controller. The CRD-5400's monitor utility offers a means to configure the controller, create and manage RAID sets, initiate rebuild operations, and view error logs, among other functions. The following diagram shows the pinout of the UART connector and how the signals should be carried to DB-25 and DB-9 connectors.

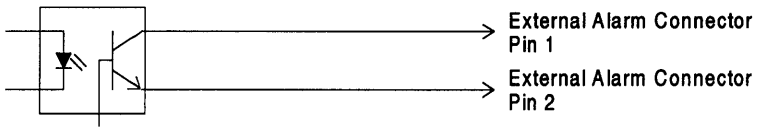


CMD Test Port: This is intended for use by CMD service personnel and has no application during normal operation of the controller.

External Alarm: You may connect an auxiliary alarm to the controller to supplement or replace the internal alarm The external alarm circuit should comply with the following power specifications.

Voltage	28 volts max. AC or DC
Current	150 milliamps max.

An opto-isolator in the controller makes it possible to use either AC or DC on the external alarm circuit, as the following schematic diagram shows:



You may disable the CRD-5400's internal alarm by removing the jumper shunt located next to the alarm, as shown in Figure 1-5.

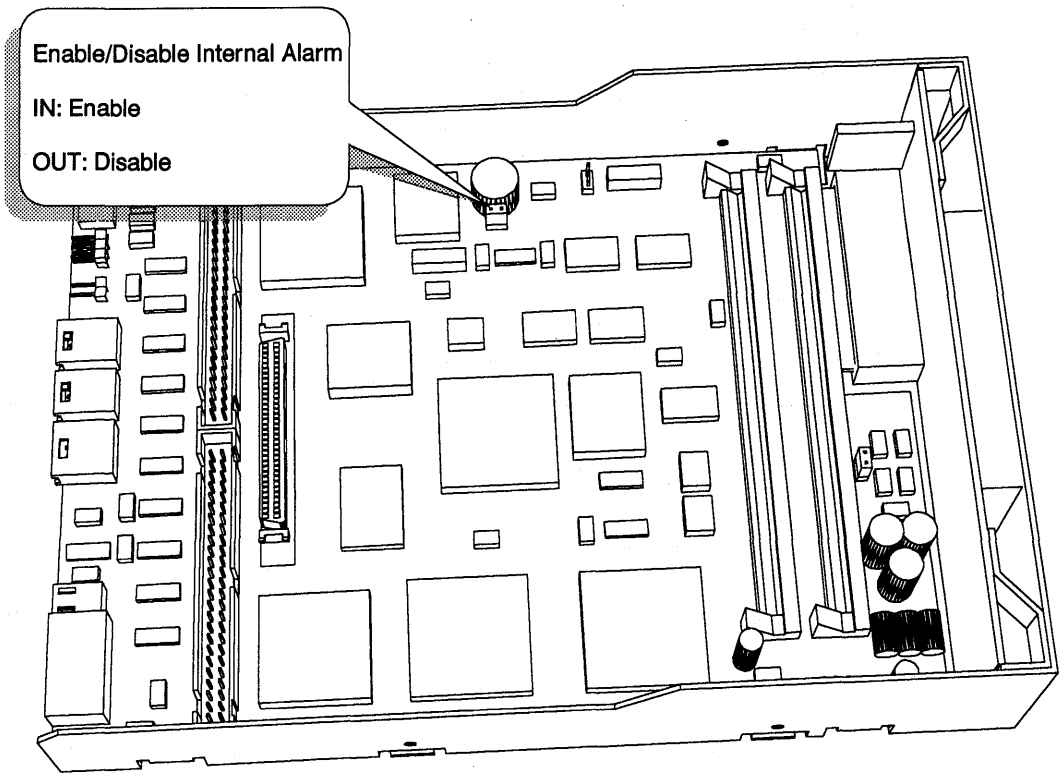


Figure 1-5: Enabling and disabling the internal alarm

1.3 Cache

The CRD-5400's cache accepts as many as four standard SIMMs for up to 256 megabytes of read and write memory.

Note
The CRD-5400 will not operate without at least one SIMM installed in the cache.

1.3.1 How To Order SIMMs

When you purchase SIMMs, they should meet the following specifications.

Speed	60 nanoseconds
Refresh Rate	2 ms
Bus Width	36 bits wide
Pins	72
Capacity	4 to 128 megabytes
IC Count	Not more than 36 chips
Parity	"True" parity

Note
The CRD-5400 does not support SIMMs with more than 36 integrated DRAM circuits (ICs) on board. Check your larger SIMMs, in particular any 128 MB SIMMs, to make sure they don't carry more than 36 ICs. The CRD-5400's self-test will reject SIMMs that do not support "true" parity. These SIMMs do not store parity. Instead, they calculate parity on the fly for each read and write from cache, report the result to the initiator, and then discard the value.

The capacity of 36-bit-wide SIMMs may not be apparent from the listings in merchandise catalogs. Use the following table as a guide.

Configuration	Size	Configuration	Size
1 x 36	4 megabytes	8 x 36	32 megabytes
2 x 36	8 megabytes	16 x 36	64 megabytes
4 x 36	16 megabytes	32 x 36	128 megabytes

1.3.2 Installing SIMMs

The CRD-5400 requires the installation of at least one SIMM to operate, and that SIMM must be installed in the slot closest to the front panel. You may install a second SIMM in the adjacent slot, but this slot need not be populated.

Warning
Be sure to remove all power (including battery power) to the controller before installing SIMMs. Before you touch a SIMM, discharge any static electricity on your body by touching an unpainted metal surface.

Figure 1-6: Installing SIMMs

1.4 Fan

The cooling fan is an essential component of the CRD-5400. Without it, the controller can overheat, which will trigger the warning alarm and ultimately cause the controller to shut itself down.

The fan is mounted in a special bracket, which attaches to the sides of the controller tray with a pair of screws. The fan's power cable connects to the header at W8, as shown in Figure 1-7.

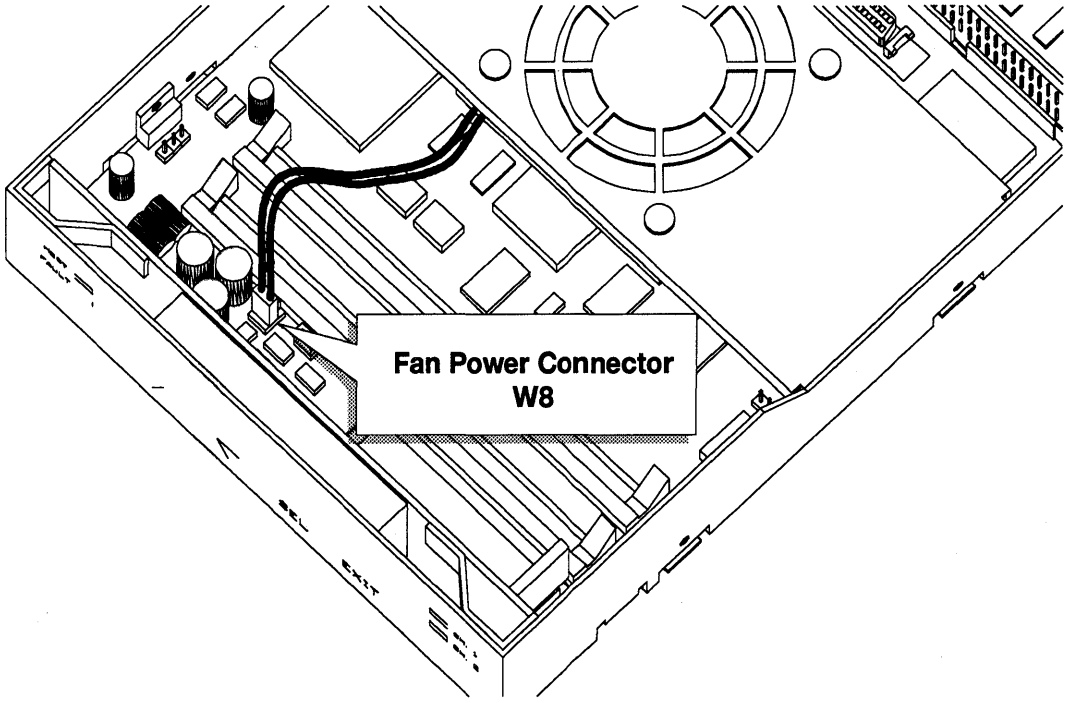


Figure 1-7: Cooling Fan Power Connection

This page left blank intentionally.

2 Special Controller Parameters

This chapter describes CRD-5400 parameters that are designed to be accessed by vendors only. These parameters can be changed only through the monitor utility and only after you enter a special vendor password. When you have secured "vendor privileges" by entering the vendor password, the special parameters will appear in the *Setup Parameters* screen. Users who do not enter the vendor password will not see these parameters.

2.1 Accessing Special Parameters

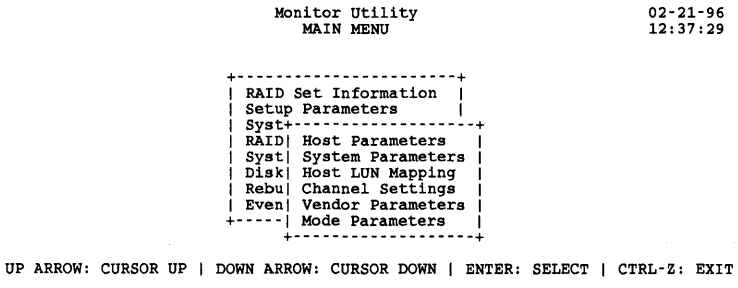
To gain access to the special parameters, enter your vendor password instead of your user password when prompted on the monitor utility's title screen.

The default vendor password is "vendor." Use the default password the first time you access the special parameters. Once you have "vendor privileges," you can change your vendor password by selecting in succession "System Functions," "Change Password," and "Vendor Password" in the monitor utility. The steps for changing your password are the same as those described in the CRD-5400 User's Manual for changing the user password. Your vendor password may contain up to nine alphanumeric characters.

If password protection is disabled, you may force the CRD-5400 to prompt you for a password at the title screen by pressing Ctrl-P. Enter your vendor password, and you will have immediate access to the special parameters. This saves the extra steps involved in entering the monitor utility to enable password protection and then backing out to the title screen to enter your vendor password.

Once you have modified the special parameters, you may restore the monitor utility to its standard user level by pressing Ctrl-Z until the title screen reappears. At this point, any user who enters the monitor utility without providing the vendor password will not have access to the special parameters.

The special parameters show up in the *Setup Parameters* screen.



2.2 Vendor Parameters

The Vendor Parameters option takes you to a screen where you can enter your own name and model number for the CRD-5400. Once you change these parameters, the information you enter will be displayed on the title screens of the front panel and monitor utility. You may also specify the strings the controller will return in response to a SCSI Inquiry command.

```
                    Monitor Utility                          02-20-96
                    VENDOR PARAMETERS                        11:57:24


+------------------------------+------------------+
| Parameter                    | Value            |
|------------------------------|------------------+
| System Display Vendor Name   | cmd technology   |
| System Display Model Name    | #(714)454-0800   |
| Inquiry Response Vendor Name | cmd-tech         |
| Inquiry Response Model Name  | crd-5400         |
+------------------------------+------------------+

ARROW KEYS: MOVE CURSOR | ENTER: SELECT | CTRL-Z: EXIT
```

2.3 Mode Parameters

The Mode Parameters screen provides a means to change the saved SCSI mode parameters related to the physical attributes and data format of the disk array or the SCSI bus disconnect/reconnect process timing and error recovery procedure to the host. The default values are designed to be compatible with the majority of host systems. There is the possibility, however, that some hosts will not communicate properly with the CRD-5400, unless these parameters are modified.

Warning
Do not attempt to change any mode parameter values unless you are familiar with SCSI mode parameters, or you are directed to by an authorized technical support engineer.

The following is an example of the Mode Parameters screen:

```
                    Monitor Utility                          02-09-96
                    MODE PARAMETERS                          13:04:33
                       Channel 0

+---+---+-----------------------------------------------------------------+
|   |   |                              Byte                               |
+---+---+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--|
|   |   | 2| 3| 4| 5| 6| 7| 8| 9|10|11|12|13|14|15|16|17|18|19|20|21|22|23|
+---+---+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--|
|   | 1 |26 1B 0C 00 00 00 00 00 00 00                                    |
|   | 2 |00 00 00 00 00 00 00 00 00 00 00 00 00 00                        |
|   | 3 |00 00 00 00 00 00 00 00 00 00 40 02 00 00 00 00 00 00 00 00 00 00|
| P | 4 |00 00 10 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|
| a | 7 |00 00 00 00 00 00 00 00 00 00 00                                 |
| g | 8 |04 00 00 00 00 00 00 00 00 00                                    |
| e | 9 |00 00 00 00 00 00                                                |
|   | A |00 10 00 00 00 00                                                |
|   | B |00 00 00 00 00 00                                                |
|   | C |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|
+---+---+-----------------------------------------------------------------+

N: NEXT CH | P: PREV CH | ENTER: SELECT | S: SAVE | D: DEFAULT | CTRL-Z: EXIT
```

Use the "N" and "P" keys to select a host channel. To modify a parameter value, use the arrow keys to position the cursor over the byte value you want to change and press Enter. Then use the arrow keys to increment or decrement the displayed value. When you reach the desired value, press Enter to save the value, or Ctrl-Z to abandon the change.

3 SCSI Commands

3.1 Test Unit Ready (00h)

Byte\Bit	7	6	5	4	3	2	1	0
0	00h							
1	LUN			0				
2	0							
3	0							
4	0							
5	0							

The Test Unit Ready command confirms that the logical unit is ready. If the logical unit is ready, the CRD-5400 returns a GOOD status. Otherwise, the CRD-5400 reports a CHECK CONDITION status. The sense data that is generated indicates the current status of the CRD-5400.

3.2 Rezero Unit (01h)

Byte\Bit	7	6	5	4	3	2	1	0
0	01h							
1	LUN			0				
2	0							
3	0							
4	0							
5	0							

The CRD-5400 does not implement the rezero unit command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.3 Request Sense (03h)

Byte\Bit	7	6	5	4	3	2	1	0
0	03h							
1	LUN			0				
2	0							
3	0							
4	Allocation length in bytes							
5	0							

The REQUEST SENSE command enables the initiator to request sense data resulting from a CHECK CONDITION status on the prior command. The CRD-5400 retains the sense data until the initiator requests it or until the same initiator issues another command to the same logical unit, at which time the sense data will be cleared.

The CRD-5400 does not send a CHECK CONDITION status in response to a request sense command made in error, unless the error was a fatal error. Examples of fatal errors include a nonzero reserved bit in the command descriptor block, an unrecovered parity error on the data bus, or a CRD-5400 malfunction that prevents the return of sense data. Sense data may be invalid following a fatal error on a request sense command.

Byte four of the command specifies the allocation length, which is the number of bytes the initiator has allocated for returned sense data. In the case of the CRD-5400, the allocation length should always be at least 18 bytes for the initiator to receive all sense data. Any other value indicates the maximum number of bytes that shall be transferred. The CRD-5400 terminates the Data In phase when allocation length bytes have been transferred or when all available sense data have been transferred to the initiator, whichever is less.

3.3.1 Extended sense data format

The CRD-5400 is capable of sending 18 bytes of extended sense data, and sends 18 bytes if the allocation length of the request sense command is equal or greater than 18 bytes (otherwise, the number of bytes specified by the allocation length will be sent). The extended sense data format is summarized in the following table.

Extended Sense Data								
Byte\Bit	7	6	5	4	3	2	1	0
0	Valid Bit	1	1	1	0			
		Error Class						
1	0 Segment Number							
2	0 Filemark	0 EOM	0 ILI	0	Sense Key			
3	Information Byte (MSB)							
4	Information Byte							
5	Information Byte							
6	Information Byte (LSB)							
7	Additional Sense Length							
8	0							
9	0							
10	0							
11	0							
12	Additional Sense Code (ASC)							
13	Additional Sense Code Qualifier (ASCQ)							
14	FRU Code							
15	FPV	C/D	0		BPV	Bit Pointer		
16	Field Pointer (MSB)							
17	Field Pointer (LSB)							
18-n	Product Unique Sense Data							

3.3.2 Sense Data Explanations

Valid Bit: This bit will be one if the information bytes (bytes 3-6) are valid and zero if they are not valid.

Error Class: Ones in these three bits indicate that extended sense is in use.

Segment Number: All bits contain zeros.

Filemark: This bit is always set to zero for the CRD-5400.

EOM: This is the "end of medium" indicator and is always set to zero for the CRD-5400.

ILI: The "incorrect length" indicator is always set to zero for the CRD-5400.

Sense Key: Indicates the CRD-5400's general error categories, which are listed in the next table. The additional sense code in byte 12 gives additional information about errors.

Information Bytes: When the valid bit is one, the information bytes will contain the the sense key's unsigned logical block address associated. The information bytes will contain the address of the current logical block unless otherwise specified.

Additional Sense Length: The length in bytes of additional sense data to follow. The allocation length in the command descriptor block must be sufficient to accommodate the additional sense data to avoid truncation.

Additional Sense Code/Additional Sense Code Qualifier: When the sense key is valid, gives additional information about errors.

FRU Code: The field replaceable unit code is for the use of field service personnel only.

FPV: When the field pointer valid bit is set to one, the C/D bit and bytes 16 and 17 are valid. These fields will be ignored when the FPV bit is zero.

C/D Bit: When the command/data bit is set to one, the value in the field pointer bytes identifies the byte number in the CDB that prompted an illegal request sense key. When the C/D bit is zero, the value reported in the field pointer bytes identifies the byte number in the data phase that prompted an illegal request sense key.

Field Pointer (MSB & LSB): When an illegal request sense key is issued due to an illegal parameter, this field gives the parameter's location in the command descriptor block or the data block. The next table provides detailed information about this field.

BPV: When the bit pointer valid bit is one, the next field—the bit pointer field—is valid.

Bit Pointer Field: This field pinpoints the bit that caused the illegal request sense key. A value of seven means the leftmost bit caused the error, and a zero means the rightmost bit caused the error. The byte in which the bit lies is identified by the field pointer field.

3.3.3 Sense Keys

Sense	Name	Explanation
0	NO SENSE	No particular sense key is present.
1	RECOVERED ERROR	The last executed command completed successfully with some recovery operation performed by the CRD-5400. When two or more errors occur and are recovered during processing of a command, the last is reported.
2	NOT READY	The disk drive is not accessible.
3	MEDIUM ERROR	An unrecoverable error was detected due to a defect in the medium or an error in the recorded data.
4	HARDWARE ERROR	The CRD-5400 detected the hardware error to which the recovery process cannot be applied during command execution or self-diagnostic test.
5	ILLEGAL REQUEST	An illegal value was detected in the CDB, in the parameter transferred, or the LUN is incorrect. When the CRD-5400 detects an illegal parameter in the CDB, the CRD-5400 terminates the command without rewriting the disk.
6	UNIT ATTENTION	The UNIT ATTENTION condition occurred.
B	ABORTED COMMAND	The CRD-5400 abnormally terminated the command being executed. Normally, the initiator can try recovery by reissuing the command.
E	MISCOMPARE	Source data did not match the data read from medium.

3.3.4 Sense and Subsense Codes

ASC	ASCQ	Name	Explanation	Sense Key
00	00	No additional sense information	No particular sense code is present	0
			An attempt was made to read the read prohibited area.	7
00	06	I/O process terminated	The I/O process has been terminated by a "Terminate I/O Process" message.	0
01	06	No index/sector signal	The index or sector signal was not detected in the specified period.	4
02	00	No seek complete	The seek or rezero seek operation did not	4

			complete in the specified period.	
03	00	Peripheral device write fault	Write operation to the disk abnormally terminated.	4
04	00	Logical unit not ready, cause not reportable	The disk drive is not accessible.	2
04	04	Logical unit not ready, format in progress	The drive is not accessible because it is being formatted.	2
08	01	Logical unit not ready, format in progress	A timeout occurred on a drive's internal interface.	2
08	02	Logical unit communication parity error	A parity error occurred on a drive's internal interface.	2
09	00	Track following error	The track crossing pulse was detected during the track following state.	4
0C	01	Write error recovered with auto reallocation	The error at write operation was recovered by the automatic alternate block allocation.	1
0C	02	Write error, auto reallocation failed	The automatic alternate block allocation process failed during the write operation.	3, 4
10	00	ID CRC or ECC error	A CRC error was detected in the ID field.	1, 3
11	00	Unrecovered read error	An unrecoverable error was detected when data was read.	3
11	04	Unrecovered read error, auto reallocation failed.	The automatic alternate block allocation process failed during the read operation.	3
12	00	Sync byte not found for ID field	Sync byte of the ID field cannot be detected.	1, 3
13	00	Sync byte not found for data field	Sync byte of the data field cannot be detected	3
14	01	Record not found	The desired data block (sector) could not be found	3
15	00	Random positioning error	Cylinder switching does not complete in the specified period.	1, 4
15	01	Mechanical positioning error	A seek error occurred on the drive.	1, 4
15	02	Positioning error detected by read or medium	The cylinder address of the ID field did not match.	1, 4
15	80	Settling error	After track switching/cylinder switching, the CRD-5400 is not in an on-track state.	1, 4
15	F0	Calibration error	The calibration seek or cylinder serve has failed.	4
17	01	Recovered data with retries	The data error was recovered by read retry.	1
17	02	Recovered read data with positive head offset	The data error was recovered by read retry accompanied by the head offset operation in the positive direction (positive direction is the outer direction on the disk).	1
17	03	Recovered read data with negative head offset	The data error was recovered by read retry accompanied by the head offset operation in the negative direction is the inner direction on the disk).	1
18	00	Recovered read data with error correction applied	The data error was immediately recovered by ECC correction.	1
18	01	Recovered read data with error correction and retries applied	The data error was recovered by the ECC correction after retry applied.	1
18	02	Recovered read data with	The data error was recovered by ECC	1

		error correction and/or retries, data auto-reallocation	correction and the automatic alternate block allocation process was applied.	
18	80	Recovered read data with error correction and/or retries, rewrite applied	The data error was recovered by ECC correction and rewriting to the same block.	1
19	00	Defect list error	An error was detected when the defect list (G list) was read	3
1A	00	Parameter list length error	The initiator sent a parameter of incorrect length	5
1B	00	Synchronous data transfer error	An error was detected in synchronous data transfer—either an abnormal period of ACK signal or an ACK signal response broken REQ/ACK offset.	4
1C	01	Primary defect list not found	An error was detected when the defect list (P list) was read.	5
1D	00	COMPARE ERROR	Miscompare during verify operation.	E
20	00	Invalid command operation code	CDB byte 0 (operation code) is invalid	5
21	00	Logical block address out of range	A logical block address exceeding the maximum value of the drive was specified.	5
24	00	Invalid field in CDB	Setting in the CDB is incorrect.	5
25	00	Logical unit not supported	Invalid LUN was specified.	5
26	00	Invalid field on parameter list	Setting of the parameter list transferred from the initiator during command execution is invalid.	2
27	00	Write protected	An attempt was made to write in the write-prohibited area.	7
29	00	Power-on, RESET, or BUS DEVICE RESET occurred	State immediately after power-on, state after RESET condition, or BUS DEVICE RESET message.	6
2A	00	MODE parameters changed	Another initiator changed the MODE SELECT parameter value.	6
2A	02	Log parameters changed	Log parameters have changed by another initiator.	6
31	00	Medium format corrupted	The medium format is different from the original one. (Formatting was not performed after the data format setting was changed with the MODE SELECT command.)	3
32	00	No defect spare location available	No useable alternate block area is present, or the alternate block process cannot be performed due to overflow of the control table.	4
32	01	Defect list update failure	Updating of the defect list (G list) failed.	4
37	00	Rounded parameter	The MODE SELECT parameter specified by the command was rounded.	1
3D	00	Invalid bits in IDENTIFY message	1 was specified for the reserve bit of the IDENTIFY message.	5
3E	00	Logical unit has not self-configured yet	The CRD-5400's initial setup operation is not complete.	2
3F	01	Microcode has been changed	The CRD-5400's microcode has been changed by another initiator.	6

40	nn	Diagnostic failure on component "nn"	An error was detected in self-diagnostic test.	4
43	00	Message error	The message sent from the CRD-5400 was rejected.	B
44	00	Internal target failure	A hardware error was detected in the CRD-5400.	4
45	00	Select/reselect failure	Response waiting timeout for the initiator was detected in RESELECTION phase.	1, B
47	00	SCSI parity error	A parity error was detected in the SCSI parity data bus.	1, B
48	00	INITIATOR DETECTED ERROR message received	The INITIATOR DETECTED ERROR message was received from the initiator.	1, B
49	00	Invalid message error	Unsupported or illegal message was received.	B
4C	00	Logical unit failed self-configuration	The CRD-5400's initial setup failed. (System space information could not be read).	4
4C	80	Initial seek failed	Initial seek failed and the initial setup of the CRD-5400 cannot be performed.	4
4E	00	Overlapped commands attempted	A new command was issued from the same initiator to the same logical unit before execution of a command was completed with tagged queuing disabled.	B
5B	01	Threshold condition met	Log parameter threshold condition has been met.	6
5C	00	Rotational Position Locking (RPL) status changed	Status of the spindle synchronization function has changed.	6
5C	01	Spindles synchronized	All spindles have synchronized. (In an array environment).	6
5C	02	Spindles have lost synchronization	All spindles in an array did not achieve synchronization in the required time or at least one spindle lost synchronization.	6

3.4 Format Unit (04h)

Byte\Bit	7	6	5	4	3	2	1	0
0	04h							
1	LUN			0				
2	0							
3	0							
4	0							
5	0							

The CRD-5400 does not implement the format unit command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.5 Read (08h)

Byte\Bit	7	6	5	4	3	2	1	0
0	08h							
1	LUN			Logical block address (MSB)				
2	Logical block address							
3	Logical block address (LSB)							
4	Transfer block count							
5	0							

The read command transfers data from the CRD-5400 to the initiator. The first block of data read is specified in the logical block address field of the CDB. The command continues transferring data in contiguous logical blocks until it reaches the transfer block count specified in byte 4.

If the transfer block count field is set to zero, the read command returns 256 logical data blocks. If the field is set to a number other than zero, the command transfers that number of logical blocks.

3.6 Write (0Ah)

Byte\Bit	7	6	5	4	3	2	1	0
0	0Ah							
1	LUN			Logical block address (MSB)				
2	Logical block address							
3	Logical block address (LSB)							
4	Transfer block count							
5	0							

The write command instructs the CRD-5400 to write data sent by the initiator to the array. The data is written in contiguous logical blocks, starting with the logical block specified in the logical block address field of the CDB and comprising the total number of blocks specified in the transfer block count field.

If the transfer block count field is set to zero, the command transfers 256 logical data blocks to the array. If the field is set to a number other than zero, the command transfers that number of logical blocks.

3.7 Seek (6) (0Bh)

Byte\Bit	7	6	5	4	3	2	1	0
0	0Bh							
1	LUN			Logical Block Address (MSB)				
2	Logical Block Address							
3	Logical Block Address (LSB)							LSB
4	0							
5	0							

The CRD-5400 does not implement the seek (6) command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.8 Inquiry (12h)

Byte\Bit	7	6	5	4	3	2	1	0
0	12h							
1	LUN			0				
2	0							
3	0							
4	Allocation Length in Bytes							
5	0							

The Inquiry command requests information about the identity and characteristics of the CRD-5400.

This command executes successfully even in the presence of a Unit Attention condition. It also executes if the CRD-5400 is not ready or if the specified LUN is invalid.

The following situations will cause a CHECK CONDITION status to be reported and an abnormal termination:

- The CDB contains a specification error in other than the LUN field.
- A CRD-5400 hardware error makes the inquiry data inaccessible.
- An unrecoverable error is present on the SCSI bus.
- The overlap command's exception conditions are applied.

Standard Inquiry Data

Byte\Bit	7	6	5	4	3	2	1	0
0	Qualifier 0			Device type code 0				
1	RMB 0	Device type qualifier 0						
2	ISO Vers 0		ECMA Version 0			ANSI Version 0	1	0
3	AENC 0	0	TmIOP 0	0	Response Data Format 0	0	1	0
4	1Fh (Additional data length)							
5	0							
6	0							
7	Reladr 0	WB32 0	WB16 0	Sync 1	Linkd 0	0	Cmd Que 1	SfRe 0
8-15	"CMD TECH" (Vendor ID in ASCII)							
16-31	"CRD-5400" (Product ID in ASCII)							
32-35	Microcode version in ASCII Product Revision in ASCII							

3.8.1 Data Descriptions

Qualifier	
000	The type of I/O device represented by the specified logical unit matches the "device type code" field. The logical unit does not need to be ready for this code to be sent.
001	The type of I/O device represented by the specified logical unit matches the "device type code" field
011	The specified logical unit is not being supported. The "device type code" field indicates 1Fh when this code is sent.
Device type code	
00000	Direct access device
11111	Undefined device
RMB bit	
0	When equal to 1, the storage medium is replaced. The CRD-5400 acts as a fixed disk, so this bit is always 0.
Device type qualifier	
000000	All bits are set to 0.
SCSI standard version	
010	The second byte in the data block represents the SCSI standard recognized by the CRD-5400. Since the CRD-5400 is a SCSI-2 device, it reports "010" for ANSI X3T9.86.
ISO Vers	
0	Always set to zero.
ECMA Version	
0	Always set to zero.
AENC	
0	Asynchronous Event Notification Capability. The CRD-5400 does not support this function.
TmIOP	
0	Terminate I/O Process message. The CRD-5400 does not support this message.
Response data format	
0010	The format of the Inquiry data is identified by this field. The CRD-5400 reports all data in SCSI-2 format, so this field will always equal "0010."
Additional data length	
1Fh	The length of bytes of additional inquiry data to follow.
RelAdr	
0	Relative Logical block addressing. The CRD-5400 does not support this function.
WBus32	
0	Data transfers take place on a 32-bit-wide bus. The CRD-5400 does not support this function.
WBus16	
0	Data transfers take place on a 16-bit-wide bus. The CRD-5400 does not support this function.
Sync	
1	Synchronous-mode data transfer. The CRD-5400 supports this function.
Linked	
0	Command linking. The CRD-5400 does not support this function.
Cmd Que	
1	Command queuing with tag. The CRD-5400 supports this function.

SftRe	
0	Soft RESET condition. The CRD-5400 does not support this function.
Vendor ID	
CMD TECH	This field indicates the product supplier's name in left-justified ASCII code and will always contain "CMD TECH."
Product ID	
CRD-5400	This field indicates the product model name in left-justified ASCII code and will always contain "CRD-5400."
Product revision	
	This field contains the CRD-5400 microcode revision number in ASCII code.

3.9 Mode Select (15h)

Byte\Bit	7	6	5	4	3	2	1	0
0	15h							
1	LUN			PF	0			SP
2	0							
3	0							
4	Parameter list length							
5	0							

The Mode Select command allows the initiator to configure various CRD-5400 parameters.

When the page format bit in byte 1 of the CDB is 1, the initiator is signaling that is transferring parameters in the page descriptor format.

The save pages (SP) bit in byte 1 of the CDB controls whether the parameters in the command will be saved on the disk array. An SP bit of 1 will prompt all parameters to be saved on the execution of the command. If the SP bit is 0, the parameters will not be saved.

The parameter list length field specifies the length in bytes of the parameter list being transferred from the initiator. When this field is 0, no data is transferred and the command terminates with no error reported. All parameter lists must conform to the format and length specified in the "Mode Select Parameters" table later in this section.

A parameter list length that does not exactly equal the actual length of the parameter list being transferred causes the command to terminate. All parameters that were transferred before the termination are invalid, and the CRD-5400 reports a CHECK CONDITION status (ILLEGAL REQUEST Invalid Field in CDB).

There are three types of mode select parameter values: current values, save values and default values. The current values are the parameters that actually control the CRD-5400 and are what the mode select command changes. The save values are those parameters that have been specified by the mode select command and saved on the disk array. The default values serve as current values immediately after power-up and until any saved values can be read. If there are no saved values, the default values remain in effect until a mode select command is issued.

The current values are initialized to the saved values at power-on, when a RESET occurs, or when the CRD-5400 receives a BUS DEVICE RESET message. If there are no saved values, the current values are initialized to the default values.

If the mode select command modifies a parameter page that is common to all initiators, UNIT ATTENTION conditions (MODE parameters changed) occur for all the initiators except the initiator issuing the command.

If the mode select command seeks to change an unchangeable parameter, the command will terminate with a CHECK CONDITION status (ILLEGAL REQUEST Invalid field in parameter list). In this case, all parameters specified by the initiator will be invalid.

The next three tables show the proper configuration of a mode select parameter list. A complete parameter list comprises a four-byte header, an eight-byte block descriptor, and one or more page descriptors. Also acceptable are parameter lists consisting of only a four-byte header and one or more page descriptors. The initiator may transfer the header only or the header and block descriptor only.

Header								
Byte\Bit	7	6	5	4	3	2	1	0
0	00h							
1	00h (medium type)							
2	device-specific parameter (00h)							
3	00h/08h (block descriptor length)							

Block Descriptor								
Byte\Bit	7	6	5	4	3	2	1	0
0	00h							
1	Data block count (MSB)							
2	Data block count							
3	Data block count (LSB)							
4	00h							
5	Data block length (MSB)							
6	Data block length							
7	Data block length (LSB)							

Page Descriptor								
Byte\Bit	7	6	5	4	3	2	1	0
0	0		Page Code					
1	Page length							
2-n	Parameter field							

3.9.1 Header

Medium Type: 00h (default type) must be specified in this field.

Device-Specific Parameters: 00h must be specified in this field.

Block Descriptor Length: The length (in bytes) of the block descriptor to follow is specified in this field. Do not include the length of the page descriptors. The CRD-5400 recognizes only one block descriptor. When a block descriptor is included, specify 08h in the block descriptor length field. When the initiator is sending a parameter list containing only a header followed by page descriptors, specify 00h in the block descriptor length field.

3.9.2 Block Descriptor

The logical characteristics of the data on the array may be specified in the eight-byte block descriptor.

Data Block Count: Specify the total number of logical blocks allocated to the user space on the disk array. When this field is zero, all user spaces set by the format parameter and drive parameter are configured with the logical data blocks. This field must be zero.

Data Block Length: The length (in bytes) of the logical data blocks on the disk is specified in this field. The CRD-5400 requires that the logical data block length be equal to the physical data block length.

3.9.3 Page Descriptors

A page descriptor comprises a page code byte, the page length byte, and a parameter field. Each parameter function attribute is specified on a separate page. The initiator can use the MODE SELECT command to specify one page descriptor or two or more page descriptors in an arbitrary order.

Page Code: Specifies the page descriptor type indication code (page number).

Page Length: Specifies the length (in bytes) of the parameter field. The length excludes the page header, so measurement begins from byte 2. The value that the initiator specifies for this field must equal the page length sent by the CRD-5400 in response to a MODE SENSE command.

3.9.4 CRD-5400 Mode Select Parameters

The following table lists the contents and length of the parameter list that is transferred from the initiator to the CRD-5400 with the MODE SELECT command. If the initiator specifies a page descriptor that is not supported by the CRD-5400, the command will terminate with a CHECK CONDITION status (ILLEGAL REQUEST Invalid field in parameter list) and all parameters specified in the command will be invalidated.

Parameter (page number)	Length
Header (Mode Select/Mode Select Extended)	4
Block descriptor	0/8
Read/write error recovery parameter (01)	12
Disconnect/reconnect parameter (02)	16
Format parameter (03)	24
Drive parameter (04)	24
Verify error recovery parameter (07)	12
Caching parameter (08)	12
Peripheral device page (09)	8
Control Mode page (0A)	8
Medium support page (0B)	8
Notch page (0C)	24
All page descriptors supported (3F)	152/160

3.10 Reserve Unit (16h)

Byte\Bit	7	6	5	4	3	2	1	0
0	16h							
1	LUN			3rd pty	3rd pty Dev ID			0
2	0							
3	0							
4	0							
5	0							

The reserve unit command enables individual initiators in a multiple initiator environment to reserve logical units in the disk array. An initiator may also use this command to reserve a logical unit for another SCSI device.

3.10.1 Logical unit reserve function

This command reserves the entire logical unit specified in the CDB for the exclusive use of the initiator until the reservation is superseded by another reserve unit command, the initiator issues a release unit or priority reserve command, a bus device reset message is issued by any initiator, a reset condition occurs, or power to the CRD-5400 is recycled.

When an initiator issues this command for a LUN that is already reserved by another initiator, the command terminates with RESERVATION CONFLICT status.

Once an initiator has reserved a LUN, the CRD-5400 rejects any commands other than INQUIRY, REQUEST SENSE and RELEASE UNIT from any other initiator and reports a RESERVATION CONFLICT status. The INQUIRY and REQUEST SENSE commands execute normally. The RELEASE UNIT command terminates with a GOOD status, but the CRD-5400 ignores the command if the initiator does not have the reservation right.

3.10.2 Reservation right and third party reserve function

If the 3rd Pty bit of CDB byte 1 is 0, the initiator reserves the specified LUN on the CRD-5400 and claims the reservation right for the same LUN.

If the 3rd Pty bit is 1, the initiator reserves the specified LUN on the CRD-5400 for another SCSI device, which is identified by its SCSI ID in the 3rd Pty Dev ID bit of the CDB. The reservation right always stays with the initiator, even though the LUN on the CRD-5400 is reserved for another device. The CRD-5400 maintains the LUN reservation until it is superseded by another valid Reserve command from the initiator that made the reservation or until it is released by the same initiator, by a Bus Device Reset message from any initiator, or by a "hard" Reset condition.

3.10.3 Superseded Reserve

An initiator may modify the reservation it holds by issuing another, superseding Reserve command for the same logical unit. Upon the successful execution of the superseding Reserve command, the reserved state of the CRD-5400 will conform to the new command. If the superseding Reserve command cannot be executed, the reserved state is unchanged. The superseding Reserve command permits the initiator to reserve the CRD-5400 for another SCSI device, without relinquishing its reservation right.

The following examples will help to further explain reservation rights and third party reservations.

Example 1: When initiator-A gives the RESERVE UNIT command without setting the 3rd Pty bit to 1, it exclusively reserves the CRD-5400 and holds the reservation right to the CRD-5400. Any other initiator's commands, except for INQUIRY, REQUEST SENSE, and RELEASE UNIT, are rejected by the CRD-5400

with a RESERVATION CONFLICT status. The RELEASE UNIT command terminates successfully, but the CRD-5400 takes no action in response.

Example 2: Initiator-A gives the RESERVE UNIT command containing a 3rd Pty bit of 1 and the 3rd Pty Dev ID for target/initiator-1, initiator-A reserves the CRD-5400 for target/initiator-1 but retains the reservation right for the CRD-5400. In this situation, if target/initiator-1 tries to send a RESERVE UNIT command, the CRD-5400 rejects the command with a RESERVATION CONFLICT status. A RELEASE UNIT command from target/initiator-1 terminates normally, but the CRD-5400 disregards it. Initiator-A can issue the INQUIRY, REQUEST SENSE, RELEASE UNIT or RESERVE UNIT commands to the CRD-5400, but other commands are rejected with a RESERVATION CONFLICT status. A RELEASE UNIT command from initiator-A clears the reserved state of the CRD-5400, and a RESERVE UNIT command changes the reserved state of the CRD-5400. If any other initiator attempts to issue any commands other than INQUIRY, REQUEST SENSE or RELEASE UNIT, the CRD-5400 will respond with a RESERVATION CONFLICT status. A RELEASE UNIT command from any other initiator terminates normally, but the CRD-5400 disregards it.

3.11 Release Unit (17h)

Byte\Bit	7	6	5	4	3	2	1	0
0	17h							
1	LUN			3rd Pty	3rd Pty Dev ID			0
2	X	X	X	X	X	X	X	X
3	00h							
4	00h							
5	0							

The RELEASE UNIT command ends the initiator's reservation of a CRD-5400 logical unit. If an initiator attempts to release a logical unit that it has not reserved, the CRD-5400 just disregards the command and returns a GOOD status.

3.11.1 Third party release

When the 3rd Pty bit of the CDB is 0, the command releases the specified logical unit that the initiator reserved with a previous RESERVE UNIT command that also had a 3rd Pty bit of 0.

When the CDB's 3rd Pty bit is 1, the command releases the specified logical unit, but only if the reservation was made using the third party reservation option by the initiator that is requesting the release, and for the same SCSI device specified in the third party ID field.

3.12 Mode Sense (1Ah)

Byte\Bit	7	6	5	4	3	2	1	0
0	1Ah							
1	LUN			0	DBD	0		
2	PC		Page Code					
3	0							
4	Transfer byte length							
5	0							

The mode sense command returns the values of various parameters related to the physical attributes and data format of the disk array or the SCSI bus disconnect/reconnect process timing and error recovery procedure to the initiator.

The data the CRD-5400 sends to the initiator in response to this command consists of a header, block descriptor, and one page descriptor for each parameter specified.

The Disable Block Descriptors (DBD) bit of byte 1 in the CDB controls whether the mode sense data returned will contain a block descriptor. If this bit is 0, the response will contain a header, block descriptor and one or more page descriptors. If this bit is 1, everything but the block descriptor will be sent.

The page code of the page descriptor is specified in the page code field in byte 2. The following table lists the parameters supported by the CRD-5400 and their page codes.

Parameter (page number)	Length
Header (Mode Select/Mode Select Extended)	4
Block descriptor	0/8
Read/write error recovery parameter (01)	12
Disconnect/reconnect parameter (02)	16
Format parameter (03)	24
Drive parameter (04)	24
Verify error recovery parameter (07)	12
Caching parameter (08)	12
Peripheral device page (09)	8
Control Mode page (0A)	8
Medium support page (0B)	8
Notch page (0C)	24
All page descriptors supported (3F)	152/160

If the value of the page code field is 3Fh, all page descriptors supported by the CRD-5400 are sent to the initiator. If the value is not 3Fh and is associated with a parameter supported by the CRD-5400, the page descriptor for that parameter is sent. If the value identifies a parameter that the CRD-5400 does not support, the command terminates with a CHECK CONDITION status (ILLEGAL REQUEST Invalid field in CDB).

The Page Control (PC) field of CDB byte 2 specifies the type of page descriptor parameter value sent by the mode sense command. The following table lists the PC values supported by the CRD-5400.

PC	Type of parameter sent to initiator
00	Current value: The CRD-5400 responds with the current value of each parameter. The current values can be set in three ways: 1) by a successful execution of the mode select command; 2) by retrieving the saved values of the mode parameters if a MODE SELECT command has not successfully completed since the last power-on, hard RESET condition, or BUS DEVICE RESET message; 3) or by being initialized with the default values at power up if no saved values are present.
01	Changeable value: This value indicates the parameter field/bits that may be changed on the CRD-5400. A changeable field/bit position will be indicated with a 1. A field/bit that cannot be changed will be represented with a 0.
10	Default value: The CRD-5400 responds with the default value of each parameter.
11	Saved value: The CRD-5400 responds with the saved value of each parameter.

The transfer byte-length field specifies the total number of bytes of mode sense data transferred. The CRD-5400 transfers the number of bytes of mode sense data set by the page code field, or the amount of mode sense data that is specified in the transfer byte-length field, whichever is smaller to the initiator. When the transfer byte-length field is set to 0, no data is transferred and the command terminates.

The following table illustrates the configuration of the parameter list resulting from a mode sense command. Each parameter lists comprises a 4-byte header, 8-byte block descriptor and one or more page descriptors. If the DBD bit is set to 1, the block descriptor is not sent. The page descriptor is not sent when the page code is set to 00h.

Header								
Byte\Bit	7	6	5	4	3	2	1	0
0	Sense data length							
1	00h (medium type)							
2	0							
3	00h/08h (block descriptor length)							

Block descriptor								
Byte\Bit	7	6	5	4	3	2	1	0
0	00h							
1	Number of data blocks (MSB)							
2	Number of data blocks							
3	Number of data blocks (LSB)							
4	00h							
5	Data block length (MSB)							
6	Data block length							
7	Data block length (LSB)							

Page descriptor								
Byte\Bit	7	6	5	4	3	2	1	0
0	PS	0	Page Code					
1	Page Length							
2-n	Parameter Field							

3.12.1 Header

Sense data length: This field gives the length (in bytes) of the parameter list compiled in response to the mode sense command. The length is measured from byte 1 of the header and does not include the length of the sense data length field itself. The CRD-5400 reports the length of the parameter lists required to fully satisfy the mode sense command. To ensure that all parameter lists have been received, the initiator should compare the value of this field with the transfer byte-length field of the sense data CDB. If the sense data length plus the length of the sense data length field itself is greater than the transfer byte-length value, then the initiator received a truncated mode sense parameter list.

Medium type: 00h (default type) is always reported to this field.

Block descriptor length: This field denotes the length in bytes of the block descriptor. The measurement does not include the header or page descriptor. When the DBD bit is 0, the CRD-5400 reports 08h in this field to inform the initiator that a set of block descriptors follows the header. When the DBD bit is 1, the CRD-5400 reports 00h in this field.

3.12.2 Block descriptor

Number of data blocks: This field indicates the number of logical blocks available to the user in the logical unit. Any spare sectors set aside for the alternative block process will not be included in this value.

Data block length: This field denotes the length in bytes of each logical block.

3.12.3 Page descriptor

The descriptor for each page of mode parameters begins with a 2-byte header, followed by the parameter field.

PS bit: When set to one, the Parameters Savable (PS) bit denotes that the page has savable parameters. When set to 0, the PS bit indicates that none of the parameters contain within the page are savable. All page parameters supported by the CRD-5400 can be saved.

Page length: This field denotes the length in bytes of the parameter field, excluding the page header and page descriptor.

Parameter field: Byte 2 and succeeding bytes indicates the parameter values corresponding to the type requested in the page control field of the CDB.

3.13 Start/Stop Unit (1Bh)

Byte\Bit	7	6	5	4	3	2	1	0
0	1Bh							
1	LUN			0				Immed
2	0							
3	0							
4	0							Start
5	0							

When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.14 Send Diagnostic (1Dh)

Byte\Bit	7	6	5	4	3	2	1	0
0	1Dh							
1	LUN			0				
2	0							
3	0							
4	0							
5	0							

The CRD-5400 does not implement the send diagnostic command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.15 Prevent Allow Medium Removal (1Eh)

Byte\Bit	7	6	5	4	3	2	1	0
0	1Eh							
1	LUN			0				
2	0							
3	0							
4	0							Prevent X
5	0							

The CRD-5400 does not implement the prevent allow medium removal command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.16 Read Capacity (25h)

Byte\Bit	7	6	5	4	3	2	1	0
0	25h							
1	LUN			0				
2	0							
3	0							
4	0							
5	0							
6	0							
7	0							
8	0							
9	0							

The Read Capacity command permits the initiator to request information about the capacity of a CRD-5400 logical unit.

The format of the data returned by the read capacity command is as follows:

Byte\Bit	7	6	5	4	3	2	1	0
0	Logical block address (MSB)							
1	Logical block address							
2	Logical block address							
3	Logical block address (LSB)							
4	Block size (MSB)							
5	Block size							
6	Block size							
7	Block size (LSB)							

3.17 Read Extended (28h)

Byte\Bit	7	6	5	4	3	2	1	0
0	28h							
1	LUN			0				
2	Logical block address (MSB)							
3	Logical block address							
4	Logical block address							
5	Logical block address (LSB)							
6	0							
7	Transfer block count (MSB)							
8	Transfer block count (LSB)							
9	0							

Like the read command, the read extended command transfers data to the initiator; however, the read extended command accepts a four-byte logical block address and a two-byte transfer block count. The logical block address field indicates the first logical block in the transfer. The command transfers a contiguous set of logical blocks, the number of which is determined by the transfer block count field.

If the logical block address and transfer block count fields describe a transfer that exceeds the maximum number of logical blocks on the CRD-5400, the command terminates with a CHECK CONDITION status (ILLEGAL REQUEST Logical block address out of range) and abort the disk read operation.

3.18 Write Extended (2Ah)

Byte\Bit	7	6	5	4	3	2	1	0
0	2Ah							
1	LUN			0				
2	Logical Block Address (MSB)							
3	Logical Block Address							
3	Logical Block Address							
5	Logical Block Address (LSB)							
6	0							
7	Transfer block count (MSB)							
8	Transfer block count (LSB)							
9	0							

The write extended command transfers data from the initiator to the CRD-5400, but unlike the write command, it accepts a four-byte logical block address and a two-byte transfer block count. The logical block address field indicates the first logical block in the transfer. The command transfers a contiguous set of logical blocks, the number of which is determined by the transfer block count field.

If the logical block address and transfer block count fields describe a transfer that exceeds the maximum number of logical blocks on the CRD-5400, the command terminates with a CHECK CONDITION status (ILLEGAL REQUEST Logical block address out of range) and aborts the operation.

When the transfer block count field is set to zero, the command performs a seek operation without writing any data on the cylinder/track where the data block in the logical block address is located.

3.19 Seek (10) (2Bh)

Byte\Bit	7	6	5	4	3	2	1	0
0	2Bh							
1	LUN			0				
2	Logical Block Address (MSB)							
3	Logical Block Address							
3	Logical Block Address							
5	Logical Block Address (LSB)							
6	0							
7	0							
8	0							
9	0							

The CRD-5400 does not implement the seek (10) command. When issued this command by the host, the CRD-5400 takes no action and responds with a GOOD status.

3.20 Write and Verify (2Eh)

Byte\Bit	7	6	5	4	3	2	1	0
0	2Eh							
1	LUN			0				
2	Logical block address (MSB)							
3	Logical block address							
4	Logical block address							
5	Logical block address (LSB)							
6	0							
7	Transfer block count (MSB)							
8	Transfer block count (LSB)							
9	0							

The CRD-5400 does not support verification of data. This command is treated like a write extended (2Ah) command.

3.21 Verify (2Fh)

Byte\Bit	7	6	5	4	3	2	1	0
0	2Fh							
1	LUN			0			Bytchk 0	0
2	Logical block address (MSB)							
3	Logical block address							
4	Logical block address							
5	Logical block address (LSB)							
6	0							
7	Block count (MSB)							
8	Block count (LSB)							
9	0							

The verify command instructs the CRD-5400 to verify the data stored in one or more logical data blocks. The logical block address field defines the first block to be verified, and the block count field defines the number of contiguous logical data blocks to be verified. No data is transferred.

The bytchk bit must be set to zero. The CRD-5400 will perform a medium verification (CRC and ECC) only.

3.22 Read Defect Data (37h)

Byte\Bit	7	6	5	4	3	2	1	0
0	37h							
1	LUN			0	0	0	0	0
2	0	0	0	PList	GList	Defect List Format		
3	00h							
4	00h							
5	00h							
6	00h							
7	Transfer byte length (MSB)							
8	Transfer byte length (LSB)							
9	0	0	0	0	0	0	0	0

This command transfers the list (defect data) that describes the defect position information of the disk to an initiator.

The initiator can specify the type of defect data to be transferred to the initiator with the PList (primary list) bit and GList (growth list) bit on CDB, and the format of defect data with the "defect list format" field.

PList	GList	Defect data format
1	1	PList and GList
1	0	PList only
0	1	GList only
0	0	4-byte header information only (explained in this section)

Defect list format			Defect data format
0	0	0	Block address format
1	0	0	Format of byte distance from index
1	0	1	Physical sector address format

The "transfer byte-length" field of CDB specifies the length (bytes) of defect data that is received by the initiator. When the CRD-5400 completes the transfer of defect data corresponding to the length specified in the "transfer byte-length" field or completes transfer of all defect data of the specified type, the CRD-5400 terminates data transfer. When zero is specified in the "transfer byte-length" field, data transfer is not executed and this command terminates.

The following table shows the format of defect data to be transferred to the initiator by this command.

Byte\Bit	7	6	5	4	3	2	1	0
0	00h							
1	0	0	0	PList	GList	Defect List Format		
2	Defect list length (MSB)							
3	Defect list length (LSB)							
4-n	Defect descriptor							

When the CRD-5400 receives this command, it will return the first four bytes of the defect data list header with the defect list length set to zero. No defect descriptor will be sent by the CRD-5400.

This page left blank intentionally.

4 SCSI Messages

The following messages are classified into three groups: messages from an initiator to a target (I), messages from a target to an initiator (T), and messages that can be sent or received by a target or initiator (T/I).

4.1 Command Complete (00h) T

This message signals the initiator that the execution of an I/O process has been completed and that a valid status has been sent to the initiator. The target transmits this message after the STATUS phase when the specified command execution has been completed.

This target will send this message even if the I/O process was not successfully executed. The status of the I/O process will be indicated in the status byte sent during the STATUS phase.

After sending this message, the target releases the BSY signal and enters the BUS FREE phase. The target considers the message transmission to be successful if the ATN signal is false at the time the ACK signal for this message is negated.

4.2 Save Data Pointer (02h) T

This message instructs the initiator to store the active data pointer into the saved data pointer for the current I/O process. The target will sometimes send a SAVE DATA POINTER message during disconnection, as described in the next section.

4.3 Disconnect (04h) T

With the disconnect message, the target informs the initiator that it intends to break the current connection temporarily and reconnect later to complete the I/O operation. When the target confirms that the initiator received message, it disconnects from the SCSI bus by going to BUS FREE phase. The target will deem the message transmission to be successful if the ATN signal is false at the time the ACK signal for this message is negated.

This message does not prompt the initiator to save the active data pointer. When the target disconnects during data transfer, it sends a SAVE DATA POINTER message before sending this DISCONNECT message.

4.4 Initiator Detected Error (05h) I

With this message, the initiator signals the target that it has detected an error that does not preclude the target from retrying the operation. The error may be due to previous SCSI bus operation or the operation inside the initiator.

4.5 Abort (06h) I

This message directs the target to clear the active I/O process and any queued I/O process for the selecting initiator on the specified logical unit. When the target receives this message, it immediately performs the following (regardless of the ATN signal status):

- If the LUN is specified prior to this message, the target clears the I/O process that is being executed or is queued on that logical unit and has been activated by the initiator that has sent this message, and enters the BUS FREE phase. All of the currently retained data and status related to the I/O process are cleared, and the retained sense data (if any) related to the logical unit for that initiator is also cleared. The initiator is not informed of the status byte and completion message regarding the I/O process cleared by this message. This message does not affect I/O processes which have been activated by other initiators.
- If the LUN has not been specified prior to this message, the target enters the BUS FREE phase without any other operation. Therefore, this message does not affect the I/O process that is being executed or is queued.

The target enters the BUS FREE phase without generating error conditions even if this message has been issued to the logical unit in which there is no active or queued I/O process, or no pending sense data.

After selecting the target (by the SELECTION phase), the initiator can send this message following the IDENTIFY message to clear the I/O process which is currently DISCONNECTed.

Previously established conditions, including MODE SELECT parameters, reservations, and synchronous data transfer agreement are not changed by the ABORT message.

NOTE
The BUS DEVICE RESET, CLEAR QUEUE, ABORT and ABORT TAG messages provide a means to clear one or more I/O processes prior to normal termination. The BUS DEVICE RESET message clears all I/O processes for all initiators on all logical units. The CLEAR QUEUE message clears all I/O processes for all initiators on the specified logical unit. The ABORT message clears all I/O processes for the selecting initiator on the specified logical unit. The ABORT TAG message clears the current I/O process only.

4.6 Message Reject (07h) I/T

This message indicates that the last message or message byte received was inappropriate or has not been implemented.

When the initiator sends this message, it must assert the ATN signal before negating the ACK signal for the message byte to be rejected that was received in the MESSAGE IN phase to permit the target to IDENTIFY the rejected message byte. If the target receives this message under any other circumstance, it must reject this message.

When the target sends this message, it must enter the MESSAGE IN phase and send this message prior to requesting additional message bytes in the MESSAGE OUT phase to permit the initiator to IDENTIFY the rejected message byte.

After a target sends this message and if the ATN signal is still asserted, then it must return to the MESSAGE OUT phase. The subsequent MESSAGE OUT phase must begin with the first byte of the message.

4.7 No Operation (08h) I

This message does not result in any operation. The initiator sends this MESSAGE IN response to the message request from the target (in the MESSAGE OUT phase) when the initiator does not currently have any other valid message to send.

For example, if the target does not respond to the ATTENTION condition until a later phase and at that time the original message is no longer valid, the initiator may send this message when the target enters the MESSAGE OUT phase.

4.8 Message Parity Error (09h) I

This message informs the target that the message byte received last by the initiator had a parity error.

To permit the target to IDENTIFY the message byte in which an error was detected, the initiator must assert the ATN signal before negating the ACK signal that corresponds to the message-byte containing the parity error in the MESSAGE IN phase. The target must negate the BSY signal to enter the BUS FREE phase without executing any further INFORMATION TRANSFER phase when it receives this message under any other circumstance.

4.9 Bus Device Reset (0Ch) I

This message directs a clearing of all I/O processes on that target. The target which receives this message clears not only I/O operations initiated by the connected initiator, but also I/O operations related to all initiators and then the target must enter the BUS FREE phase (regardless of the ATN signal status).

4.10 Abort Tag (0Dh) I

The ABORT TAG message is implemented in support of tagged queuing. The target will go to the BUS FREE phase following successful receipt of this message. The target will clear the current I/O process, and the execution will be halted. The medium contents may have been modified before the execution was halted. In either case, any pending status or data of the I/O process will be cleared and no status or ending message will be sent to the initiator. Pending status, data and commands for other active or queued I/O processes will not be affected. Execution of other I/O processes queued for the initiator will not be aborted.

Previously established conditions, including MODE SELECT, parameters, and reservations will not be changed by the ABORT TAG message.

4.11 Clear Queue (0Eh) I

The CLEAR QUEUE message is implemented in support of tagged queuing. The target will go to the BUS FREE phase following successful receipt of this message. The target will perform an action equivalent to receiving a series of ABORT messages from each initiator. All I/O processes, from all initiators in the queue for the specified logical unit will be cleared from the queue. All active I/O processes will be terminated. The medium may have been altered by partially executed commands. All pending status and data for the logical unit or target routine for all initiators will be cleared. No status or message will be sent for any of the I/O processes. A unit ATTENTION condition will be generated for all other initiators with I/O processes that either were active or were queued for that logical unit or target routine. When reporting the unit

ATTENTION condition, the additional sense code will be set to COMMANDS CLEARED BY ANOTHER INITIATOR.

4.12 Queue Tag Messages

QUEUE TAG message format												
Byte\ Bit	7	6	5	4	3	2	1	0				
0	Message Code (20h, 21h, 22h)											
1	Queue Tag											

The preceding table defines the format for queue tag messages. Tagged queuing is defined only for logical units, not target routines.

The queue tag messages are used to specify an identifier, called a queue tag, for an I/O process. The queue tag field is an 8-bit unsigned integer assigned by the initiator during an initial connection. The queue tag for every I/O process for each initiator should be unique. A queue tag becomes available for re-assignment when the I/O process ends. The numeric value of a queue tag has no effect on the order of execution.

Whenever an initiator connects to a target, the appropriate queue tag message will be sent immediately following the IDENTIFY message and within the same MESSAGE OUT phase to establish the tag for the I/O process. Only one tag may be established during a connection. If a queue tag message is not sent, then the I/O process is treated as an untagged command.

Whenever a target reconnects to an initiator to continue a tagged I/O process, the SIMPLE QUEUE TAG message will be sent immediately following the IDENTIFY message and within the same MESSAGE IN phase to revive the initiator connection for the I/O process. Only one I/O process may be revived during a reconnection. If the SIMPLE QUEUE TAG message is not sent, then an untagged command is revived for the I/O process.

4.12.1 Head of Queue Tag (021h) I

The HEAD OF QUEUE TAG message specifies that the I/O process be placed first in that logical unit's command queue. An I/O process already being executed by the target will not be pre-empted. A subsequent I/O process received with a HEAD OF QUEUE TAG message will be placed at the head of the command queue for execution in last-in, first-out order.

4.12.2 Ordered Queue Tag (022h) I

The ORDERED QUEUE TAG message specifies that the I/O process be placed in that logical unit's command queue for execution in the order received. All queued I/O processes for the logical unit received prior to this I/O process will be executed before this I/O process is executed. All queued I/O processes received after this I/O process will be executed after this I/O process, except for I/O processes received with a HEAD OF QUEUE TAG message.

4.12.3 Simple Queue Tag (020h) I/T

The SIMPLE QUEUE TAG message specifies that the I/O process be placed in that logical unit's queue. The order of execution is determined by the CRD-5400.

4.13 Identify (80h to FFh) I/T

Bit	7	6	5	4	3	2	1	0
	1	D	0	0	0	LUN		

This message specifies the LUN for the device under the target and establishes a logical connection among the initiator, target and logical unit.

Bit 6: DISCONNECT Privilege—Only the initiator can set this bit to one. When this bit is one, it indicates that the initiator permits the target to execute DISCONNECTion processing. When this bit is zero, the target must not execute DISCONNECTion operation. When the target sends this message, this bit must be zero.

Bit 2 to 0: LUN—These bits specify the LUN for the device under the target.

Function of message—This message is normally sent immediately after the SELECTION phase from the initiator to the target; it specifies the logical unit number for the I/O process. The target must send this message immediately after the RESELECTION phase to inform the initiator of the logical unit number to be reconnected.

When one SELECTION or RESELECTION sequence has successfully established an I/O operation path between the initiator and the target, only one LUN can be specified and another IDENTIFY message specifying a different LUN must not be issued until the SCSI bus is released (generating a BUS FREE phase).

When the initiator receives this message from the target in the reconnection sequence, it must store the saved pointers of the specified logical unit into the active pointers before completing the transfer phase of this message (before negating the ACK signal).

4.14 Synchronous Data Transfer Request (I/T)

Byte	
0	01h (Extended MSG)
1	03h (Extended MSG length)
2	01h (Sync Data Request)
3	Transfer Period [4 x m (ns)]
4	REQ/ACK Offset

Parameters for synchronous mode data transfer are defined by exchanging this message between two SCSI devices.

When a SCSI device with the synchronous mode transfer implemented is connected to another SCSI device for the first time after it receives the BUS DEVICE RESET message, encounters the RESET condition ("hard" RESET), or power is turned on, data transfer mode between the two SCSI devices is negotiated by exchanging the SYNCHRONOUS DATA TRANSFER REQUEST message. Each SCSI device must respond to this message initiated by another SCSI device. (The SCSI devices may exchange this message to set or change the data transfer mode at a time other than first connection.)

4.14.1 Data transfer mode parameters

This message exchange establishes the permissible Transfer Period and REQ/ACK offset for data transfer between the two SCSI devices. These values are applicable for all logical units on the two SCSI devices.

The Transfer Period is the minimum repetition cycle for REQ and ACK pulses that is permissible for data reception operation of the SCSI devices. (It is the minimum time between the leading edge of a REQ pulse

and the leading edge of the next REQ pulse, or between the leading edge of an ACK pulse and the leading edge of the next ACK pulse.)

The REQ/ACK offset is the maximum number of REQ pulses (offset value) which can be sent by the target before receiving an ACK pulse response (leading edge of the ACK signal) from the initiator. The selected value must not cause an overflow of a SCSI device data reception buffer and offset counter. A REQ/ACK offset of zero indicates asynchronous data transfer mode, and a value of FFh indicates unlimited REQ/ACK offset.

When the SCSI device sends the SYNCHRONOUS DATA TRANSFER REQUEST message the first time, the device specifies the Transfer Period and REQ/ACK offset values within the range where data can be received successfully. After receiving the SYNCHRONOUS DATA TRANSFER REQUEST message, the SCSI device sends back the message with the same Transfer Period and REQ/ACK offset values if it is possible to receive data with the specified parameter values (or smaller Transfer Periods or larger REQ/ACK offsets or both). If a greater Transfer Period or smaller REQ/ACK offset value is required to receive data successfully, the SCSI device substitutes only the parameter value required to be changed to a value which satisfies the SCSI device condition and sends back the SYNCHRONOUS DATA TRANSFER REQUEST message.

When executing data transfers, SCSI devices must not send REQ or ACK pulses that exceed the parameter values specified by the other SCSI device at the SYNCHRONOUS DATA TRANSFER REQUEST message exchange. Data transfer with a greater Transfer Period value and/or a smaller REQ/ACK offset value is allowed.

After the completion of the SYNCHRONOUS DATA TRANSFER REQUEST message exchange, both SCSI devices must set the data transfer mode as shown in the following table by the response (sending back the message) of the SCSI device that received the message the first time.

Responded Message	Data Transfer Mode
SYNCHRONOUS DATA TRANSFER REQUEST REQ/ACK offset does not equal zero	Synchronous mode: Each SCSI device executes data transfer using the Transfer Period value equal to or greater than the specified value in the SYNCHRONOUS DATA TRANSFER REQUEST message from the other SCSI device, or using the REQ/ACK offset value equal to or smaller than the specified value in the SYNCHRONOUS DATA TRANSFER REQUEST message from the other SCSI devices.
SYNCHRONOUS DATA TRANSFER REQUEST REQ/ACK offset equals zero	Asynchronous mode
MESSAGE REJECT	Asynchronous mode

4.14.2 Message exchange procedures for initiator

When the initiator recognizes that a synchronous data transfer has to be negotiated, it asserts the ATN signal to initiate the message exchange and requests the target for the SYNCHRONOUS DATA TRANSFER REQUEST message. When the MESSAGE OUT phase is successfully completed, the target must respond with the SYNCHRONOUS DATA TRANSFER REQUEST message or the MESSAGE REJECT message to the initiator. If the ATN signal is still true at the completion of receiving the SYNCHRONOUS DATA TRANSFER REQUEST message, the target can terminate the MESSAGE OUT phase and enter the MESSAGE IN phase to return the message. If the target fails to return the message, both SCSI devices must go to asynchronous transfer mode for data transfers.

When the initiator asserts the ATN signal in the MESSAGE IN phase and sends the MESSAGE PARITY ERROR message or the MESSAGE REJECT message to the target, data must be transferred in the asynchronous mode even if the target has returned the SYNCHRONOUS DATA TRANSFER REQUEST message with the REQ/ACK offset greater than 0, because both the initiator and the target regard this outcome as a failed synchronous transfer negotiation.

If the MESSAGE PARITY ERROR message is sent from the initiator, however, the target can retry to return the message (SYNCHRONOUS DATA TRANSFER REQUEST message) for the data transfer mode negotiation. If the retry is successful, the initiator and target must assume that synchronous mode transfer has been negotiated. If the MESSAGE PARITY ERROR message is still sent from the initiator after executing as many retries as defined, the target must terminate the retry operation by either entering another INFORMATION TRANSFER phase and transferring at least one byte of information or entering the BUS FREE phase. (If the CRD-5400 operates as the target, it enters the BUS FREE phase.) The initiator considers this condition a message exchange failure. Therefore both SCSI devices must go to asynchronous transfer mode for data transfers between the two devices.

4.14.3 Validity of data transfer mode

The data transfer mode and the synchronous mode transfer parameters established as a result of the SYNCHRONOUS DATA TRANSFER REQUEST message exchange must be kept valid until any of the following occur.

- Reception of the BUS DEVICE RESET message
- Occurrence of the RESET condition ("hard" RESET)
- Change of the parameter or the transfer mode between the same SCSI devices
- Power off

Default mode for data transfer is the asynchronous mode. The mode for data transfer must be initialized to the asynchronous mode after power on, the BUS DEVICE RESET message was received, or the RESET condition occurred.

When the data transfer mode is changed from synchronous to asynchronous mode on the SCSI device that has completed the setting of synchronous mode transfer for some reason that the other SCSI device cannot find, the SCSI device must re-negotiate the transfer mode by sending the SYNCHRONOUS DATA TRANSFER REQUEST message.

Re-negotiating the data transfer mode at every SELECTION is not suggested, due to negotiation's negative impact on performance.

4.15 Wide Data Transfer Request (I/T)

Byte\Bit	7	6	5	4	3	2	1	0
0	Extended message (01h)							
1	Extended message length (02h)							
2	WIDE DATA TRANSFER REQUEST code (03h)							
3	Transfer width exponent							

A WIDE DATA TRANSFER REQUEST (WDTR) message exchange is initiated by a SCSI device whenever a previously arranged transfer width agreement may have become invalid. The agreement becomes invalid after any condition that may leave the data transfer agreement in an indeterminate state such as:

- after a hard reset condition
- after a BUS DEVICE RESET message
- after a power cycle

In addition, a SCSI device may initiate a WDTR message exchange whenever it is appropriate to negotiate a new transfer width agreement. SCSI devices that are capable of wide data transfers (greater than eight bits) will not respond to a WDTR message with a MESSAGE REJECT message.

Note
Renegotiation at every selection is not recommended, since a significant performance impact is likely.

The WDTR message exchange establishes an agreement between two SCSI devices on the width of the data path to be used for DATA phase transfers between the two devices. This agreement applies to DATA IN and DATA OUT phases only. All other information transfer phases will use an eight-bit data path.

If a SCSI device implements both wide data transfer option and synchronous data transfer option, then it will negotiate the wide data transfer agreement prior to negotiating the synchronous data transfer agreement. If a synchronous data transfer agreement is in effect, then a SCSI device that accepts a WDTR message will reset the synchronous agreement to a asynchronous mode.

The transfer width is two to the transfer width exponent bytes wide. The transfer width that is established applies to all logical units on both SCSI devices. Valid transfer widths are 8 bits (m = 00h), 16 bits (m = 01h), and 32 bits (m = 02h). Values of m greater than 02h are reserved.

The originating SCSI device (the SCSI device that sends the first of the pair of WDTR messages) sets its transfer width value to the maximum data path width it elects to accommodate. If the responding SCSI device can also accommodate this transfer width, it returns the same value in its WDTR message. If it requires a smaller transfer width, it substitutes the smaller value in its WDTR message. The successful completion of an exchange of WDTR messages implies an agreement as follows:

Responding Device WDTR Response	Implied Agreement
Non-zero transfer width	Each device transmits and receives data with a transfer width equal to the responding SCSI device's transfer width
Transfer width equal to zero	Eight-bit data transfer
MESSAGE REJECT message	Eight-bit data transfer

If the initiator recognizes that negotiation is required, it asserts the ATN signal and sends a WDTR message to begin the negotiating process. After successfully completing THE MESSAGE OUT phase, the target responds with the proper WDTR message. If an abnormal condition prevents the target from returning an appropriate response, both devices go to eight-bit transfer mode for data transfers between the two devices.

Following target response, the implied agreement for wide data transfers will be considered to be negated by both the initiator and the target if the initiator asserts ATN and the first message out is either MESSAGE PARITY ERROR or MESSAGE REJECT. In this case, both devices will go to eight-bit data transfer mode for data transfers between the two devices.

Index

This page left blank intentionally.