

SDC-RQD11-RLL

**5-1/4" Winchester Disk
Drive Controller
Manual**



SDC-RQD11-RLL
5-1/4" Winchester Disk
Drive Controller
Manual

Copyright © Sigma Information Systems

3407 E. La Palma Avenue

Anaheim, California 92806

MA401350 REV A - April 1989

Notes

Table of Contents

| | |
|--|-----------|
| 1. General Description..... | 1 |
| 1.1 Controller Specifications | 7 |
| 2. Installation..... | 9 |
| 2.1 Unpacking and Inspection..... | 9 |
| 2.2 Factory Configurations..... | 10 |
| 2.3 ST506 Interface Connections | 11 |
| 2.4 Jumper Plug Settings..... | 12 |
| 2.4.1 Base (CSR) Address | 12 |
| 2.4.2 Automatic Bootstrap Select | 13 |
| 2.4.3 Interrupt Priority..... | 13 |
| 2.5 Optional Accessories..... | 14 |
| 2.5.1 Drive Signal & Control Cable Set | 15 |
| 2.5.2 RS232 Maintenance Terminal Adaptor | 16 |
| 2.5.3 Front Panel Connections | 16 |
| 2.6 On-Board Bootstrap..... | 18 |
| 3. Programming..... | 21 |
| 3.1 Overview of MSCP | 21 |
| 3.2 Controller Communications | 22 |
| 3.2.1 Command and Response Rings..... | 24 |
| 3.2.2 Message Packets | 25 |
| 3.3 Message Transmission..... | 27 |
| 3.3.1 Command Transmission | 27 |
| 3.3.2 Response Transmission..... | 27 |
| 3.3.3 Interrupts | 27 |
| 3.4 Data Transmission..... | 28 |
| 3.5 Initialization | 28 |
| 3.5.1 Initialization Process | 28 |
| 3.5.2 Initialization Parameters..... | 29 |
| 3.6 Registers..... | 30 |
| 3.6.1 Initialize and Poll Register (IP)..... | 30 |
| 3.6.2 Status and Address Register (SA) | 31 |
| 3.7 MSCP Commands..... | 31 |
| 3.8 Error Handling..... | 32 |
| 3.9 Fatal Controller Error..... | 34 |

| | | |
|-----------|---------------------------------------|-----------|
| 4. | WOMBAT Utilities..... | 35 |
| 4.1 | Introduction to WOMBAT Utilities..... | 35 |
| 4.2 | Outalk..... | 35 |
| 4.3 | Starting Up Wombat..... | 36 |
| 4.4 | Setting Up The Disk Structure..... | 39 |
| 4.5 | Wombat Menu Options | 40 |
| 4.5.1 | Disk Structure Menu | 42 |
| 4.5.2 | Test Disk Menu | 51 |
| 4.5.3 | Bad Block Management Menu..... | 54 |
| 4.5.4 | Initialize Controller | 56 |
| 4.5.5 | Position Head on Shipping Zone..... | 56 |
| 4.6 | Wombat Error Messages..... | 56 |
| 4.7 | Wombat Self Diagnostics | 58 |

List of Figures

| | |
|---|----|
| Figure 2-1: Connector And Jumper Locations..... | 10 |
| Figure 2-2: Base (CSR) Address Configurations – Jumpers A1-A6.... | 12 |
| Figure 2-3: Automatic Bootstrap Select – Jumpers B1-B3..... | 13 |
| Figure 2-4: Interrupt Level Configurations – Jumpers P1-P3 | 14 |
| Figure 2-5: Drive/controller Cabling | 15 |
| Figure 3-1: Memory Communications Format | 23 |
| Figure 3-2: Descriptor Format | 24 |
| Figure 3-3: 16-bit Word-Aligned Message Envelope Format | 25 |

List of Tables

| | |
|--|----|
| Table 2-1: 34-pin Control Connector (J1) | 11 |
| Table 2-2: 20-pin Data Connector (J2 and J3) | 11 |
| Table 2-3: RS232 Maintenance Terminal Adapter (DLV11J Compatible) | 16 |
| Table 2-4: Front Panel Connector | 17 |
| Table 2-5: CSR Addresses for Bootstrap | 20 |
| Table 3-1: Initialization Operations | 30 |

1. General Description

The Sigma SDC-RQD11-RLL is a dual height Q-bus interface to ST506/412 compatible 5-1/4 inch Winchester disk drives. Implementing DEC's Mass Storage Control Protocol (MSCP), the SDC-RQD11-RLL couples any size disk to all standard DEC operating systems without software modification. Comprehensive on-board interactive formatting and diagnostic firmware provides engineering support across the range of LSI-11, MicroVAX, and various non-DEC implementations of the Q-bus.

ST506/412 Interface The Seagate ST506/412 interface has become the de facto industry standard for 5-1/4 inch Winchester disk drives. Inexpensive, reliable units spanning capacities of 2 to 138 megabyte and access times of 20 to 200 millisecond are available.

Q-bus Interface Originally introduced in 1975 by Digital Equipment Corporation to support the LSI-11 CPU range, the Q-bus architecture has evolved in speed and functionality to the point where it now outperforms most small computer bus systems. The SDC-RQD11-RLL fully implements all current Q-bus enhancements, including block mode transfers and 22-bit addressing, and flexibly supports LSI-11/2, LSI-11/23, LSI-11/73, MicroVAX I, MicroVAX II, Motorola 68000 and National 32032 Q-bus CPU designs.

-
- Block Mode DMA** When used with block mode memory, the SDC-RQD11-RLL almost doubles Q-bus throughput by interleaving address references with bursts of data, fully conforming with Q-bus Block Mode DMA protocol. With non block mode memory, the SDC-RQD11-RLL reverts automatically to simple DMA.
- MSCP Universal Disk Architecture** The SDC-RQD11-RLL communicates with the software through a simple register pair to memory resident command packets. Disk geometry factors such as sectors, heads and cylinders are invisible to the host computer as the SDC-RQD11-RLL accepts 32-bit binary block numbers, converting them to physical disk addresses. Disk capacity is inherently communicated back and never assumed by the software, therefore any size disk may be fully accessed without software modification. Supported operating systems include RT-11 version 5, RSX-11M plus version 2.1, TSX-plus version 4, RSTS/E version 8, Micro-VMS and UNIX.
- Seek Optimization** Queuing of up to 32 commands is permitted within the SDC-RQD11-RLL. The optimum order of execution of these packets is dynamically computed to minimize disk head movement and enhance throughput in heavily loaded systems.
- Manufacturer's Defect Map** Manufacturer's Media Defect information is entered by the user. WOMBAT takes the data entered by the user the defect map information and replaces defective blocks immediately.

| | |
|--|---|
| Bad Block Replacement | Disk surface defects are detected and flagged by the SDC-RQD11-RLL during formatting and pattern testing, and redirected to a user-specified reserved area at the end of the disk. During all subsequent operations the SDC-RQD11-RLL simulates fault-free media through transparent bad block replacement. |
| Automatic Bad Block Replacement | Dynamic bad block replacement presents error-free media to the host computer. During normal operation the controller replaces any blocks it detects as bad with an alternative block from a replacement block pool. Bad blocks can be displayed and manually or automatically replaced. Bad block replacement is transparent to the host computer. |
| Transparent Read Retry | Four Error Correction Code (ECC) bytes are added to each disk sector during write, and verified during read. On miscompare, the SDC-RQD11-RLL automatically repeats the read operation up to 10 times before attempting ECC correction of the disk data. ECC corrects error bursts up to 22 bits in length, which considerably reduces the number of irrecoverable errors reported to the operating system. |
| Multi Drive Capability | Two ST506/412 compatible 5 1/4" winchester disk drives, of any capacity, may be connected to each SDC-RQD11-RLL controller, and up to three controllers may co-exist in one system. To reduce access time with multiple disk drives, the SDC-RQD11-RLL will overlap seek by having the drives seek simultaneously. |

- Multiple Logical Units** The SDC-RQD11-RLL supports one or two ST506/412 drives. Each drive may support up to 8 logical units with unit numbers in the range 0-126. There is no restriction on the size of any unit.
- On-Board Bootstrap** A miniature programming plug permits the SDC-RQD11-RLL to bootstrap on power-up. The bootstrap can be set to boot to DU, DL, DY, MS or MU, with a default boot to DU0. The bootstrap option can be disabled.
- Overlapped Seeks** When two drives are attached to the SDC-RQD11-RLL controller, and multiple commands are sent to the controller by the host, the seeks of the disk drives will be overlapped by the controller, i.e., if one drive is seeking, an I/O request for the other drive may be issued. If this is a seek, the first drive whose seek finishes will then have its I/O request done.
- ECC** The firmware utilizes a 56-bit error correction code (ECC), allowing error bursts of up to 22 bits in one 512 byte block to be corrected.
- Data Buffer** All data transfers are staged through a track buffer to ensure that 'data late' errors will not occur during periods of heavy Q-bus traffic. Transfer rate is smoothly adjusted downwards until the bus again becomes available.

| | |
|--------------------------------|---|
| Outtalk | WOMBAT outloads a communication program from its program ROM into the host memory, which then manages communication between the console and WOMBAT. Communication between WOMBAT and a console terminal enables WOMBAT diagnostics and formatting capability on systems that have the console terminal as an integral part of the CPU (e.g., KDF11-B, KDJ11-B, MicroVAX). |
| Command Queue Size | The commands queue is requested for both drives. The queue size of the first drive to be put on-line is used. The first drive to be put on-line is defined as drive zero. |
| Head Stepping Rates | Drive head stepping rates can be selected at 2.1us, 11us, and 24us - or from .33ms to 4.3ms in 0.5ms increments. |
| Cylinder Numbers | The SDC-RQD11-RLL supports disks with up to 2048 cylinders with 15 as the maximum number of heads. |
| Controls and Indicators | At the rear edge of the circuit board is a red LED indicator to signal Board Failure and a green LED to signal Access in Progress. An output is provided for off-board indication of Access, and inputs are provided for two optional Write Protect switches. |
| WOMBAT Utilities | WOMBAT is a set of interactive formatting, diagnostic and debug utilities totally contained within the SDC-RQD11-RLL firmware. An on board serial connection is provided for communication with an ASCII terminal, permitting disk formatting and maintenance operations to be carried out with no other hardware present. |

No external software, media, or program loading device is required in maintenance of the SDC-RQD11-RLL or its attached disk drive.

WOMBAT Formatter WOMBAT initializes a fresh disk drive by writing sector addresses and zero data blocks through the entire recording surface. On invoking the formatter, the user is prompted at the terminal to supply parameters such as numbers of cylinders, heads and sectors, sector interleave factor, bad block replacement capacity, positioner step rate, and shipping zone cylinder. This data is stored twice in reserved areas of track zero during the format process, and retrieved by a simple homeseek-read sequence at each power-up. No special PROMs or switch settings are required to fully characterize the connected disk drive.

WOMBAT Self-diagnostics On bootstrap, the SDC-RQD11-RLL is prompted by the operating system to enter a comprehensive series of controller and disk confidence tests. On failure, a red on board LED is illuminated to highlight the faulty module.

WOMBAT Interactive Diagnostics Terminal oriented engineering utilities contained within the WOMBAT firmware include a continuous read / write / seek exerciser, a disk surface pattern tester, and a bad block replacement routine.

1.1 Controller Specifications

| | |
|--------------------------------------|--|
| Bus interface: | DEC Q-bus |
| Transfer mode: | Block mode DMA |
| Memory address capacity: | 4 megabyte (22-bit) |
| Software emulation: | DEC Mass Storage Control Protocol |
| Command buffer capacity: | Up to 32 MSCP commands |
| Data buffer capacity: | 512 bytes (one sector) |
| CSR address: | 172150,160334,160354,160374 160414,160434,jumper selectable |
| Interrupt vector: | Software selectable |
| Interrupt priority: | Level 4 through 7 jumper selectable |
| Q-bus loads: | 1 DC, 2 AC |
| Drive interface: | Seagate ST506/412 |
| Access time overhead: | 3 mS (plus drive access time) |
| Single block transfer rate: | 937.5 Kbyte/sec |
| Full track transfer rate: | 468 Kbyte/sec |
| 56-bit Error Correction Code: | ECC allows error bursts of up to 22 bits in one 512 byte block to be corrected |
| Automatic bootstrap: | 173000, jumper selectable |
| Disk connectors: | One 34-pin control, two 20-pin data |
| Power requirement: | 5 volt 2.6 amp typical |

On-board LED indicators: RED - board failure
GREEN - disk access in progress

Output: Disk access in progress

TTL inputs: Write Protect Drive 0 switch
Write Protect Drive 1 switch

2. Installation

2.1 Unpacking and Inspection

The SDC-RQD11-RLL is shipped in a special packing carton designed to keep the module from vibrating and to give it maximum protection during shipment. The packing carton should be retained in case the unit requires reshipment. The packing carton should contain the following:

- | | |
|------------------------|---|
| P/N 401350 | SDC-RQD11-RLL dual-wide module |
| MA 401350 | Manual entitled "SDC-RQD11-RLL, 5-1/4" Winchester Disk Drive Controller" |
| | Optional single drive configuration cable kit with one 20-pin data cable and one 34-pin control cable (also available in 8" length) |
| | Optional dual drive configuration cable kit with two 20-pin data cables and one 34-pin control cable daisy chained to second drive (controller-to-drive-to-drive) (also available in 8" length) |
| | Optional dual drive configuration cable kit with two 20-pin data cables and two 34-pin control cables (controller-to-drive and drive-to-drive) |
| P/N 700757-0115 | Optional maintenance adapter |

Unpack the SDC-RQD11-RLL and visually inspect for physical damage. If any damage has occurred, contact the factory immediately.

2.2 Factory Configurations

The SDC-RQD11-RLL is shipped with switch and jumper configurations as shown in Figure 2-1. Verify that these configurations are correct. If other configurations are required, refer to the appropriate paragraphs in this section.

Figure 2-1 shows the locations of connectors and jumpers that are used to configure the SDC-RQD11-RLL.

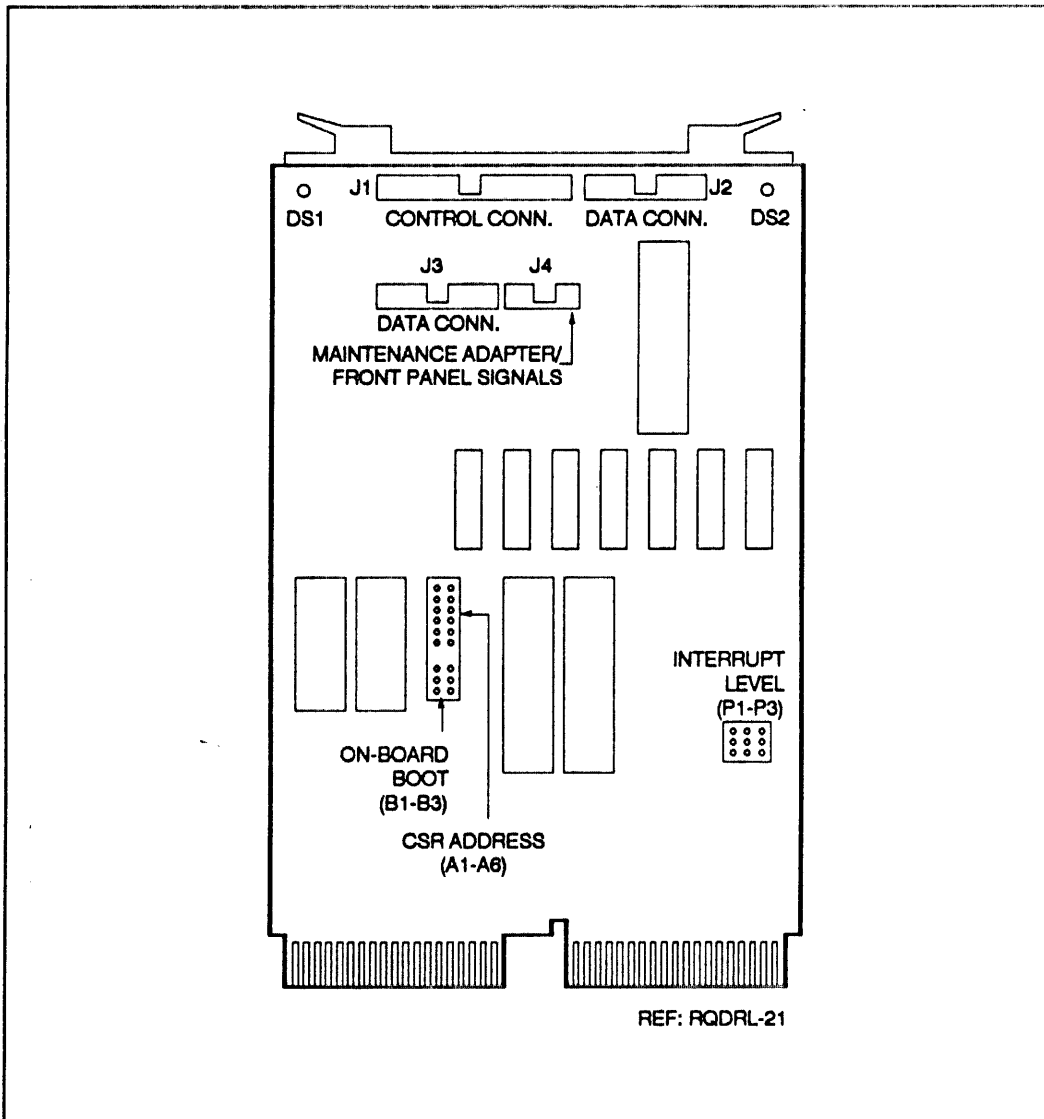


Figure 2-1:
Connector And Jumper Locations

2.3 ST506 Interface Connections

The following tables define the pinouts for the 34-pin control and 20-pin data connectors.

| Function | Pin | Function | Pin |
|----------------|---|-----------------|-----|
| *SEEK COMPLETE | 8 | *DRIVE SELECT 1 | 26 |
| *TRACK 0 | 10 | *DRIVE SELECT 2 | 28 |
| *WRITE FAULT | 12 | *DRIVE SELECT 3 | 30 |
| *INDEX | 20 | *DRIVE SELECT 4 | 32 |
| *READY | 22 | *HEAD SELECT 0 | 14 |
| *WRITE GATE | 6 | *HEAD SELECT 1 | 18 |
| *STEP | 24 | *HEAD SELECT 2 | 4 |
| *DIRECTION IN | 34 | *HEAD SELECT 3 | 2 |
| RESERVED | 16 | | |
| GROUND | 1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33 | | |
| * = ACTIVE LOW | | | |

Table 2-1:
 34-pin Control Connector (J1)

| Function | Pin | Function | Pin |
|-----------------|------------------------------|----------------|---------|
| *DRIVE SELECTED | 1 | +RLL READ DATA | 17 |
| +RLL WRITE DATA | 13 | -RLL READ DATA | 18 |
| -RLL WRITE DATA | 14 | RESERVED | 3,5,7,9 |
| GROUND | 2,4,6,8,10,11,12,15,16,19,20 | | |
| * = ACTIVE LOW | | | |

Table 2-2:
 20-pin Data Connector (J2 and J3)

2.4 Jumper Plug Settings

Miniature movable configuration plugs permit easy selection of base address, automatic bootstrap select, and interrupt priority.

2.4.1 Base (CSR) Address

Jumpers A1-A6 define the CSR address for LSI-11 and MicroVAX systems as shown in Figure 2-2.

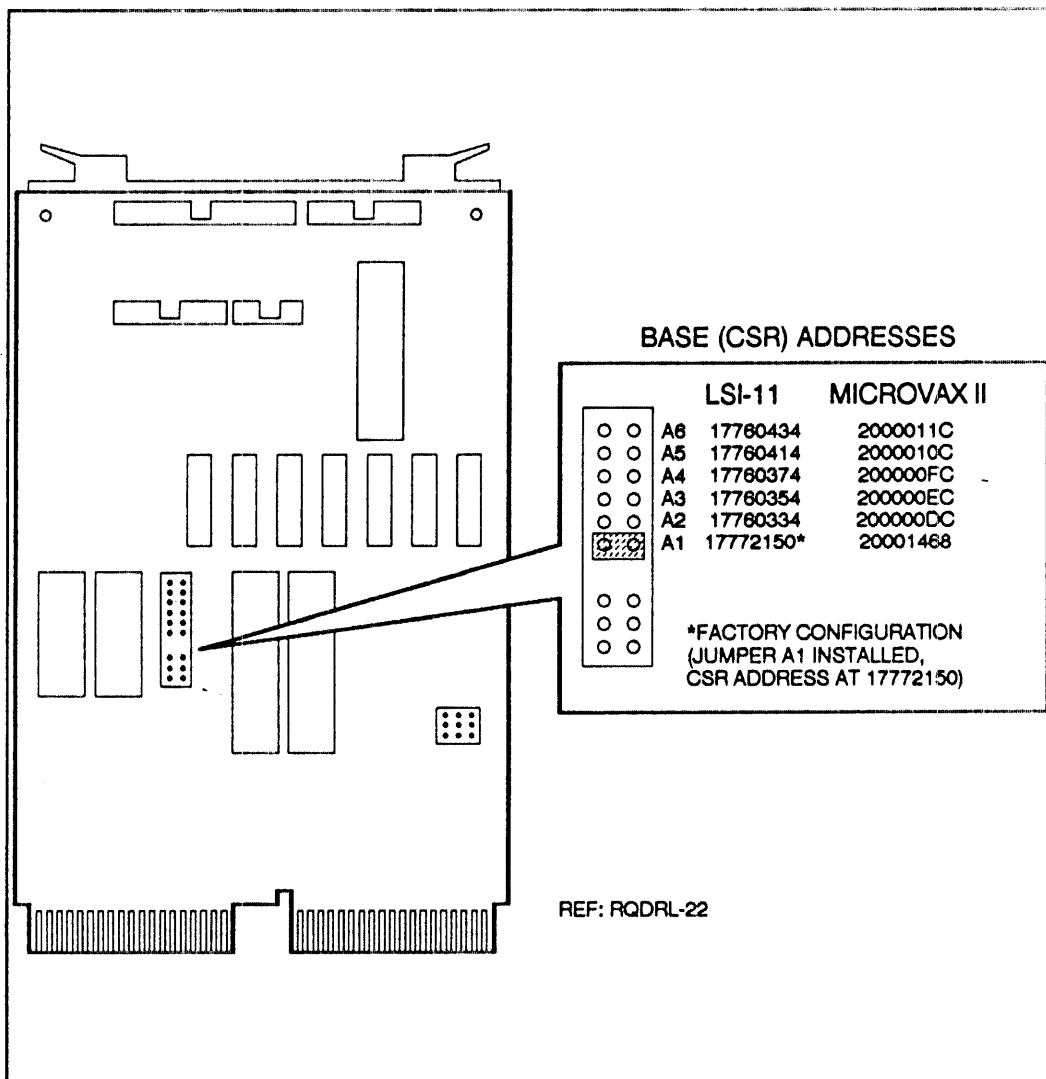


Figure 2-2:
Base (CSR) Address Configurations – Jumpers A1-A6

arbitration implements priority according to the priority levels set on the device hardware. When devices with equal priority generate an interrupt, the processor gives preference to the device which is electrically closest. A previous bus transaction must have been completed before another can be commenced.

Figure 2-4 shows the interrupt priority.

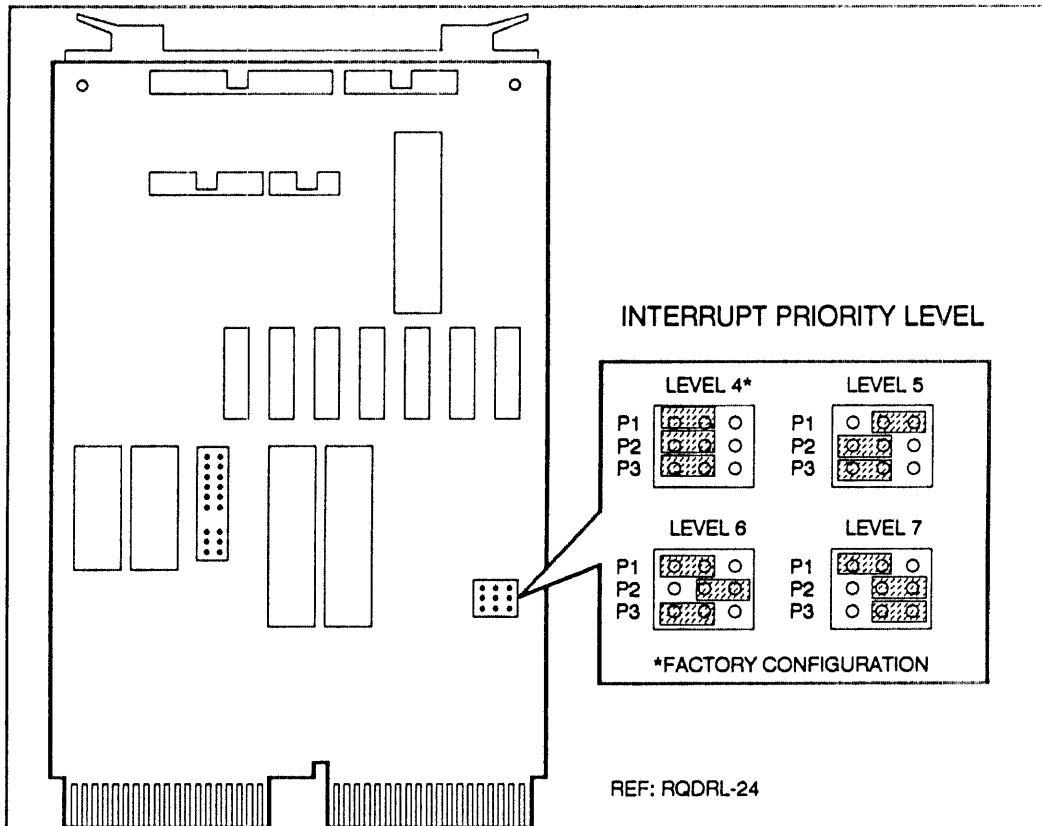


Figure 2-4:
Interrupt Level Configurations – Jumpers P1-P3

2.5 Optional Accessories

Optional accessories for the SDC-RQD11-RLL include:

- Drive Signal and Control Cable Set
- RS232 Maintenance Terminal Adapter (DLV11J Compatible)
- Front Panel Connector

2.5.1 Drive Signal & Control Cable Set

Both the drive signal and control cables consist of a flat or twisted pair cable joining a displacement-type flat cable socket at the controller end, and a displacement-type flat cable PC edge connector at the drive end. The drive controller cable and connector are 34-pin types, and the signal cable and sockets are 20-pin. Both cables have a maximum length of 20 feet. Drive/controller cabling is shown in Figure 2-5.

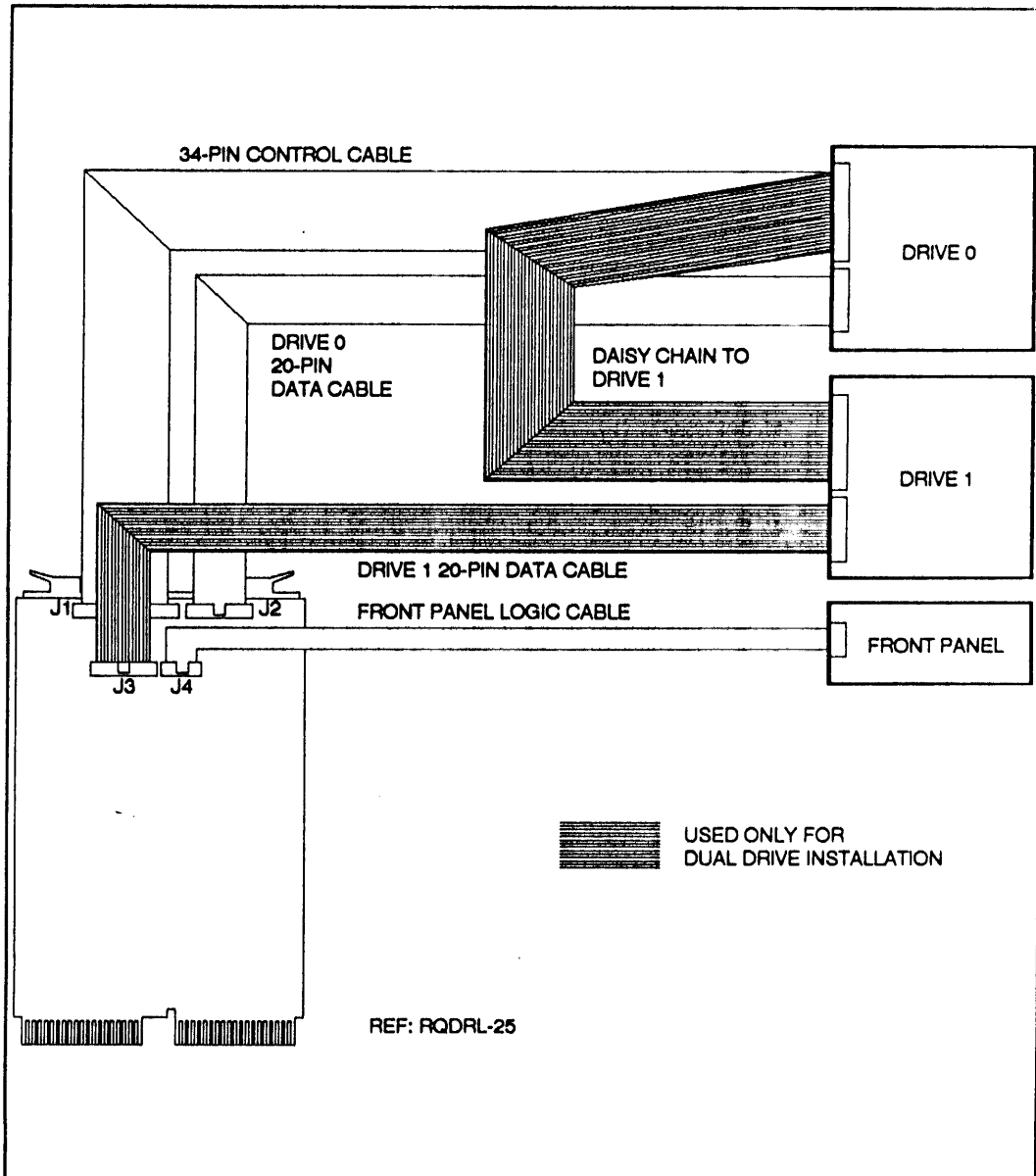


Figure 2-5:
Drive/controller Cabling

2.5.2 RS232 Maintenance Terminal Adaptor

The optional RS232 maintenance terminal adaptor allows the controller to be connected to an ASCII terminal. It consists of a 10-pin socket on a 10-conductor cable that is terminated with either a male DB25 connector (P/N 931010-00xx, where xx = 15 or 25 inches) or a female DB25 connector (P/N 931010-01xx).

The communication format is:

ASCII RS232 9600 Baud
8 Data Bits
1 Stop Bit
No parity.

Note that if a VT220 terminal is used the communication format must be set-up for space parity rather than no parity.

If normal disk access is attempted with this cable connected to a terminal, garbage will appear on the terminal due to the shared RS232 Output/Access Light Function. This is normal.

| 10-pin Connector Pins | DB25S Pins | Function |
|-----------------------|------------|--------------|
| 7 | 7 | RS232 Enable |
| 8 | 2 | RS232 Input |
| 3 | 3 | RS232 Output |
| 2,4,5,6,9 | 7 | Ground |

Table 2-3:

RS232 Maintenance Terminal Adapter (DLV11J Compatible)

2.5.3 Front Panel Connections

If required, a front panel can be connected to J4, the front panel/maintenance connector. The panel provides write protection and access light for two drives. See Table 2-4.

| PIN | FRONT PANEL FUNCTION |
|---|---|
| *7 | Write-protect switch input. Connecting this input to ground will write-protect drive 1. Has an on-board 1K ohm pull-up. |
| 8 | Write-protect switch input. Connecting this input to ground will write-protect drive 0. Has an on-board 10K ohm pull-up. |
| 3 | Access Light. When active, this will indicate drive 0 or drive 1 is being accessed. The levels are: Access: -5V through 1.5K ohm No Access: +3.5V @ 5 ma max. |
| 6 | Drive 0 Access Light. When active, this will indicate that drive 0 is being accessed. |
| 5 | Drive 1 Access Light. When active, this will indicate that drive 1 is being accessed. The levels for both Drive 0 and 1 Access Lights are: Access: 0V No Access: +3.5V |
| 2,4,9 | Ground. |
| *RS232 Maintenance Terminal Adapter. When pin 7 of J4 is grounded and a terminal is connected to this connector, WOMBAT will communicate to this terminal when WOMBAT is invoked. This connector is DLV11-J compatible. | |

Table 2-4:
Front Panel Connector

NOTE

If both Write Protect switches are closed, the controller assumes a front panel or maintenance cable is connected and ignores the switch functions — that is the drive will be "on-line" and "write-enabled." Keep switches open for WOMBAT.

2.6 On-Board Bootstrap

If the SDC-RQD11-RLL bootstrap is enabled, the following occurs:

1. On initialization location 773000 responds to the CPU fetch instruction with a BR .+2 (400) instruction to address 773002 where the CPU loops on a BR . instruction (777).
2. A "jump to zero" instruction (JMP @ #0 (137, 000)) is loaded into locations zero and two.
3. The controller program forces the CPU to start executing at location zero by changing the contents of location 773002 to CLR#PC (5007).
4. Bootstrap code is loaded into host memory at location 2000 and the rest of memory is cleared.
5. The controller changes location 2 from zero to 2000 changing the JMP @ #0 to JMP @ #2000, starting the execution of the loaded bootstrap program and the following message is typed:

BOOT Vx.x.

(where x.x is the firmware revision level.)

6. The controller boot program allows approximately 2 seconds for the operator to strike any key on the keyboard. If no key is struck the boot types:

Booting from DU0:

The bootstrap program reads in that device's boot block starting at location zero.

7. If any key is struck by the operator within two seconds the boot prompts with:

>

The operator may then key in a device DU, DL, DY, MS or W. Note that device W will invoke WOMBAT. By further specifying A, B, C, D, E or F after W, WOMBAT on controller A, B, C, D, E or F will be invoked. For example, WB will invoke WOMBAT on controller B.

The following is the syntax of the SDC-RQD11-RLL bootstrap procedure:

> [DEVICE] [Controller Number] [Unit Number]:

where [] = options below. If no options are specified defaults = A,0.

| DEVICE | [Controller Number] | [Unit Number] |
|--------|---------------------|---------------|
| DU | [A, B, C, D, E, F] | [0-7]: |
| DL | | [0-7]: |
| DY | | [0-1]: |
| MS | [A, B, C, D]: | |
| W | [A, B, C, D, E, F]: | |

Examples:

DU1 = Second drive on the first DU controller

DY1 = Second RX02 floppy disk drive

8. If any error occurs, a message from the following set is printed and the boot re-prompts for step five.

"? - Device must be DU, DY, DL or MS"

"Unit must be 0 - 7" - For DU or DL

"Unit must be 0 or 1" - For DY

"Controller must be A, B or C" - For DL

"Controller must be A, B, C or D" - For MS

"Boot failure" - Device unavailable (or not ready if DL)

The selected controller is commanded to read the boot block (block zero) from the specified device into the host memory and then waits to be initialized. The host computer commences execution of the instructions in the boot block. The devices "assumed" by the SDC-RQD11-RLL bootstrap are detailed in Table 2-5.

| DEVICE | ADDRESS |
|----------|----------|
| CONSOLE: | 17777560 |
| DU: | 17772150 |
| DUA: | 17772150 |
| DUB: | 17760334 |
| DUC: | 17760340 |
| DUD: | 17760374 |
| DUE: | 17760414 |
| DUF: | 17760434 |
| DL: | 17774400 |
| DY: | 17777170 |
| MS: | 17772520 |
| MSA: | 17772520 |
| MSB: | 17772524 |
| MSC: | 17772530 |
| MSD: | 17772534 |

*Table 2-5:
CSR Addresses for Bootstrap*

3. Programming

3.1 Overview of MSCP

Mass storage control protocol (MSCP) is the message-oriented set of rules by which the SDC-RQD11-RLL controller module communicates with the host system. This protocol allows the host to simply send message requests for reads or writes to the controller and receive response messages back from the controller. The host does not concern itself with details such as device type, media geometry, media format, or error recovery.

All software and hardware functions are partitioned into two independent layers (host and controller), where changes can be made within one layer without affecting the other. Within this framework, commands and data are transmitted from one layer to the other, through the controller, in the form of message packets.

In the host layer, the computer runs users' applications programs that make demands on the mass storage (disk) medium. The controller layer's various functions ensure that the host layer is able to read or write data, without error, at its own speed, and when it wishes. The disk layer receives the data, stores it as long as necessary, and makes it available to the controller on command.

The host layer uses two layers of software to accomplish input/output operations. The first sub-layer includes a mass-storage class driver

which constructs the message packets in order to perform I/O functions, such as reading and writing, and on the same level, a diagnostics and utilities class driver which constructs message packets in the diagnostics and utilities protocol. The other sub-layer is the controller driver which passes the message packets along the bus between the host and controller.

The controller layer includes routines that receive messages from or transmit messages to the host. Its other functions concern the disk, which include controlling head motion, accepting commands from the controller, reporting status to the controller, and reading and writing.

The mass-storage class driver handles all message exchange between the operating system and any mass-storage device of a specific class, which means that any size disk may be fully accessed without software modification. The disk drive itself contains a parameter table of all of its own characteristics, such as geometry and retry counts for error handling. At system startup, this information is passed to the controller so that it may manage operation of that particular disk configuration.

In addition to relieving the host-resident driver of disk-specific data, the controller and disk together provide the host with "clean" data. This implies data for which all necessary error detection, correction, and recovery have already been done. The disk drive handles some positioner errors entirely by itself and performs certain error-recovery operations under direction of the controller.

3.2 Controller Communications

The host designates an area of memory to be used as a communications area. This area is made up of two sections:

1. The header area containing interrupt identification words.
2. A variable-length section containing the response (receive) and command (send) rings, organized into ring buffers.

The following diagram shows the format of the memory communications area.

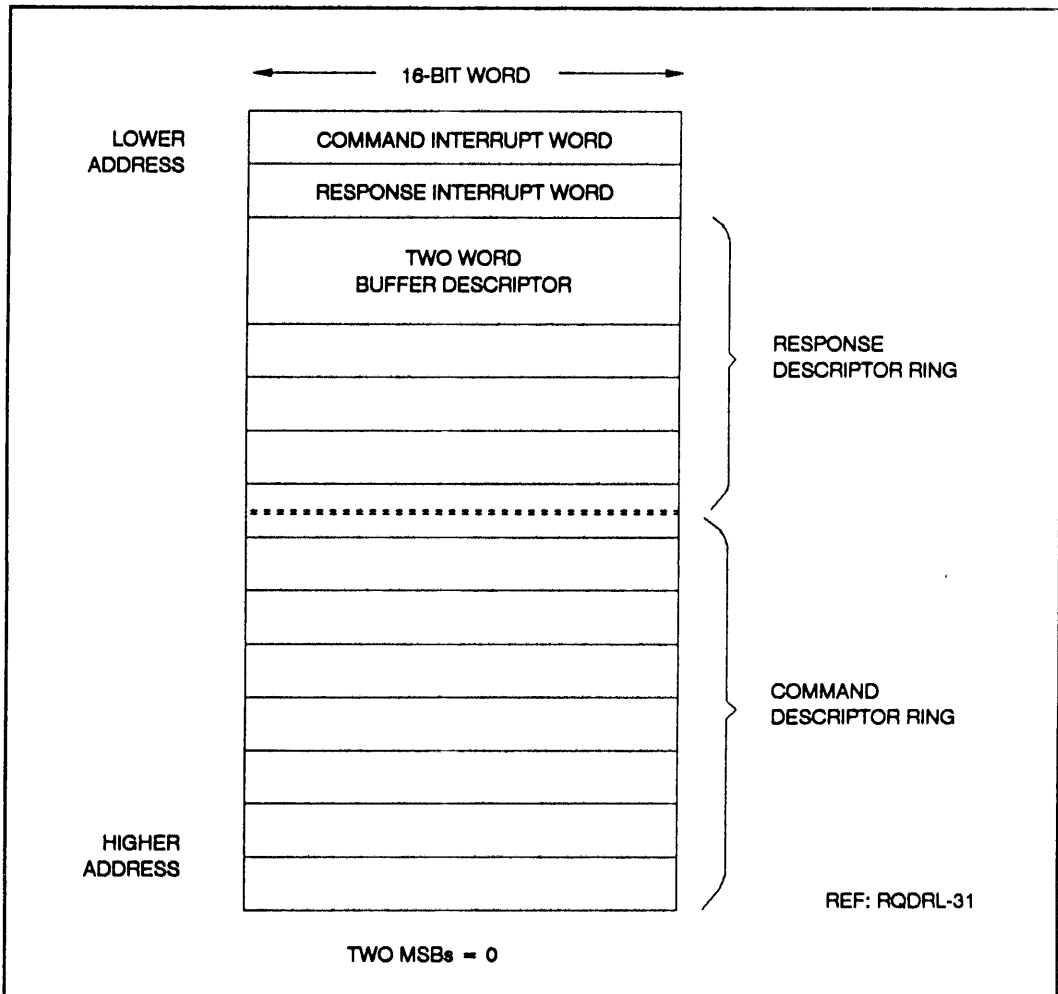


Figure 3-1:
Memory Communications Format

3.2.1 Command and Response Rings

Command and response rings are each organized into a ring of 32-bit descriptors with a format illustrated in Figure 3-2. The length of each ring is determined by the relative speeds with which the host and controller generate and process messages. The host sets the ring lengths at initialization time.

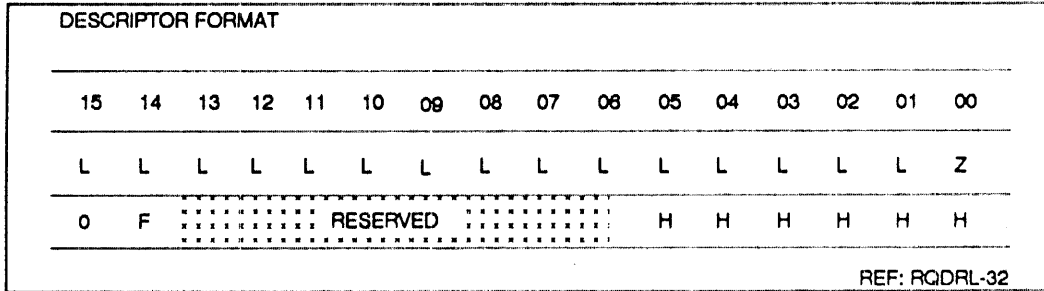


Figure 3-2:
Descriptor Format

| Code | Description |
|------|--|
| Z | Is zero, as envelope address (text + 0) is word-aligned. The controller will always assume that Bit 00 is set to zero. |
| L | Low-order envelope address. |
| H | High-order envelope address. |
| F | Flag bit. |

When the controller returns ownership to the host it sets $F=1$ to indicate that it has completed action on the descriptor.

When the controller acquires ownership of a descriptor from the host, $F=1$ indicates that the host is requesting a ring transition interrupt. If $F=0$, the host is not requesting a ring transition interrupt. The interrupt will occur only if this descriptor causes a ring transition and if transition interrupts were enabled during initialization.

The controller always sets $F=1$ when returning a descriptor to the host, so if a host wishes to override ring transition interrupts it must always clear F when passing ownership of a descriptor to the controller.

- O Ownership bit. Set to 0 if owned by the host or 1 if owned by the controller. Interlocks the descriptor against premature access by either party.

3.2.2 Message Packets

The command or response descriptor points to word ($\text{text}+0$) of a 16-bit word-aligned message envelope formatted as follows:

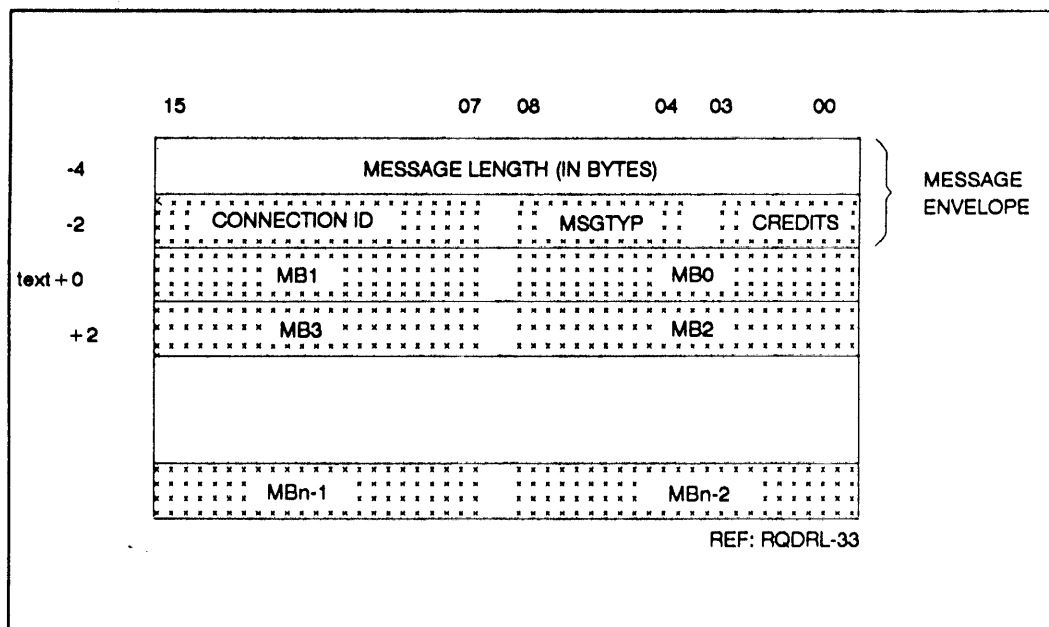


Figure 3-3:
16-bit Word-Aligned Message Envelope Format

| Word | Envelope Contents |
|------|-------------------|
|------|-------------------|

- | | |
|---|---------------------------|
| 0 | Message length, in bytes. |
|---|---------------------------|

For commands, this length is equal to the size of the command (in bytes), beginning with $[\text{text}+0]$.

3.3 Message Transmission

3.3.1 Command Transmission

When the ownership bit (O) of a command ring descriptor is equal to 1, it means that the host has filled the descriptor and is releasing it to the controller. When the ownership bit (O) resets to zero, it means that the controller has emptied the command ring descriptor and is returning ownership of the descriptor to the host.

To ensure that the controller sees every command, the host must read the IP register whenever it inserts a command in the command ring. This forces the controller to poll the command if it was not already accessing the command ring.

3.3.2 Response Transmission

When the ownership bit (O) of a response ring descriptor is equal to zero, it means that the controller has filled the descriptor and is releasing it to the host. When the ownership bit (O) sets to 1 it means that the host has emptied the response ring descriptor and is returning ownership of the descriptor to the controller. Just as the controller must poll for commands, so must the host poll for responses.

3.3.3 Interrupts

The transmission of a message will result in a host interrupt from the controller under the following circumstances.

1. During the initialization process (open a 'connection').
2. When the command ring buffer transitions from 'full' to 'not full'. This interrupt means that the host may place another command in the command ring.
3. When the response ring buffer transitions from 'empty' to 'not empty'. This interrupt means that there is a response for the host to process.

4. When a fatal controller error is detected and an interrupt can be generated. These are:

- Failure to become Q-bus master for data transfer
- Failure to become Q-bus master for interrupt
- Failure to access I/O page registers or communication area
- Q-bus parity error detected.

3.4 Data Transmission

In the command ring, the descriptor points to a command packet. Within the command packet is a buffer descriptor which contains a pointer and a byte or word count. The buffer descriptor points to the data buffer which holds data transfers. The data is moved by the controller into or out of the buffer as DMA transfers to/from Q-bus addresses.

3.5 Initialization

The purpose of initialization is to identify the parameters of the host-resident communications region to the controller, provide a confidence check of controller integrity, and bring the controller on-line to the host.

3.5.1 Initialization Process

This paragraph describes the activity within the SA register during an initialization process. This is dependent on whether SA is being read or written.

By moving 4000 into IP, the controller initializes and passes back the 'step' response in SA. Then, the initialization parameters are written into SA. There are 4 words of initialization, and the controller must reflect each step by the appropriate step response, which is also returned in SA.

3.5.2 Initialization Parameters

| Word | Contents |
|------|--|
| 0 | <p>Command and Response ring sizes, interrupt enable and vector.</p> <p>The host writes into SA the lengths of the rings, whether interrupts are to be armed, and if so, the address of the interrupt vector. The controller then runs a complete internal integrity check and signals either success or failure.</p> |
| 1 | <p>Low order address of communications area, ie., ring buffer address.</p> <p>The host reads an echo of the ring lengths from SA, and then writes into SA the low-order portion of the ring base address.</p> |
| 2 | <p>High order address of communications area, bits 0-14.</p> <p>The interrupt vector address and the master interrupt arming signal are echoed in SA. The host then writes the high order portion of the ring base address to SA along with a signal that conditionally triggers an immediate test of the polling functions of the controller.</p> |
| 3 | <p>Burst transfer control, last failure flag, and the 'GO' bit.</p> <p>The controller tests the ability of the Q-bus to perform DMA transfers. If successful, the controller zeros the entire communications area, and then signals the host that initialization is complete.</p> |

3.6 Registers

The programmable registers contained on the SDC-RQD11-RLL are the Initialize and Poll register (IP) and the Status and Address register (SA).

3.6.1 Initialize and Poll Register (IP)

The host begins the initialization sequence by either issuing a bus initialize or by using the IP initialize operation. Any write to that address will cause an initialization of the controller. When read while the controller is operating, it causes the controller to initiate polling. While DEC's controller always performs an initialization when the IP register is written to, the SDC-RQD11-RLL responds to the following initialization words:

| CPU | Word (octal) | Function |
|---------------------|---------------|--|
| LSI-11 only | *250 | Call WOMBAT |
| LSI-11 only | 252 | Read block zero into host computer memory at location zero (simple boot procedure) |
| LSI-11 and MicroVAX | Anything else | Initialization |

*See Section 4.3 for invoking WOMBAT on MicroVAX CPUs.

*Table 3-1:
Initialization Operations*

3.6.2 Status and Address Register (SA)

The SA register consists of a set of two registers, the SA read register and the SA write register.

When read by the host during initialization, it communicates data and error information relating to the initialization process. When written by the host during initialization, it communicates certain host-specific parameters to the controller. When read by the host during normal operation, it communicates status information including fatal errors detected by the controller.

3.7 MSCP Commands

The following commands are supported by the SDC-RQD11-RLL controller.

| COMMAND | FUNCTION |
|---------------------------|---|
| Access | Reads data from the specified unit. |
| Abort | Guarantees that referenced MSCP command will complete within the controller time-out period. |
| Available | If specified unit is on-line, returns it to the unit-available state. If specified unit is currently in the unit-available state, this command has no affect. |
| Compare Host Data | Reads data from the disk and compares it with the data in the host buffer. |
| Erase | Writes zeros to the specified logical blocks on the unit. (No data is accessed from the host). |
| Get Command Status | Reports the status of a specified command with a number that reflects the command's progress. |
| Get Unit Status | Reports on the status of a specified unit. |

| | |
|----------------------------|--|
| On Line | Places the specified unit on line, if possible. |
| Read | Reads data starting from the specified logical block on the disk, into host memory. |
| Set Controller Char | Sets host-settable controller characteristics. |
| Set Unit Char | Sets host-settable unit characteristics. |
| Write | Writes data starting at the specified logical block on the disk, from the host memory. |

3.8 Error Handling

High data integrity is achieved by the controller through error code and correction (ECC) and transparent bad block replacement. A read operation is performed up to 10 times before a hardware error is reported to the operating system.

MSCP STATUS CODE MESSAGES

| | |
|-------------------------|--|
| Command Aborted | The current command was aborted before it could be completed normally. |
| Compare Error | While performing a Compare command, a discrepancy was found while comparing the disk data to the host data. |
| Controller Error | The SDC-RQD11-RLL controller detected an internal error, but is able to continue processing its outstanding commands. |
| Data Error | Data could not be read or written due to CRC errors, "Header Not Found", or due to a sector being read whose forced error bit was set. |
| Drive Error | A drive-related error was detected (such as a seek failure). |

| | |
|---------------------------------|---|
| Media Format Error | Indicates that the media mounted on the unit was incorrectly formatted. |
| Host Buffer Access Error | Reports bus time-outs and parity errors during data transfers. (Applies only to the data portion of an MSCP command). |
| Invalid Command | The SDC-RQD11-RLL controller found some field in the command to be in error. |
| Success | The command was successfully completed. |
| Unit Available | The SDC-RQD11-RLL controller is not on line, but it can accept an On Line command from the host. |
| Unit Off-line | The SDC-RQD11-RLL controller is not on line, and it cannot be brought on line. |
| Write Protected | A Write or Erase command was attempted to a unit that is logically write-protected. |

3.9 Fatal Controller Error

If a fatal error is detected when the controller is initialized, the error LED is lit, and the fatal error status set in the SA register

| Error Codes (octal) | Description |
|--------------------------------|----------------------|
| 100004 | RAM test failure |
| 100005 | ROM checksum failure |
| 100011 | No drive |
| 100100 | Disk unformatted |
| 100101 | Disk unstructured |
| 100103 | No RCT table |
| 100103 | No FCE table |

A full description may be found under Section 4.6 : Wombat Error messages.

4. WOMBAT Utilities

4.1 Introduction to WOMBAT Utilities

WOMBAT utilities provide a controller resident means of formatting, testing and maintaining the drive and controller sub-system.

Interactive communication with WOMBAT may be achieved by connecting a 9600-baud terminal to the SDC-RQD11-RLL as described in Chapter 2.

When no terminal is connected directly to the SDC-RQD11-RLL, the WOMBAT communication program will automatically search for and use the console terminal at address 177560 on the Q-bus.

4.2 Outalk

When the WOMBAT code is deposited into the SDC-RQD11-RLL base address (250 octal for LSI CPUs and AC hex for MicroVAX CPUs shown above), WOMBAT outloads a communication program from its program ROM into its host's memory. When this program executes, it manages communication between the console and WOMBAT. Communication between WOMBAT and a console terminal enables WOMBAT diagnostics and formatting capability on systems that have the console terminal as an integral part of the CPU (e.g., KDF11-B, KDJ11-B, MicroVAX).

NOTE

The second and third base address conform to the MicroVMS software requirements.

4.3 Starting Up Wombat

WOMBAT is invoked by depositing a special data pattern into the SDC-RQD11-RLL control and status register. WOMBAT may be started by depositing a WOMBAT code into the IP register via console ODT. WOMBAT can be started from boot code. At the boot prompt (>) enter:

```

W (CSR 17772150)
WA (CSR 17772150)
WB (CSR 17760334)
WC (CSR 17760340)

```

The procedures for invoking WOMBAT on LSI-11, and the MicroVAX II are given in the following paragraphs. WOMBAT can be stopped by simply re-booting the system.

In each case input the appropriate CSR address for the controller to be accessed where indicated by 'CSR*'. Table 4-1 lists the possible CSR addresses for the SDC-RQD11-RLL.

| LSI11 | MicroVAX II |
|----------|-------------|
| 17772150 | 20001468 |
| 17760334 | 200000DC |
| 17760354 | 200000EC |
| 17760374 | 200000FC |
| 17760414 | 2000010C |
| 1776434 | 2000011C |

*Table 4-1:
CSR Addresses*

WOMBAT on LSI processors is invoked by:

| KEYBOARD ENTRY | SYSTEM RESPONSE | KEYBOARD ENTRY | COMMENTS |
|----------------|-----------------|----------------|---|
| CSR* | 000000 | 250 <RET > | ask WOMBAT to load the communication program into memory. |
| R7/ | XXXXXX | 2000 <RET > | set up program start address |
| RS/ | 000000 | 340 <RET > | set PSW to block interrupts |
| P <RET > | | | start program without the bus reset that a micro ODT "G" would cause. |

WOMBAT on a MicroVAX II processor is invoked by:

| SYSTEM PROMPT | KEYBOARD ENTRY | COMMENTS |
|---------------|------------------------------|---|
| > > > | D/P/W 20001F40 20 <RET > | enable Q-bus access to memory |
| > > > | D/L 20088008 80000002 <RET > | set up the appropriate Q-bus map entry |
| > > > | D/W CSR* AC <RET > | ask WOMBAT to load the communications program into memory |
| > > > | S 400 <RET > | start the program |

When P (LSI-11) or S 400 (MicroVAX) is entered WOMBAT should start and appear on the console terminal.

When WOMBAT is invoked it will immediately try to determine how many disk drives are connected to the SDC-RQD11-RLL. The following will be displayed on the terminal:

```
**WOMBAT Version 5.X**  
DRIVE 0: [READY/NO DRIVE]  
DRIVE 1: [READY/NO DRIVE]  
SELECT DRIVE [0-1]
```

Upon selecting a drive, WOMBAT will immediately try to read the disk and assess if it has a valid structure. The structure incorporates such factors as the number of cylinders, number of heads, and other details relating to the disk. This information is always written on the first two sectors of the first track of the disk, which makes this area and the remainder of track zero permanently unavailable to the user. Once the disk is structured, WOMBAT can access this information without having to know anything about the disk. When a disk does not have a valid structure a warning message is displayed:

```
* Warning - disk is not formatted  
* Warning - disk has no valid structure
```

You cannot perform any function with WOMBAT until the disk has first been structured (refer section 4.4.1). A previously structured disk will automatically return the Master Menu when WOMBAT is invoked.

4.4 Setting Up The Disk Structure

When attaching a disk drive to the controller, go through the process of setting up the disk structure, and then write that structure on to the disk. Do this by:

1. Creating the disk structure.
2. Formatting the disk (the structure is still in the SDC-RQD11-RLL memory).
3. Writing the disk structure.
4. Entering Manufacturer's Defect Map
5. Testing the disk drive, if you are unsure how reliable it is. At this stage you may still write to the disk as the structure is still in the SDC-RQD11-RLL memory.
6. If the disk drive is not suspect and does not have errors on blocks zero or one, write the disk structure to it. If the disk has errors on blocks zero or one, you cannot attach and use it with the SDC-RQD11-RLL controller.
7. If necessary, do bad block management.

When you have set up the disk structure, then formatted and written the structure to disk, you may let the host operating system use the disk.

The instructions for these procedures and the other available options, are described in the next paragraphs. For consistency and clarity, each option is described within its Menu set.

4.5 Wombat Menu Options

You communicate with WOMBAT by selecting the required option from the master menu set which is given below:

```
*** Master Menu ***
  1 Structure Disk
  2 Test Disk
  3 Manage Bad Blocks
  4 Initialize Controller
  5 Position Head on Shipping Zone
```

Select an option by typing 'option number' followed by a carriage return (RETURN). Options 4 and 5 are single function options, while accessing options 1, 2 or 3 will provide you with a further sub-menu of options as detailed in the following:

```
*** Option 1 : Disk Structure Menu ***
  1 Create Disk Structure
  2 Format Disk
  3 Write Disk Structure
  4 Update HDR blocks
  5 Display Disk Structure
  6 Set up Unit Structure
```

***** Option 2 : Disk Test Menu *****
(! means all data on disk destroyed)

- 1 Read all Disk (preserves all data)
- 2 ! Write Disk !
- 3 ! Pattern Test !
- 4 ! Random Writes !
- 5 ECC Validation (preserves all data)
- 6 Display Error Statistics
- 7 Zero Error Statistics

***** Option 3 : Bad Block Management Menu *****

- 1 Manually Replace Bad Block
- 2 Automatically Replace Bad Block from Error Statistics
- 3 Manually Enter Manufacturer's Defect Map
- 4 Display Replaced Bad Blocks

***** Option 4 : Initialize Controller *****
Warning - Initializing will clear any error statistics.
Is this what you want to do [Y/N]?

***** Option 5 : Position Head on Shipping Zone *****
Disk Heads now on Shipping zone!
This disk drive may now be safely removed.

To return to the Master Menu while in a sub-menu, enter RETURN at an option prompt.

4.5.1 Disk Structure Menu

Option 1 - Create Disk structure

When you attach a new disk drive to the SDC-RQD11-RLL controller, this option **MUST** be the **FIRST** selected as you cannot do anything else until the disk structure is specified.

WOMBAT has a completely 'soft' disk structure. That is, it assumes nothing about the various disk parameters: cylinders, heads, and other pertinent factors. This option allows these parameters to be specified.

WOMBAT first displays its name and version. This is used as an identifier when the disk structure is read to ensure that the various routines accessing the disk agree about the structure format. The warning message simply highlights the fact that the disk is not structured.

If a disk drive has been previously structured, WOMBAT displays-



Disk Characteristics
Created by: WOMBAT Version: 4.5

You may override any characteristic with different data, or press RETURN to accept the current parameter and display the next line.

If you incorrectly type in any one of the 10 parameters, simply press the RETURN key enough times to bypass the remaining parameter fields, until the 'Disk Structure Menu' is returned. Select this option again and enter the correct data.

Example of a Disk's Structure

The following example is representative of a particular 20 Mbyte disk drive. You would enter the correct figures relevant to your disk. All numeric values are decimals unless otherwise stated. Do not enter the decimal point to signify a decimal number as this is automatically returned by WOMBAT.

All figures shown in the right-hand column are entered at the keyboard.

| DESCRIPTION | DEFAULTS | EXAMPLE ENTRIES |
|--|-----------|-----------------|
| Size, Cylinders: | [0.]? | 306 |
| Heads: | [0.]? | 8 |
| Sectors per Track:[26] | [26.]? | * <RET> |
| Replacement Cylinders: | [0.]? | 6 |
| Head Step Rate: | [15.]? | 0 |
| Command Queue Size:[1-32] | [8.]? | * <RET> |
| Seek Optimization: 0-None 1-Nearest 2-Elevator 3-Forward | | |
| Optimization Strategy:[0-3] | [0.]? | 1 |
| Fairness Count:[1-255] | [25.]? | * <RET> |
| Shipping Zone Cylinder: | [340.]? | * <RET> |
| Media Type:(AAAnn) | [WCCxx.]? | WCC31 |
| Removable Media | [Y/N] | N |
| Serial Number:(Octal) | [0.]? | * <RET> |
| Unit Number | [0.] | * <RET> |

* <RET> retains the default value.

Use 'Set up unit structure' option for multiple units on drive 1. Once units are defined, the display for Multiple Units Defined is described in the 'Set up unit structure' display.

SYSTEM READBACK MESSAGE:

USR area: 43038 blocks

'RETRN' to continue: (Returns the Disk Structure Menu)

Field Descriptions

The drive's data sheet is required for reference as WOMBAT needs to know some of the specifications. It is important that the details are accurately transcribed to ensure that WOMBAT runs effectively. Please realize that WOMBAT trusts you. Entering nonsensical data will result in chaos. Press the RETURN key after each entry and the next prompt will be displayed.

| | |
|------------------------------|---|
| Size, Cylinders | The total (decimal) number of cylinders on the disk. You will find this information on the drive data sheet. |
| Heads | The number of read/write data heads (other than servo heads) that the drive has, as stated on the data sheet. |
| Sectors per Track | The number of 512-byte sectors (26). |
| Replacement Cylinders | <p>This allocates room for the bad block replacement table. Allow at least 6 cylinders. When a structure is created on a formatted disk, the allocation is made as follows:</p> <ul style="list-style-type: none">◦ 2 tracks for RCT (Replacement Control Table)◦ 1 track for WRK (Controller work area)◦ 1 track for FCE (Forced error table)◦ remainder for BAD (Bad block replacement area) <p>If you do not allow enough space serious problems may occur.</p> |
| Head Step Rate | Specify the rate at which the disk heads can step. Zero is default, and is used for all disks with buffered stepper motor or voice coil head positioners. |

| Rate | Time between Stepping Pulses |
|------|------------------------------|
| 0 | 24us |
| 1 | .33ms |
| 2 | .67ms |
| 3 | 1.0ms |
| 4 | 1.3ms |
| 5 | 1.7ms |
| 6 | 2.0ms |
| 7 | 2.3ms |
| 8 | 2.7ms |
| 9 | 3.0ms |
| 10 | 3.3ms |
| 11 | 3.7ms |
| 12 | 4.0ms |
| 13 | 4.3ms |
| 14 | 2.1us |
| 15 | 11us |

WOMBAT defaults to rate 15.

**Command Queue
Size**

The MSCP protocol allows the controller to stack a number of commands; this parameter allows you to specify the size of the command stack.

Please realize that the larger the stack, the more overhead the disk controller incurs when it scans it; also that some operating systems (RSX-11M-Plus 2.1B is a good example) have a maximum limit for the size of the stack. The default size (8.) is a good compromise, and acceptable to most operating systems.

**Optimization
Strategy**

The seek optimization strategy can be either:

- 0-None
- 1-Nearest
- 2-Elevator
- 3-Forward

This is a user preference feature. WOMBAT displays '? Invalid' if the number entered is out of range, and repeats the prompt.

Explanation:

None No optimization done. First request found executed.

Warning : This may not be the next sequential request.

Near Nearest cylinder strategy selects the request that is closest to the current cylinder.

Elevator This processes requests like an elevator - as it moves in one direction along the disk until it reaches the last request in that direction. This means that "Elevator" favors the center of the disk, as it passes it twice as often as the periphery.

Forward This processes requests from the lowest cylinder number to the highest; like "Elevator" except in only one direction. This is the generally recommended strategy for most purposes, and will yield an approximately 2:1 improvement in apparent performance in random access applications.

Note that optimization is only effective if the host operating system supports multiple accesses. RT-11, TSX-Plus, and RSX-11M

normally do not without provision of special device handlers.

Fairness Count

The fairness count relates to disk commands. A reasonable count for normal use would be around 25. If the number entered is not within the range 1 - 255, WOMBAT displays the message 'Number too large', and the cursor moves to the next line. Enter a valid number.

This count determines the number of times an I/O request will be passed over by seek optimization before it is executed. Every time a request is passed over its fairness count is decremented. When that count reaches zero that request will be selected, no matter what optimization strategy is in effect.

This count has no effect if no optimization is selected.

**Shipping Zone
Cylinder**

WOMBAT is setup to use cylinder 340, being standard for a number of disk drives. Refer to the Appendix to ensure that this is appropriate for your type of drive.

Media Type

This field allows the media type to be specified. The MSCP protocol returns - as part of unit status when a "Get Unit Status" command is issued - a 5 character media type.

As a default, WOMBAT sets up "WCCnnn", where nnn is the size of the drive in megabytes. To change this, enter 1 to 3 alphabetic characters and 2 digits, e.g. RD52, to emulate DEC's 31MB Winchester.

This field is displayed by some operating systems when you inquire about the type of drive. For example, RSX-11M-PLUS responds

to a "DEV DU:" command with: "DUO:
Public Mounted Loaded Label =
RSX11MPBL15 Type = WCC31."

Serial Number

The MSCP protocol returns - as part of its response when an "on- line" command is issued - a 32-bit volume serial number. WOMBAT defaults this field to zero.

To change this, enter, in octal, the desired serial number.

This field is used, for example, by RSX-11M-PLUS, when you initialize a disk with the "INI DU:" command. It sets up the volume serial number.

Unit Number

The unit number will default to the drive number of the selected drive.

WOMBAT will then compute and report the user area on the disk. The user area (USR area) is defined as the size, in blocks, of the user area on the disk, as reported to the host operating system.

The user area is the ONLY portion of the disk that the host operating system can see.

Option 2 - Format Disk

The disk structure is now set up in the SDC-RQD11-RLL controller's local memory. Before the structure can be written out to disk, the disk must be formatted.

Formatting destroys ALL INFORMATION on the disk, and this includes previous bad block allocations recorded in the Replacement Control Table. Take note that if a disk is restructured and not reformatted, then previously allocated bad block numbers could create an error when the structure is written to disk. The error message 'Disk Error Writing Disk Structure at NNNNN' would be displayed. NNNNN represents a bad block number allocated prior to the last formatting procedure.

When you select this option WOMBAT displays:

***** Warning: all information on this disk will be destroyed, including any bad block mapping. Is this what you want to do [Y/N]?**

Ensure that you do in fact want to format the disk and enter Y. If you enter N, the Disk Structure menu is returned.

While the disk is being formatted WOMBAT displays the mapped cylinder counter which commences from zero and increments by 64, thus showing you when the formatting operation is nearing completion. The final cylinder number (the parameter you entered when structuring the disk) is not displayed. When formatting is complete, the message '*Disk Formatted*' is displayed and the disk structure is written. The Disk Structure menu is then returned.

Option 3 - Write Disk Structure

When you set up the disk structure, it is only recorded in the controller's local memory. This option writes the structure on the disk, where it is read by either the controller program or WOMBAT, to set up the disk parameters. Once this procedure is complete the host operating system may use the disk. If an override structure is written to disk and the disk has not been reformatted, an error could occur if there were blocks previously marked as bad (refer "BAD BLOCK MANAGEMENT MENU") and the following message will be displayed:

***** Warning - replaced bad blocks exist, this option will clear replacement table but leave blocks marked as bad on disk!
Is this what you want to do [Y/N]?**

No information is displayed during the write process and when finished, the Disk Structure menu is returned.

Option 4 - Update HDR Blocks

This option allows you to just update the Disk Structure as recorded in the HDR blocks. If you, in the "Create Disk Structure" option, have ONLY altered any of the following parameters:

- Command Queue Size
- Optimization Strategy
- Fairness Count
- Shipping Zone Cylinder
- Media Type
- Serial Number
- Unit Number
- Set Up Unit Structure (Option 6),

you may update the HDR blocks to include these changed parameters without destroying any data on the disk. A good example of this is if you are testing various disk optimization strategies.

WARNING

Do not alter any other parameter and try to use this option; you must "Write Disk Structure" to update all the disk structures in this case.

No information is displayed during the update process, and when finished, the Disk Structure Menu is returned.

Option 5 - Display Disk Structure

This is an optional sequence which may be used at any time to review parameters entered during a 'Create Disk Structure' procedure.

Option 6 - Set Up Unit Structure

Unit parameters entered during the 'Create Disk Structure' procedure can be redefined using this option. For example, when the following is displayed,

| Unit | Start Block | Size |
|------|-------------|-------|
| 1 | 26 | 60480 |
| 3 | 60498 | 33624 |

Do you wish to redefine the unit structure [Y/N]?

simply enter a 'Y' if any of the parameters need to be changed. WOMBAT will then prompt for the number of the unit to be redefined, and once entered, the respective unit parameters will be displayed. At this point, any changes made will overwrite previous entries. If there are no changes necessary, enter an 'N,' and the Disk Structure Menu will be returned.

If the units are not defined, the 'No units defined' message will be displayed.

4.5.2 Test Disk Menu

A disk can be tested after it has been formatted and before the structure is written to it. Testing does not overwrite the HDR or RCT blocks.

All tests continue indefinitely until aborted by one of the following methods:

1. If an ASCII terminal is attached to the controller, press BREAK.
2. If WOMBAT is running from the Console terminal, type CTRL/C. To abort, multiple CTRL/C may have to be entered rapidly.

When a test is aborted the Test Disk menu options are returned. If tests are run from an ASCII terminal attached to the controller, beware of system activity on the host computer as Q-bus initializations will cause the disk controller firmware to re-initialize and so leave WOMBAT.

All tests give 10 retries on an error, reporting every error by displaying the block number and an error code. The codes are defined as follows:

| |
|---|
| <pre> BB Bad Block; block marked as bad. ID Id error; sector header cannot be found on disk. ECC Cyclic redundancy error; miscalculation of the 16-bit cyclic redundancy word on a write or read verification sequence. </pre> |
|---|

Option 1 -Read All Disk

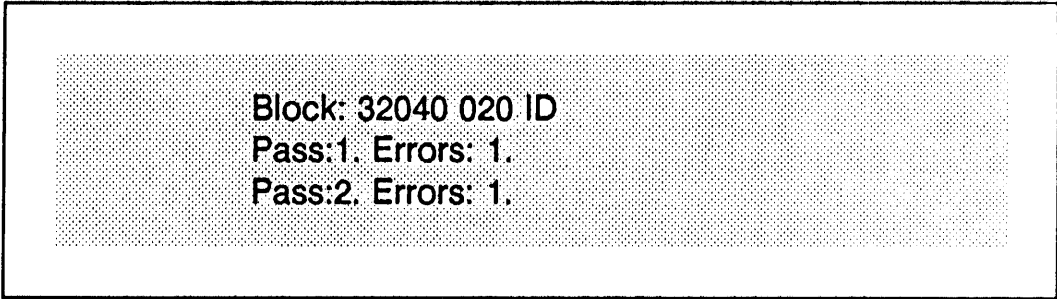
This test reports any read errors. Successful operation will be reported in the following format:

| |
|--|
| <pre> Pass: 1. Errors: 0. Pass: 2. Errors: 0. </pre> |
|--|

This function does not destroy any information.

Option 2 - Write Disk

This test reports any write errors while writing a test pattern to the whole disk. **ALL INFORMATION** on the disk, excepting HDR and RCT blocks, is **DESTROYED**. Errors are displayed in the standard format:



```
Block: 32040 020 ID  
Pass:1. Errors: 1.  
Pass:2. Errors: 1.
```

Displayed error count is cumulative until the test is terminated.

Option 3 - Pattern Test

This test writes a worst case pattern to each block along with the block number. It does one write and 10 read and compare passes, where it checks that it is reading the right block and that the data pattern is correct. This test reports any errors in the standard format as shown above.

Option 4 - Random Writes

As for Pattern test on USR area of disk, then performs 5000 random writes, reporting every 1000 writes for timing purposes. Finally it does a check of all data on the disk again. This test reports:



```
Initializing Disk ...  
#5000 Random Writes  
1000.  
2000.
```

Option 5 - ECC Validation

Enables ECC logic to be checked. This writes blocks with both correctable and uncorrectable ECC errors, thus verifying the ECC operation. This test is non-destructive of data on the disk.

Option 6 - Display Error Statistics

Displays the error statistics gathered by any of the above disk testing options in the following format:

```
**** Error Statistics ****  
BlockNumber (of errors)  
32040 1.  
Blocks in error:1.
```

Option 7 - Zero Error Statistics

Zeroes the error statistics table & redisplay Test Menu options.

4.5.3 Bad Block Management Menu

Bad blocks are replaced by marking them as bad on the disk (by flagging them as "BAD" in the sector header), and recording the block number in the Replacement Control Table, pointing at a replacement block in the BAD area of the disk.

When the disk structure is written, a zeroed replacement table is written to the RCT. Therefore, any bad blocks on the disk will exist without a replacement block so that any testing of the disk will then report them as "BAD" blocks (BB).

Bad block management cannot be done until the structure has been written to the disk.

Option 1 - Manually Replace Bad Block

This option prompts operator entry of an arbitrary block to be replaced as "bad".

If you enter a block number that is higher than the total number of blocks available, or enter an illegal non-numeric block number, then

WOMBAT displays the error message '* Failed *'.

Option 2 - Automatically Replace Bad Block from Error Statistics

This option replaces all bad blocks discovered during any or all of the three tests, subsequently displayed in the 'Display Error Scan Statistics' function. Any block with 10 or more errors will be marked as bad and replaced with a block in the BAD area.

Option 3 - Manually Enter Manufacturer's Defect Map

This option prompts for the drive manufacturer's defect map information as follows:

Enter defect map for drive 0
(Enter all values in decimal!!)
^C to exit at any time.
RETURN to back up 1 prompt.

Head: Enter the head number of defect.
Cylinder: Enter the cylinder number of defect.
Bytes past index: Enter the location past index of defect.

This data is then used to compute the address of a block on the disk. If it does not match a block on the specified disk track the error message "!! Beyond last sector" is produced. This may mean that the defect is located beyond the last data sector on the track, or that the entered data was wrong. WOMBAT then calculates the block number, displays it and replaces it. The "Head:" prompt is then repeated.

Option 4 - Display Replaced Bad Blocks

Displays all blocks in the Replacement Control Table and gives the total number of blocks replaced.

4.5.4 Initialize Controller

Option 4 forces the controller to perform its initialization functions: RAM clear and test, ROM test, and checks on the disk and its structure. Initialization also allows selection of a new drive.

4.5.5 Position Head on Shipping Zone

When option 5 is selected, the head is positioned on the default shipping zone cylinder number which is 340 unless an alternate zone is provided by the manufacturer (see drive data sheet). This provides a safeguard for data areas on the disk.

4.6 Wombat Error Messages

No Drive

Initialization tried to read the disk structure and failed. Check that the disk is connected correctly to the controller.

Disk Unstructured

Initialization can read the disk but cannot identify either copy of the data in the header blocks.

Cannot read RCT Table

Cannot read either copy of the RCT Table because of disk errors or the structure is not as expected.

Format error at block

An error was encountered while trying to format the disk at the specified block number. Since this is the first thing you do to a disk, it normally means that either the disk is not working at all, or the disk is not connected properly (e.g. wrong cables) to the controller. If the message occurs at other times the disk drive is probably very sick, or intermittent.

No Disk structure

An attempt has been made to display a disk structure on a disk that has not been structured.

Cannot replace this block

The block number entered is not within the bounds of the user area of the disk.

No RCT Table

Initialization could read neither of the two copies it keeps of the RCT (Replacement Control Table). Though you will destroy all information about replaced bad blocks, try writing the disk structure again. If errors persist in the RCT area, you may have to move the RCT to another cylinder on the disk. If the writing of the disk structure is successful this time, you can recover any replaced bad blocks by just performing a read test, which will bring up any unreplaced bad blocks as hard (10 retries) BB (Bad Block) errors. You may then replace them by using the "Automatically Replace Bad Blocks From Error Statistics" option.

No FCE Table

Initialization cannot locate the "Forced Error Table". Comments for "No RCT Table" apply.

RCT error

WOMBAT found the answer invalid when it computed the address of the RCT on disk. Check and respecify the bad block replacement table size in cylinders.

Disk error writing Disk structure at: (block)

For some reason, WOMBAT cannot write the structure details on to the disk. Test the disk thoroughly with the "Test Disk" menu. If that works properly, move the RCT and try again. If it still fails, you may have a bad disk or controller. Try component swapping.

Command Time-out

When all commands are executed, a counter is started to stop them "hanging". This counter has expired. If this is the first time the disk has been used, is the disk connected properly? Are the cables correct? Is there power to the disk? If the disk has a select light, is it on? Try powering off and on. Try component (disk and controller) swapping.

Fatal - Respecify Structure

An error has been detected in the disk structure specified; check and re-enter.

4.7 Wombat Self Diagnostics

A common initialization procedure exists for both WOMBAT and the MSCP firmware. It performs : a RAM integrity test, a ROM checksum, and various checks on the disk drive and its structure

The errors which can result from this are described under Section 3.9 Fatal Controller Errors.

ST506/412 is a trademark of Seagate Technology.

DEC, MSCP, LSI11, MicroVAX, RT-11, RSX-11M plus, RSTS/E, Micro#VMS, and Q-bus are trademarks of Digital Equipment Corporation.

TSX-plus is a trademark of S & H Computer Systems.

WOMBAT is a trademark of Webster Computer Corporation.