VH01 VMEBUS
SCSI/FLOPPY HOST ADAPTER
PROGRAMMER'S REFERENCE MANUAL

3545 Harbor Boulevard
Costa Mesa, California 92626
(714) 662-5600 TWX (910) 595-2521

VH0151002-00 Rev A
November, 1989

WARNING

This equipment generates, uses and can radiate radio frequency energy, and if not installed and used in accordance with the technical manual, might cause interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of Federal Communications Commission (FCC) Rules, which are designed to provide reasonable protection against such interference when operating in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures might be required to correct the interference.

VH01-01

Unix(R) is a registered trademark of AT&T. VMEbus is a trademark of the VMEbus International Trade Association. Motorola is a registered trademark of Motorola Corporation.

The information in this manual is for information purposes and is subject to change without notice.

Emulex Corporation assumes no responsibility for any errors which might appear in the manual.

Printed in U.S.A.

Audience

This manual is intended for use by people who will be installing, configuring, or programming a device driver for the VH01 Host Adapter, such as a system integration company technical staff person, a programmer, or a technician. The manual assumes familiarity with the VMEbus Specification, Rev. C (April 1982), and the Small Computer System Interface (SCSI) Standard.

References

In addition to this manual, the reader might need to refer to the following documents:

ANSI X3.131-1986, Small Computer System Interface (SCSI) Standard, American National Standards Institute, 1430 Broadway, New York, New York 10018

Emulex VMEbus UNIX Software Installation and User's Guide, VS9950901-00, Emulex Corporation, 3545 Harbor Boulevard, Costa Mesa, California 92626

The VMEbus Specification C.1, HB212, Motorola Literature Distribution, P.O. Box 20912, Phoenix, Arizona 85036

Format Conventions

Address and data values in this manual are in hexadecimal notation, while bit numbering is in decimal. The diagrams and tables provided show all parameter field values in hexadecimal and bit numbering in decimal.

Except as otherwise noted, the terms "base," "base address," and "BA," which are used throughout this manual, all refer to the value set for the VMEbus Base I/O Address of the VH01's VMEbus DRAM Window. The settings of switches SW1-2 through SW1-6 determine this base address. The base address is usually shown with an offset in bytes (Base+X). The offset is in hexadecimal.

The text paragraphs of this manual show all hexadecimal values with a leading "0x".

Bits shown as "R" are reserved for the exclusive use of the VH01, and should be set to zero by the host processor.

BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PROGRAM LISTINGS

BLANK

STATEMENT OF WARRANTY

BASIC WARRANTY - In the absence of any optional warranty or continuing provisions extended by formal agreement, Emulex warranties its Products in accordance with the schedules listed below.

In all cases where equipment is to be returned to Emulex, a "Return Authorization" must be issued by the "Repair Center," (714) 662-5600.

CONTROLLERS - PERIPHERALS - DATA COMMUNICATION PRODUCTS - Emulex warrants for a period of twelve (12) months, sixty (60) months in case of FAX modems and Data Pump modems, from the date of shipment, that each product shall be free of defects in material and workmanship. These products include all Disk and Tape Controllers, Magnetic Disk and Tape Drives, Optical Disk Drives, Communication Multiplexers, and modem products.

During this period, if the customer experiences difficulties with an Emulex product and is unable to resolve the problem via the phone with Emulex Technical Support, a Return Authorization will be issued. Following receipt of a Return Authorization, the customer is responsible for returning the product to Emulex, freight prepaid. Emulex, upon verification of warranty will, at its option, repair or replace the product in question, and return to the customer freight prepaid.

PACKAGED SUBSYSTEMS - Emulex warrants all packaged subsystems for a period of six (6) months from the date of shipment. These products include all versions of Sabre, Vault, Medley, Decathlon, and Javelin. In the event of difficulty, the procedure for problem resolution is the same as defined above. Please note that major assemblies of these packages including controllers and peripherals are covered under the 12 month warranty above if returned to Emulex as separate assemblies.

SOFTWARE WARRANTY - Emulex warrants for a period of ninety (90) days, from the date of shipment, that each software package supplied shall be free from defects and shall operate according to Emulex specifications under those Digital Equipment Corporation ("DEC"), IBM, Intel, and Unix, operating system versions supported by Emulex. Emulex does not warrant its software products under any operating system which has not been specifically identified. Any software revisions required hereunder will cover supply of distribution media only and will not cover on-site installation or integration.

CABLE WARRANTY - (Return to Factory) - All Emulex-provided cables are warranted for ninety (90) days from the time of shipment. Questionable cables should be returned to Emulex, freight prepaid, where they will be repaired or replaced by Emulex at its option and returned to the customer freight prepaid.

MEDIA WARRANTY - Emulex warrants all media to be free of defects and material for a period of thirty (30) days from date of shipment. The customer is responsible for return of media to Emulex, and Emulex is responsible for freight associated with replacement media being returned to the customer.

RETURN TO FACTORY - All of the warranties referenced above provide for corrective action on products returned to the Emulex factory accompanied by a "Return Authorization".

On-site services are not included as a part of these warranties.

GENERAL TERMS - The above warranties shall not apply to expendable components such as fuses, incandescent lamps, and the like, nor to connectors and other items not a part of the basic product. Emulex shall have no obligation to make repairs or to cause replacement required through normal wear and tear or necessitated in whole or in part by catastrophe, fault or negligence of the user, improper or unauthorized use of the Product, or use of the Product in such manner for which it was not designed, or by causes external to the Product, such as, but not limited to, power failure or air conditioning. Emulex's sole obligation hereunder shall be to repair or replace items covered in the above warranties, unless covered by an Emulex On-Site Warranty Contract. Purchaser shall provide for removal of the defective Product, shipping charges for return to Emulex, and installation of its replacement. Above warranties are subject to change without notice.

RETURNED MATERIAL -Warranty claims must be received by Emulex within the applicable warranty period. A replaced product, or part thereof, shall become the property of Emulex and shall be returned to Emulex at Purchaser's expense. All returned material must be accompanied by a Return Authorization Number assigned by Emulex.

In the event product(s) returned under the provisions of this Warranty are subsequently determined by Emulex to be functionally operational and in accordance to its published specifications, i.e., "No Defect Found" (NDF), Purchaser will be charged a NDF fee and the product shall be returned to Purchaser freight collect.

THE EXPRESS WARRANTIES SET FORTH IN THIS AGREEMENT ARE IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, AND ALL SUCH OTHER WARRANTIES ARE HEREBY DISCLAIMED AND EXCLUDED BY EMULEX. THE STANDARD EXPRESS WARRANTIES ARE IN LIEU OF ALL OBLIGATIONS OR LIABILITIES ON THE PART OF EMULEX FOR DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE PRODUCTS.

Section 1
GENERAL DESCRIPTION

1.1 Introduction

This manual describes the installation, programming, and application of the Emulex VH01 VMEbus SCSI/Floppy Host Adapter. It is designed to be used for setup, installation, cabling, and programming of the VH01 Host Adapter. This manual does not cover software driver installation or usage; that information is contained in the *Emulex VMEbus UNIX Software Installation and User's Guide,* Emulex part number VS9950901-00.

The contents of the manual are as follows:

Section 1	**General Description**
Section 2	**Specifications**
Section 3	**Installation**
Section 4	**VH01 Architecture**
Section 5	**Programming the VH01**
Section 6	**Application Information**
Appendix A	**Troubleshooting**
Appendix B	**Connector Pin Assignments**
Appendix C	**SCSI Command Summary**
Appendix D	**Floppy Command Set**
Appendix E	**VH01 Error Codes**

Glossary

1.2 Overview

The VH01 is an MC68000-based SCSI Host Adapter, designed for excellent performance in interfacing the SCSI bus to the VMEbus. The VH01 maintains low-level SCSI protocol control, thereby requiring less intervention by the VMEbus host processor. The VH01 features a high-speed FIFO capable of supporting VMEbus DMA burst rates of 35M bytes per second. In addition, the VH01 also supports both contiguous and scatter/gather memory transfers as well as block mode DMA transfers. (A host adapter differs from a controller in that a controller controls one or more peripherals, whereas a host adapter adapts a different bus to the bus of the host computer.)

Through the use of the Emulex SCSI Processor (ESP) IC, the VH01 offers very low SCSI phase change overhead, as well as 4M bytes per second synchronous and 3M bytes per second asynchronous SCSI data transfer rates. Further, the VH01 supports disconnect/reconnect, multi-threaded operations, and a full SCSI implementation including the Common Command Set (CCS).

The VH01 offers three configuration options. Each of the three VH01 configurations can support up to seven controllers (SCSI targets). Since a SCSI target can have up to 8 peripherals, the VH01's SCSI port can support a total of up to 56 peripherals. Also, the VH01 configuration with the floppy port can support up to two floppy disk drives, in addition to the SCSI peripherals.

The three VH01 configuration options are as follows:

- SCSI port with single-ended drivers
- SCSI port with single-ended drivers plus floppy port
- SCSI port with differential drivers

The VH01 is built on a standard, double height, expanded Eurocard (6U-EXP), using surface mount technology (SMT). In addition, the VH01 is fully compliant with the P-1014/REV.C.1 VMEbus specification. Accordingly, the VH01 is completely hardware compatible with any VMEbus system that complies with this industry standard specification and module size.

1.3 Models and Options

Table 1-1 lists the model numbers and part numbers for the Emulex VH01 SCSI/Floppy Host Adapter and accessories.

Table 1-1. VH01 Model and Part Numbers

Model	Part Number	Description
VH01/S	VH0110202-02	VH01 Host Adapter Board, single-ended SCSI port
VH01/SF	VH0110202-03	VH01 Host Adapter Board, floppy port plus single-ended SCSI port
VH01/D	VH0110203-02	VH01 Host Adapter Board, differential SCSI port
-----	-----	Cables; see Tables 3-3 and 3-4 for choices
-----	VS9951801-01	Emulex Unix Drivers and Utilities; media on 1/2 inch 9-track magnetic tape
-----	VS9951801-02	Emulex UNIX Drivers and Utilities; media on low-density (360K bytes) 5.25 inch floppy disk
-----	VS9951801-04	Emulex UNIX Drivers and Utilities; media on 1/4 inch magnetic tape cartridge (Sun format)
-----	VH0151002-00	*VH01 VMEbus SCSI/Floppy Host Adapter Programmer's Reference Manual*
-----	VS9950901-00	*Emulex VMEbus UNIX Software Installation and User's Guide* (included with Drivers and Utilities)

1.4 Product Features and Capabilities

The following is a list of the product features and capabilities of the VH01. Each of these is described in detail in the following subsections.

- Packet Protocol for VMEbus Host Communication
- VMEbus data transfer rates of up to 35M bytes/sec
- VMEbus block mode transfers (BLT)
- Programmable VMEbus burst length
- VMEbus transfers starting and ending on odd byte boundaries are supported in hardware
- VMEbus block scatter/gather
- VMEbus address pipelining

- Programmable address modifier during VMEbus mastership
- Switch-selectable VMEbus address setting
- VMEbus overlapped arbitration
- Programmable VMEbus Release
- Supports multiple VMEbus host processors
- Command queuing
- Command combining
- Overlapped commands
- Seek ordering
- SCSI data transfer rates of greater than 3M bytes/sec (asynchronous), and 4M bytes/sec (synchronous)
- SCSI port supports up to 56 peripherals -- 8 peripherals on each of 7 controllers (SCSI targets)
- Switch-selectable SCSI ID setting
- Differential or single-ended option for SCSI port
- Optional floppy port supports up to two SA450-compatible double-density or quad-density floppy disk drives
- Target mode

1.4.1 Packet Protocol for VMEbus Host Communication

The VH01 uses an Emulex-designed packet communication protocol for receiving commands from and issuing responses to the VMEbus host processor. These packets contain the actual SCSI Command Descriptor Block (CDB) to allow for complete SCSI "pass-through".

1.4.2 VMEbus Data Transfer Rates

The VH01 can perform either Block Mode Transfers (BLT), or non-BLT transfers to and from a VMEbus memory module at speeds up to 35M bytes/sec; however, since the VH01 is designed with asynchronous state machines this time might be slightly better or slightly worse.

1.4.3 VMEbus Block Mode Transfers (BLT)

Block Mode Transfers (BLT) on the VMEbus are defined as back-to-back transfers which do not broadcast a new address with each transfer; thus, the address is only broadcast once, at the beginning of the transfer. The address must be rebroadcast when crossing a 256-byte boundary. The VH01 does this by disconnecting from the VMEbus, rearbitrating, and then continuing the BLT transfer, with the first transfer broadcasting the address.

1.4.4 Programmable VMEbus Burst Length

The VMEbus host processor can control the length of time that the VH01 remains on the VMEbus. The host does this by specifying the number of transfers that the VH01 can perform each time it acquires the VMEbus. The host can specify from 1 to 16 transfers. Each transfer is either a longword or word, as specified by the host. Once the transfer limit is reached, the VH01 will release the VMEbus, and it will not rearbitrate for the bus until it can maintain a VMEbus burst of at least eight words. The burst length applies to both BLT and non-BLT transfers.

1.4.5 Hardware-supported VMEbus Odd Byte Boundaries

The VH01 hardware automatically handles VMEbus transfers that start and/or end on an odd byte boundary or odd word boundary, eliminating firmware overhead.

1.4.6 VMEbus Block Scatter/Gather

Scatter/gather allows the host to gather the source data blocks or scatter the destination data blocks in noncontiguous blocks of memory. Hardware on the VH01 assists scatter/gather. Subsection 5.6.1.18 describes scatter/gather in detail.

1.4.7 VMEbus Address Pipelining

When the VH01 is transferring data to a VMEbus slave memory module, and BLT is not enabled, then the VMEbus address must be stable on the VMEbus until the slave responds with a VMEbus DTACK signal, at which time the VH01 will increment the address in preparation for the next transfer. The VMEbus slave must tolerate the address changing at this time, even if the VH01 keeps the VMEbus data strobes active. Address pipelining always occurs on the VH01 when BLT is not enabled. When BLT is enabled, address pipelining is not necessary, since the address is only broadcast once at the beginning of the burst.

1.4.8 VMEbus Address Modifiers

During transfers as a bus master, the VH01 supports all VMEbus address modifier codes. The VMEbus host processor can specify which address modifier it desires with each command it issues to the VH01. However, when the host writes or reads to the VH01's VMEbus Window, the VH01 responds only to address modifier codes 0x29 and 0x2D if short space is selected, or 0x39 and 0x3D if standard space is selected.

1.4.9 Switch-selectable VMEbus Address Setting

The VMEbus host processor communicates with the VH01 via 512 bytes of the VH01's internal memory. This block of memory is accessable in either VMEbus host short space (0x29 or 0x2D) or standard space (0x39 or 0x3D). Switches on the VH01 select both the starting address of this block of memory and the type of addressing (standard or short space).

1.4.10 VMEbus Overlapped Arbitration

Overlapped arbitration on the VMEbus occurs when the VH01 releases the VMEbus BUSY signal before the last data transfer has completed, allowing bus arbitration to occur while the last transfer is still in progress. The VH01 will optionally perform overlapped arbitration during BLT FIFO data transfers.

1.4.11 Programmable VMEbus Release

The VMEbus host processor can select either Release-On-Request (ROR) or Release-When-Done (RWD) transfers for the VH01. If the host selects ROR, the VH01 will release its ownership of the VMEbus when another VMEbus device requests the bus. If, however, the host selects RWD, the VH01 will not release the VMEbus until the VH01 reaches its burst limit.

1.4.12 Multiple VMEbus Host Processors

The VH01 can function as either a VMEbus master or slave. The VH01 is a master during command, data, and status transfers to and from the VMEbus. The VMEbus host processor communicates with the VH01 via 512 bytes of the VH01's internal memory. This block of memory is accessable as a VMEbus slave, in VMEbus host short space (0x29 or 0x2D) or standard space (0x39 or 0x3D). The VH01 can issue an interrupt to any of the seven VMEbus interrupt lines, and then respond with an 8-bit vector when the interrupt is acknowledged. Thus, multiple VMEbus host processors can communicate with the VH01 with minimal resource control.

1.4.13 Command Queuing

The VH01 internally queues all VMEbus host commands that are directed to any of the SCSI devices. The VH01 executes all commands that are directed to the same SCSI device in the order received, except when seek ordering and/or command combining are enabled.

1.4.14 Command Combining

When the host issues several read commands to the VH01 that reference sequential areas on the same peripheral, the VH01 can optionally combine them into one SCSI command, to optimize SCSI bus performance. Similarly, the VH01 can combine several write commands that reference sequential areas on the same peripheral.

1.4.15 Overlapped Commands

The VH01 internally queues all VMEbus host commands that are directed to any of the SCSI devices. During a SCSI disconnect with a device awaiting data transfer, the VH01 will overlap commands on the SCSI bus by issuing commands to other devices.

1.4.16 Seek Ordering

When the host issues random access commands for the same direct access device to the VH01, the VH01 can optionally sequence them according to block address, using one of several ordering algorithms.

1.4.17 SCSI Data Transfer Rates

The VH01 supports both synchronous and asynchronous modes of operation on the SCSI bus. The VH01 negotiates with the SCSI target for synchronous transfers the first time it issues a command to it, unless the VMEbus host processor has disabled the VH01 from entering synchronous mode negotiations. The VH01 can transfer data to and from the SCSI device at rates greater than 3M bytes/sec in asynchronous mode and 4M bytes/sec in synchronous mode.

1.4.18 SCSI Targets

The SCSI port on the VH01 supports up to 7 controllers (SCSI targets), each of which can support up to 8 peripherals, for a total of 56 possible peripherals.

1.4.19 Switch-selectable SCSI ID Setting

Switches on the VH01 set the VH01's SCSI ID.

1.4.20 Differential or Single-ended SCSI Port

The VH01's SCSI port supports either single-ended SCSI or differential SCSI, depending upon the model. A 50-pin ribbon connector on the VH01 front panel provides the SCSI signals. This connector is labeled SCSI on the front panel, and is also designated J2. The VMEbus P2 connector also provides the SCSI signals.

1.4.21 Floppy Port

A floppy disk drive port is available as an ordering option on the single-ended SCSI VH01 only. This floppy port supports up to two 3-1/2 inch or 5-1/4 inch Shugart SA450-compatible double-density or quad-density floppy disk drives, with data transfer rates of 250K bit/sec, 300K bit/sec, and 500K bit/sec.

1.4.22 Target Mode

In addition to acting as a SCSI initiator, the VH01 can also respond as a SCSI target device and accept commands from another initiator. As a target device, the VH01 can be selected by another SCSI initiator (host adapter), and will respond as a processor type of device. This feature allows high-speed host-to-host data transfers to take place over the SCSI bus.

1.5 Performance

The VH01 includes many features designed to enhance performance. An MC68000 microprocessor, running at 12 MHz, controls all VH01 functions and handles communications with the VMEbus host processor. A PAL-based asynchronous state machine performs read and write timing and data routing.

1.5.1 Speed

The VH01 is built around the Emulex SCSI Processor (ESP) IC, now an industry standard. The ESP IC implements the full SCSI protocol, performing bus arbitration and supporting disconnect/reconnect with multithreaded operation.

By incorporating the ESP IC, the VH01 can achieve the highest data rates possible with SCSI. In addition to the 4M bytes per second transfer rate achievable in synchronous mode, the VH01 can handle asynchronous rates greater than 3M bytes per second.

When activity on the VMEbus is high, data being transferred to the VMEbus host can be stored in a high-speed FIFO on the VH01 that holds 64 words (128 bytes). Block mode transfers (BLT) can then be made to the host, with burst rates up to 35M bytes per second and non-BLT transfers exceeding 20M bytes per second. For further system efficiency, the VH01 also supports overlapped arbitration, releasing Bus Busy early in the cycle.

1.5.2 Versatility

The VH01 features a full SCSI implementation, as well as pass-through and the SCSI Common Command Set (CCS). As a result, the VH01 supports virtually all SCSI-based devices, including optical devices. This provides the benefit of maximum flexibility in selecting the ideal device complement for any VMEbus-based system.

With intelligent command processing ability, the VH01 supports multiple VMEbus hosts through a packet interface, as well as SCSI disconnect/reconnect for better system efficiency. With SCSI disconnect/reconnect capability, an intelligent peripheral can release the SCSI bus during time-consuming operations, freeing the bus to service another device. During a disconnect, the VH01 and the peripheral are still logically threaded together, and physically reconnect when the task is finished. By multithreading, several SCSI I/O commands can execute in the system simultaneously, so that operations on multiple devices can overlap.

The versatile VH01 also features block scatter/gather DMA transfers on the VMEbus. This improves VMEbus memory utilization and efficiency. This feature is particularly beneficial when large transfers are being made to memory that consists of small blocks in noncontiguous address space. By using noncontiguous memory locations, this technique frees up large contiguous areas for use by other processes or controllers that do not support scatter/gather.

1.5.3 Easy Integration and Application

The VH01 is designed to be efficient and easy to use. Switches allow easy selection of both the VH01's SCSI bus ID and the VH01's VMEbus base address. The VH01 also features an extensive internal self test. This self test allows the installation of the VH01 to be easily verified and helps to isolate faults, resulting in easier installation and less down time.

1.6 Compatibility

1.6.1 VMEbus Compatibility

The VH01 conforms to the VMEbus Specification, Rev. C (April 1982).

1.6.2 SCSI Compatibility

The SCSI port on the VH01 conforms to the SCSI standard (ANSI X3.131-1986), and supports the SCSI Common Command Set (CCS), SCSI "pass-through," and any vendor-unique commands provided with a SCSI device.

1.6.3 Floppy Disk Drive Compatibility

The optional floppy port on the VH01, if present, is compatible with the Shugart SA450 floppy disk drive.

1.6.4 Software Compatibility

The VH01 is compatible with most operating systems running on VMEbus systems. Emulex supplies example drivers, to be used to create drivers to run on a specific operating system, for some of the different versions of the UNIX Operating System, such as AT&T System V.2, AT&T System V.3, Sun 4.0, and BSD 4.3. See Table 1-1 for the Emulex part numbers of the example drivers.

BLANK

Section 2
SPECIFICATIONS

2.1 Overview

This section contains general and electrical, VMEbus interface, mechanical, and environmental specifications for the VH01, as well as lists of supported devices and media. This section contains the following subsections:

Subsection	Title
2.1	Overview
2.2	General and Electrical Specifications
2.3	VMEbus Interface Specifications
2.4	Mechanical Specifications
2.5	Environmental Specifications
2.6	Floppy Drives and Media
2.7	Devices Certified for Use with the VH01

2.2 General and Electrical Specifications

Table 2-1 contains general and electrical specifications for the VH01.

Table 2-1. VH01 General and Electrical Specifications

Parameter	Description
Memory:	256K bytes RAM (512 bytes dual-ported), 128 bytes on-board FIFO memory
Switches:	One 10-position DIP switch module for resetting the VH01, setting the VH01 VMEbus base address, and setting the VH01 SCSI ID
Indicators:	Four LEDs for status display
Electrical: Power	6 A max. (5.5 A typ.) at +5 VDC 100 mA max. (50 mA typ.) at +12 VDC

2.3 VMEbus Interface Specifications

Table 2-2 contains VMEbus interface specifications for the VH01.

Table 2-2. VH01 VMEbus Interface Specifications

Parameter	Description
VMEbus Specification Version:	VMEbus Specification, Rev. C (April 1982)
Bus Interface Options:	
Master Data Transfer Options	A32, A24, A16, D32, D16, D8 Address modifiers are programmable
Slave Data Transfer Options	A24, A16, D16, D8 Short supervisory (0x2D) Short nonprivileged (0x29) Standard supervisory (0x3D) Standard nonprivileged (0x39)
Requester Options	Any one of R(0) through R(3) Static (STAT) selection of all request lines Release-When-Done (RWD) Release-On-Request (ROR) during BLT
Interrupter Options	Any one of I(1) through I(7) Programmable
VMEbus Starting Address	Switch selectable to any of the following hexadecimal values: 0000, 0200, 0400, 0600, 0800, 0A00, 0C00, E000, F000, F200, F400, F600, F800, FA00, FC00, or FE00 (short space); FF0000, FF0200, FF0400, FF0600, FF0800, FF0A00, FF0C00, FFE000, FFF000, FFF200, FFF400, FFF600, FFF800, FFFA00, FFFC00, or FFFE00 (standard space) Switch selectable standard or short space addressing

2.4 Mechanical Specifications

Table 2-3 contains mechanical specifications for the VH01. Figure 2-1 depicts the VH01 PCB dimensions.

Table 2-3. VH01 Mechanical Specifications

Parameter	Description
Dimensions:	
PCB	6U/EXP (expanded) double-height Eurocard - 160 mm x 233.35 mm (6.299 inches x 9.187 inches)
Front Panel	Single width, double-height VME panel - 20 mm x 261.85 mm (0.787 inches x 10.309 inches)
Connectors	Front panel: one standard 50-pin SCSI connector and one standard SA450-compatible 34-pin floppy connector Backplane: P1 and P2 standard 96-pin DIN connectors

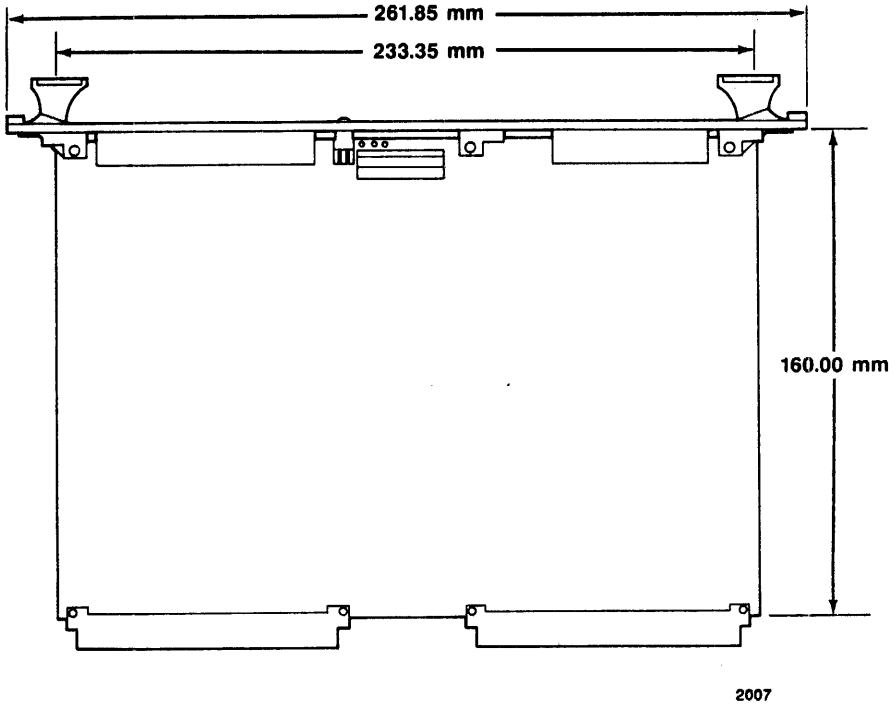


Figure 2-1. VH01 PCB Dimensions

2.5 Environmental Specifications

Table 2-4 contains environmental specifications for the VH01.

Table 2-4. VH01 Environmental Specifications

Parameter	Description
Operating Temperature:	0°C to 50°C (32°F to 120°F) Maximum temperature is reduced 1.8°C per 1000 meters (1°F per 1000 feet) altitude
Storage Temperature:	-40°C to 66°C (-68°F to 151°F)
Humidity Range:	10% to 90%, noncondensing
Emission:	FCC Part 15 Class A approved (in a hardened enclosure)

2.6 Floppy Drives and Media

2.6.1 Drive Types

At VH01 system startup, the VH01 automatically interrogates each floppy drive and determines the drive type. The VH01 supports the following drive types:

- 5-1/4 inch double-sided, 48 tracks per inch, 300 RPM, 250 KBPS
- 5-1/4 inch double-sided, 96 tracks per inch, 360 RPM, 500 or 300 KBPS
- 3-1/2 inch double-sided, 135 tracks per inch, 300 RPM, 250 or 500 KBPS

2.6.2 Medium Types

Whenever a formatted diskette is inserted into the drive, the VH01 automatically interrogates the drive to determine the medium type. The VH01 supports only double-sided diskettes recorded in the IBM Modified Frequency Modulation (MFM) (double density) format. The VH01 supports the following media configurations:

- 5-1/4 inch double-sided, 48 tracks per inch drive

  256 byte sectors, 15 sectors per track
  512 byte sectors, 9 sectors per track (360 KB)
  1024 byte sectors, 4 sectors per track

- 5-1/4 inch double-sided, 96 tracks per inch drive

  256 byte sectors, 26 sectors per track
  512 byte sectors, 15 sectors per track (1.2 MB)
  1024 byte sectors, 8 sectors per track

- 3-1/2 inch double-sided, 135 tracks per inch drive

  256 byte sectors, 15 or 30 sectors per track
  512 byte sectors, 9 or 18 sectors per track (720 KB or 1.44 MB)
  1024 byte sectors, 5 or 10 sectors per track

The VH01 also supports media formatted for 48 tracks per inch in a 96 tracks per inch drive at a data rate of 300 KBPS.

In addition to the preceding media configurations, the VH01 also supports other user-specified sectors-per-track configurations; for example, 5-1/4 inch 48 tracks per inch media with 8 sectors per track and 512-byte sectors (320 KB). In such cases, the user must also specify the correct gap length parameters for formatting and read/write. The new media parameters must be specified via the SCSI Mode Select command prior to inserting the diskette. However, if the VH01 detects a sector size different from the currently-configured sector size on newly inserted media, it will override the user specified sectors per track and gap lengths with one of the preceding default configurations, based on drive type and sector size.

If the VH01 does not recognize the media as one of the preceding configurations, it will report a "medium error" and will not support the medium.

2.7 Devices Certified for Use with the VH01

The following subsections list devices that Emulex has specifically certified for use with the VH01.

2.7.1 Certified SCSI Devices

Unlike a number of proprietary systems, the VH01 does not have to emulate any specific device protocol. This, combined with a full SCSI implementation, allows the VH01 to support virtually any SCSI-based device. Since verifying all possible SCSI devices for use with the VH01 would be an unrealistic endeavor, the following SCSI devices are specifically certified to operate with the VH01's SCSI port:

- Emulex MTxx series 1/4-inch tape controllers
- Emulex MDxx series winchester controllers (with CCS compatibility)
- MAXTOR 5-1/4 inch winchester drive with embedded SCSI controller
- Micropolis 5-1/4 inch winchester drive with embedded SCSI controller
- CDC 5-1/4 inch winchester drive with embedded SCSI controller
- LMSI (CDC) LD1200 12-inch optical disk drive
- Optimem 1000 12-inch optical drive
- Exabyte EXB-8200 8mm cartridge tape drive

2.7.2 Certified Floppy Disk Drives

The following Shugart SA450-compatible floppy disk drives are specifically certified to operate with the VH01's floppy port:

5-1/4 inch dual density:

- TEAC FD-55FV-13
- Hitachi HFD-505B
- Fujitsu 2551A
- NEC FD1055

5-1/4 inch quad density:

- Panasonic JU-475-2
- Fujitsu M2553K
- Toshiba ND-08DE
- YE Data YD-380
- Mitsubishi MF504A

3-1/2 inch dual density:

- Fujitsu M253A
- Panasonic JU-364

In addition, the VH01's floppy port supports any other floppy disk drives that meet the specifications of subsection 2.6.

BLANK

Section 3
INSTALLATION

3.1 Overview

This section describes the procedures for installing and checking the VH01 Host Adapter. The subsection titles are listed below to serve as an outline of the procedure.

Subsection	Title
3.1	Overview
3.2	Inspection
3.3	Considerations for Use of the VH01 in VMEbus Systems
3.4	Configuring the VH01 Board
3.5	Installing the VH01 in a Backplane
3.6	Cabling
3.7	Power-Up and System Verification

3.1.1 Maintaining FCC Class A Compliance

Emulex has tested the VH01 Host Adapter for FCC compliance. The VH01 complies with FCC Class A limits for radiated and conducted interference.

3.2 Inspection

Emulex products are shipped in special containers designed to provide full protection under normal shipping conditions. Immediately upon receipt, the shipping container should be inspected for evidence of possible damage incurred in transit. Any obvious damage to the container, or indications of actual or probable equipment damage, should be reported to the carrier company in accordance with instructions on the form included in the container.

Unpack the VH01 Host Adapter and, using the shipping invoice, verify that all equipment is present. Verify also that model or part number (P/N) designation, revision level, and serial numbers agree with those on the shipping invoice. Subsection 1.3 explains the VH01 model and part numbers. These verifications are important to confirm warranty. If evidence of physical damage or identity mismatch is found, notify an Emulex representative immediately. If the equipment must be returned to Emulex, it should be shipped in the original container.

Visually inspect all components after unpacking. Check for such items as bent or broken connector pins, damaged components or any other evidence of physical damage.

Examine all socketed components carefully to ensure they are properly seated.

CAUTION

Be sure to take static-handling precautions when handling the VH01 Host Adapter. Damage to some of the components on the VH01 Host Adapter might occur due to static electricity if precautions are not taken.

3.3 Considerations for Use of the VH01 In VMEbus Systems

The VH01 requires good airflow to cool the comparatively large number of parts on it. Forced air cooling is mandatory for the VH01 board.

The VH01 requires a backplane with low-impedance grounds to avoid noise problems. If the VMEbus backplane grounds are poor, intermittent errors are likely to occur. A low-noise backplane permits the host adapter to operate correctly.

3.4 Configuring the VH01 Board

The following subsections describe how to set the switches and jumpers on the VH01 board so that they are compatible with your VMEbus system.

3.4.1 VH01 Board Component Locations

Figure 3-1 shows the location on the VH01 board of switch block SW1, jumper blocks J3 and J4, and other important components.

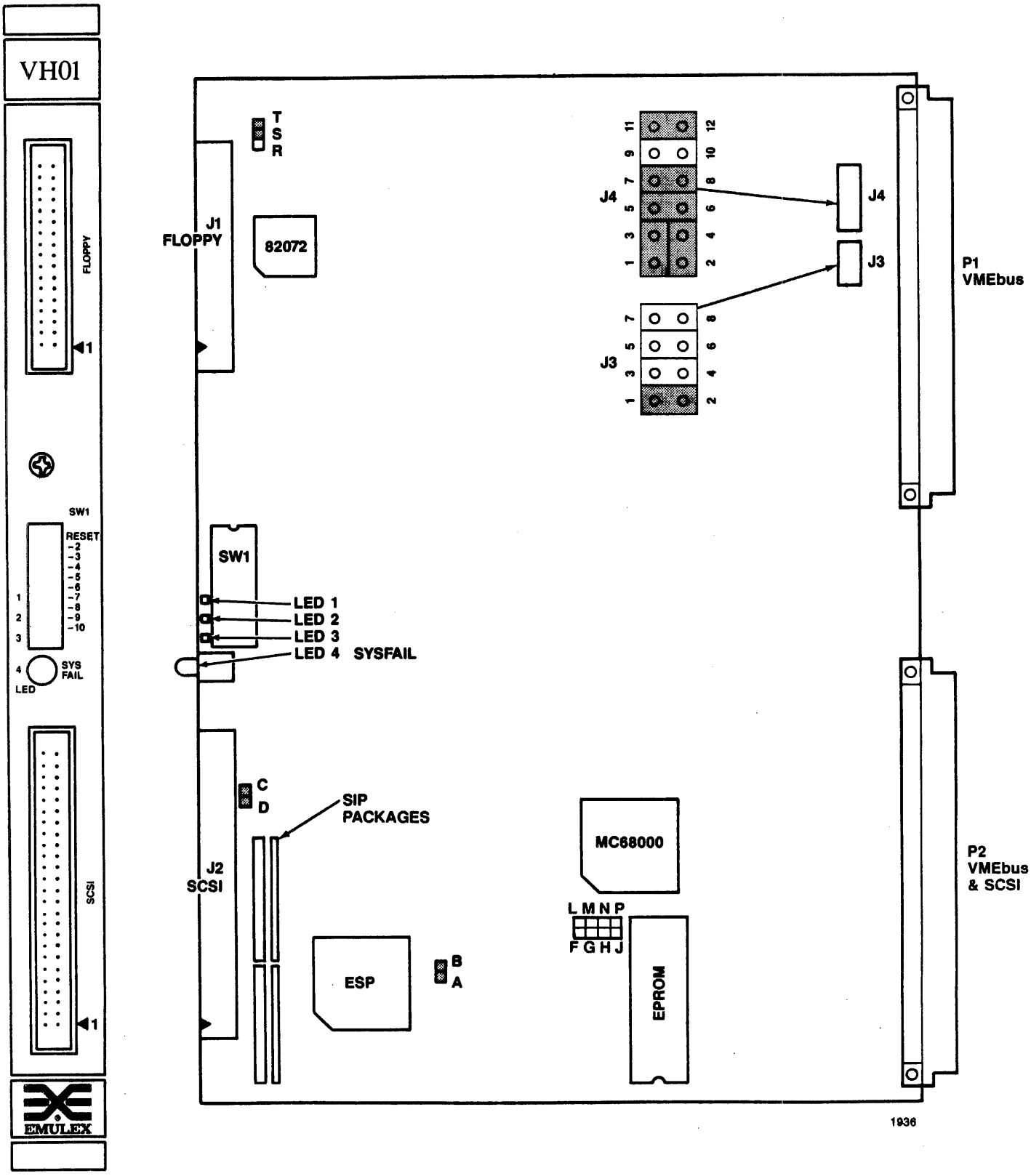
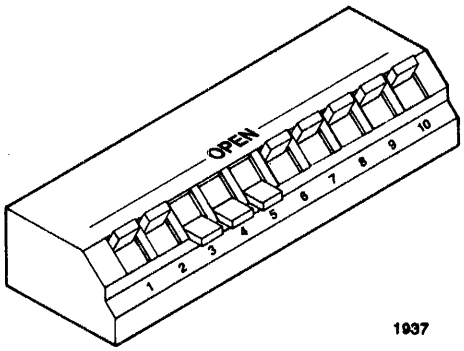


Figure 3-1. VH01 Board Component Locations

3.4.2 DIP Switch Types

Switch-setting tables in this manual use the numeral zero (0) to indicate the OFF (open) position and the numeral one (1) to indicate the ON (closed) position.

The DIP switch type used in this product is shown in Figure 3-2. The switch is set to the code shown in the switch setting example.



--- SW1 ---									
1	2	3	4	5	6	7	8	9	10
0	0	1	1	1	0	0	0	0	0

Figure 3-2. Switch Setting Example

3.4.3 SW1 Switch Settings

The VH01 board has one dual-inline package (DIP) switch, labeled SW1 (see Figure 3-1 for location). The following subsections describe the switch functions.

3.4.3.1 Reset VH01 (SW1-1)

This switch must be OFF (open) for normal operation. When this switch is ON (closed), the VH01 is disabled, and it will continuously assert the SYSFAIL signal on the VMEbus. When this switch is toggled from on to off, the VH01 will assert the SYSFAIL signal on the VMEbus, set the Adapter Self Test Byte to one, de-assert SYSFAIL, and begin its self-test and self initialization. If the self-test and self initialization complete without errors (the normal case), the VH01 will then set the Adapter Self Test Byte to zero. The self-test and self initialization together require approximately ten seconds or less to complete.

NOTE

Toggling the reset switch (SW1-1) should never be done when the operating system is running. Toggling this switch while the operating system is running might cause a system crash.

3.4.3.2 VH01 VMEbus Base Address (SW1-2 through SW1-6)

Switch SW1-2 together with switches SW1-3 through SW1-6 control the type of addressing and the VMEbus base address of the VH01's shared memory.

Switch SW1-2 selects the type of addressing for the shared memory. If switch SW1-2 is set to the OFF (open) position, the shared memory is accessible to the VMEbus using VMEbus short (16-bit) addressing, with VMEbus address modifier codes of 0x29 and 0x2D. If switch SW1-2 is set ON (closed), the shared memory is accessible using VMEbus standard (24-bit) addressing, with address modifiers of 0x39 and 0x3D.

Switches SW1-3 through SW1-6 select the VMEbus base address (BA) of the 512 bytes of shared memory on the VH01 board.

Table 3-1 lists the switch settings and their corresponding base addresses.

Table 3-1. VH01 VMEbus Base Address Switch Settings

VMEbus Base Address (BA) (hexadecimal)		SW1-3	SW1-4	SW1-5	SW1-6
Short Space (SW1-2 OFF)	Standard Space (SW1-2 ON)				
0000	FF0000	1	1	1	1
0200*	FF0200	1	1	1	0
0400	FF0400	1	1	0	1
0600	FF0600	1	1	0	0
0800	FF0800	1	0	1	1
0A00	FF0A00	1	0	1	0
0C00	FF0C00	1	0	0	1
E000	FFE000	1	0	0	0
F000	FFF000	0	1	1	1
F200	FFF200	0	1	1	0
F400	FFF400	0	1	0	1
F600	FFF600	0	1	0	0
F800	FFF800	0	0	1	1
FA00	FFFA00	0	0	1	0
FC00	FFFC00	0	0	0	1
FE00	FFFE00	0	0	0	0

*Default.

NOTE: A zero (0) indicates that a switch is OFF (open).

3.4.3.3 VH01 SCSI ID (SW1-7 through SW1-9)

Table 3-2 shows how to set switches SW1-7, SW1-8, and SW1-9, to select the SCSI bus ID of the VH01.

Table 3-2. VH01 SCSI ID Selection

SCSI ID	SW1-7	SW1-8	SW1-9
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	0	0	1
7*	0	0	0

*Default.

NOTE: A zero (0) indicates that a switch is OFF (open).

3.4.3.4 Reserved Switch (SW1-10)

Switch SW1-10 is reserved and should be set to the OFF (open) position.

3.4.4 Jumper Settings on the VH01

There are six sets of jumper blocks on the VH01 board. See Figure 3-1 for the locations of the jumper blocks on the board. The following subsections describe the functions of the jumper settings.

3.4.4.1 DMA-Priority Jumpers (Jumper Blocks J3 and J4)

The jumper blocks labeled J3 and J4 set the VH01's DMA priority level. There are four possible settings. Priority 3 gives the VH01 the highest DMA priority, while Priority 0 gives the board the lowest priority. The default DMA priority for the VH01 is 3, and the VH01 board is shipped with this setting. Figure 3-3 depicts the jumper settings for each of the four DMA priority levels. Use Figure 3-1 for proper orientation of the board.

NOTE

DMA Priority 3 **MUST** be selected in systems using single-level arbitration (SGL Arbiter option).

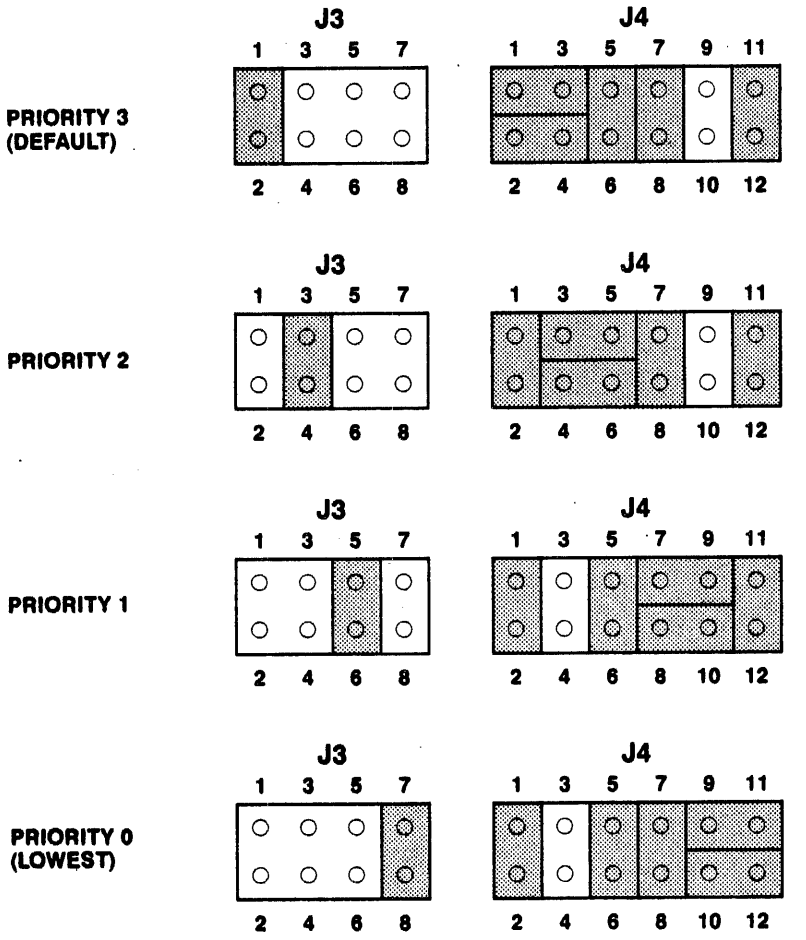


Figure 3-3. Jumper Settings for DMA Priority

3.4.4.2 Factory Test Jumpers

Referring to Figure 3-1, there is a small jumper block labeled **AB** near the ESP IC. This jumper block is used only for factory testing. The VH01 board is shipped with a jumper installed from A to B. This jumper must remain in place for normal operation.

Referring again to Figure 3-1, there is a small jumper block containing four jumpers labeled **FL**, **GM**, **HN**, and **JP**. This jumper block is located near the EPROM and MC68000 ICs. This jumper block is used only for factory testing. For normal operation, there must be a connection from F to L, from G to M, from H to N, and from J to P. These connections can be made either by jumpers installed on the jumper pins, or by uncut etch traces on the printed circuit board.

3.4.4.3 SCSI Termination Power Jumper

Referring to Figure 3-1, there is a small jumper block labeled **CD** near connector J2. With a jumper installed from C to D (the default setting), the VH01 will provide termination power to the SCSI bus.

On the SCSI bus, normally two SCSI devices, one at each end of the bus, provide termination power, while intermediate devices on the bus (if any) do not provide termination power. Thus, you should determine for your installation whether or not the VH01 will be at the end of the SCSI bus, and install or remove the jumper at CD accordingly.

3.4.4.4 Floppy Jumper

Referring to Figure 3-1, there is a small jumper block labeled **RST** near the 82072 floppy IC. With a jumper installed from R to S, the VH01 will sense the floppy drive ready signal on pin 2 of floppy port connector J1. With a jumper from S to T, the VH01 will sense the ready signal on pin 34. The default setting for this jumper is from S to T (pin 34), and the VH01 board is shipped with this setting. If you plan to use the floppy port on the VH01, you must refer to the documentation for the floppy disk drives to determine which pin on the drive cable provides the ready signal.

Most newer floppy drives can provide both the DISKETTE READY signal and the DISKETTE CHANGE signal. These drives usually provide a jumper for selecting which signal will be output. For operation with the VH01 floppy port, these drives *must* be jumpered to output the DISKETTE READY signal.

3.5 Installing the VH01 In a Backplane

Use the following procedure to install the VH01 in a backplane:

1. Ensure that the VH01 is properly configured for your system (see subsection 3.4).

2. Shut down the operating system and **REMOVE THE AC POWER.**

3. Remove the covers from the CPU cabinet to make the VMEbus backplane accessible.

4. Select a card slot for the VH01. The card cage must have a slot at the proper DMA priority available for the VH01. The VH01 uses DMA to transfer data. The location of the VH01 in the DMA priority chain could be critical. The amount of bus bandwidth that the VH01 uses will be high at times; this might affect other boards in the system. Likewise, other boards might not allow enough time for the VH01 to DMA enough data to keep up with its peripherals; consider this when choosing a slot. If the VH01's priority is too low, then its DMA will fall behind what its peripherals require. If the VH01's priority is too high, then it receives enough DMA time, but other boards having insufficient buffers might "starve" from lack of DMA time. The priorities must be balanced for the VMEbus system to work properly. Other factors that will affect VMEbus performance are the burst length and ROR/RWD options on the VH01.

5. Remove any jumpers that are installed for the slot that you have selected (for example, bus grant and interrupt acknowledge jumpers).

6. Install the VH01 board by firmly plugging it into the backplane. Orient the components on the board in the same direction as the CPU and other modules.

7. Connect cables using the information in subsection 3.6.

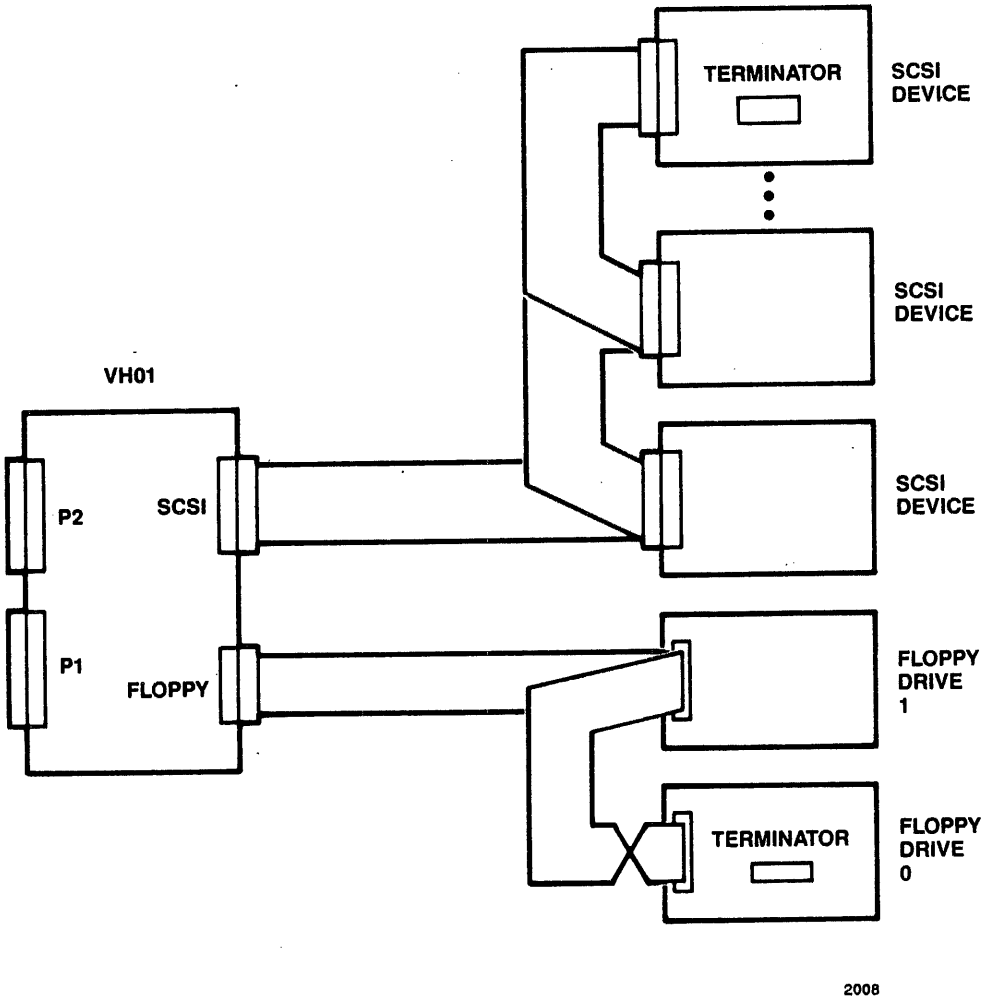
3.6 Cabling

Figure 3-4 illustrates a typical cabling arrangement between the VH01 and its controlled devices.

The VH01 Host Adapter interfaces to a SCSI device via a 50-conductor ribbon cable that connects to the VH01 at the connector labeled SCSI on the VH01 front panel. If more than one SCSI device is to be used, then the remaining devices must be connected together in daisy-chain fashion. In all cases, the last device at each end of the SCSI bus must provide SCSI bus termination. Terminators, if present, must be removed or disabled on all devices except the two end devices. The maximum cumulative length for the SCSI bus cabling is 6 meters (19 feet 8 inches) for the single-ended version and 25 meters (82 feet) for the differential version.

If the optional floppy port on the VH01 is to be used, the VH01 connects to the first (or only) SA450-compatible floppy disk drive via a 34-conductor SA450 ribbon cable. This cable originates from the connector labeled FLOPPY on the VH01 front panel. If only one SA450-compatible drive is to be used, it must have a floppy terminator installed on it. If two SA450-compatible drives are to be used, the drives must be connected to the VH01's floppy port with a 34-conductor IBM-style cable. This cable must have a half twist in the section of cable that connects the two drives, when both drives are internally set to unit 0. When one drive is internally set to unit 0 and the other drive is set to unit 1, a straight cable can be used. When using two drives, a floppy terminator must be installed on the second drive and removed (if present) from the first drive. Consult the floppy disk drive documentation for specific information on the terminator, such as its location on the drive. The maximum cumulative length for the SA450-compatible floppy drive cabling is 3 meters (9 feet 10 inches).

To prevent excessive RFI, some cabinet manufacturers surround their computers with a grounded metal shield. These shields are built into the computer cabinet. Cabling has a direct effect on the amount of electromagnetic interference radiated by a computer system. When installing the VH01 and its controlled devices, you must take steps to preserve the integrity of the shield built into FCC-compliant cabinets. If the VH01 and all of its controlled devices are installed in the same cabinet, then you need only replace the shields that you have removed, to keep the computer compliant with FCC regulations.



NOTE: VH01 SCSI Termination Power Jumper installed.

Figure 3-4. Cabling

3.6.1 Cabling for Same Cabinet Installations

When the VH01 and all of its controlled devices are installed in the same cabinet, it is possible that the cabinet itself provides sufficient shielding to prevent excessive RFI radiation and conduction. In such cases, it is not necessary to shield the cables that connect the VH01 to its peripherals.

NOTE

If the cabinet in which the VH01 and CPU are installed was manufactured before 1 October 1983, it might not provide sufficient shielding or filtering to prevent excessive RFI radiation or conduction. In case of complaint, it is the operator's responsibility to take whatever steps are necessary to correct the interference.

The cables listed below are flat ribbon cables and are not FCC compliant.

NOTE

Emulex does not provide cabling for separate cabinet installations.

Emulex makes unshielded floppy disk drive and SCSI cables in several lengths. Table 3-3 shows standard 50-conductor SCSI cables, while Table 3-4 lists 34-conductor floppy cables of different lengths.

Table 3-3. SCSI Cables

Part Number	Description	Length
MU0111201-01	50 Conductor SCSI Ribbon Cable	3 ft
MU0111201-02	50 Conductor SCSI Ribbon Cable	6 ft
MU0111201-03	50 Conductor SCSI Ribbon Cable	9 ft
MU0111202-04	50 Conductor SCSI Ribbon Cable	12 ft
MU0111202-05	50 Conductor SCSI Ribbon Cable	18 ft

Table 3-4. Floppy Disk Drive Cables

Part Number	Description	Length
QU0111201-01	34 Conductor SA450 Ribbon Cable	3 ft
QU0111201-02	34 Conductor SA450 Ribbon Cable	6 ft

The items listed in Tables 3-3 and 3-4 can be ordered from your Emulex sales representative or directly from the factory. The factory address is:

Emulex Customer Service
3545 Harbor Boulevard
Costa Mesa, CA 92626

(714) 662-5600

TWX (910) 595-2521

To cable the subsystem, see Figure 3-4 and use the following procedures:

3.6.1.1 Floppy Cabling

1. Look at the connectors at the ends of the floppy cable. One end of the cable will have a pin style connector and the other two ends will have edge-connector style connectors. Find the molded-in arrow that identifies pin 1 of each style of connector.

2. The front panel of the VH01 has "FLOPPY" silkscreened directly adjacent to the floppy connector. Pin 1 for the floppy connector is marked with an arrow next to the connector on the VH01 front panel. Align the arrow on the pin-style floppy cable connector with pin 1 of the FLOPPY connector on the board and press the cable connector into the VH01 connector.

3. Find the arrow or pin 1 designation on the first floppy disk drive's edge connector. Align the arrow on the connector near the center of the floppy cable with pin 1 of the first drive's edge connector and press the cable connector onto the drive connector. Make sure that the cable connector is properly seated on the drive connector.

4. Connect the free end of the floppy cable to the edge connector on the second floppy drive supported by the VH01. Note that there will be a half twist in the cable between the two floppy disk drives.

5. Install the floppy terminator (supplied with the drive) on the second floppy disk drive. Refer to the drive documentation for information on the terminator and its location.

3.6.1.2 SCSI Cabling

1. Look at one of the connectors on the SCSI cable. Find the molded-in arrow that identifies pin 1 of the connector.

2. Connect the SCSI cable in daisy-chain fashion to all of the SCSI devices. Be careful to match pin 1 of the SCSI connector on each SCSI device with pin 1 of the corresponding connector on the SCSI cable. The front panel of the VH01 has "SCSI" silkscreened directly adjacent to the SCSI connector. Pin 1 for the VH01's SCSI connector is marked with an arrow next to the connector on the VH01 front panel.

3. For SCSI bus termination requirements, see subsection 3.4.4.3.

3.6.1.3 Grounding

Proper grounding is extremely important for proper operation of the VH01 and its controlled devices. There must be a good connection between each peripheral in the VH01 subsystem and the CPU logic ground. The recommended grounding method is to connect each peripheral to a common point on the CPU chassis using a half-inch or larger braided ground strap, preferably insulated, or an AWG number 10 (or heavier) wire.

Another option, though a less desirable one, is to daisy-chain a ground wire between the peripherals, then connect it to the CPU.

For most applications, connect the peripheral chassis and logic grounds (AC and DC grounds). However, in environments with noise problems, it might work better to separate these two grounds. **Whatever peripheral ground method is used, all peripherals in the VH01 subsystem must be configured the same way.** Consult the manual for each peripheral for specific instructions on connecting the chassis and logic grounds.

Because each subsystem is unique, it is impossible to predict which method will work best for each. Be aware that some experimentation might be required.

NOTE

Failure to observe proper signal grounding methods generally results in marginal operation with random error conditions.

3.7 Power-Up and System Verification

The VH01 has four LEDs that are visible on the front panel, between the SCSI and floppy connectors. Their locations are shown in Figure 3-1. If LED1 is blinking after the VH01 is powered up, then the selftest has completed without errors, the board is functioning properly, and Table 3-5 shows the function of the LEDs. However, if LED1 is not blinking after powerup and the SYSFAIL LED (LED4) is on, then the VH01 detected an error during its selftest. In this case, the LEDs indicate the failed test, as shown in Table 3-6. The VH01 should rarely, if ever, fail its selftest. If it does, however, refer to Appendix A, TROUBLESHOOTING, for information on how to isolate the problem, how to obtain technical assistance from Emulex, and how to return the VH01 to Emulex for service, should this be necessary.

After the selftest has completed successfully, the blink rate of the VH01 Heartbeat LED (LED1) indicates the VH01's relative workload. The blink rate will be slower during periods of greater VH01 activity, and faster during less activity.

Table 3-5. VH01 LEDs After Successful Selftest

LED	Function
1	VH01 Heartbeat
2	SCSI Bus Activity
3	VMEbus Bus Activity
4	VH01 SYSFAIL generation

Table 3-6. VH01 LEDs When Selftest Fails

LED1	LED2	LED3	LED4	Failed Test
1	0	0	1	CPU Test
0	1	0	1	ROM Checksum Test
1	1	0	1	DRAM Test
0	0	1	1	ESP Test
1	0	1	1	Local DMA FIFO Test
0	1	1	1	Floppy IC Test

3.8 Replacing Floppy Disk Drives

If a floppy disk drive connected to the VH01 is removed and replaced with a different drive type, you **must** reset the VH01. The VH01 can be reset either by turning the power to the VH01 board off and back on, or by toggling the VH01 reset switch SW1-1 from off to on and back to off. (Resetting the VH01 is not required when replacing a drive with another drive of the **same** type.)

BLANK

Section 4
VH01 ARCHITECTURE

4.1 Overview

This section contains the following subsections:

Subsection	Title
4.1	Overview
4.2	Format Conventions
4.3	Hardware Description

4.2 Format Conventions

Address and data values in this section are in hexadecimal notation, while bit numbering is in decimal. The diagrams and tables provided show all parameter field values in hexadecimal and bit numbering in decimal.

Except as otherwise noted, the terms "base," "base address," and "BA," which are used throughout this section, all refer to the value set for the VMEbus Base I/O Address of the VH01's VMEbus DRAM Window. The settings of switches SW1-2 through SW1-6 determine this base address. The base address is usually shown with an offset in bytes (Base + X). The offset is in hexadecimal.

The text paragraphs of this section show all hexadecimal values with a leading "0x".

4.3 Hardware Description

Figure 4-1 shows a hardware block diagram of the VH01. The processor section of the VH01 uses a 68000 microprocessor to control all of its functions, and to handle communications with the VMEbus host processor. The VH01's microprocessor can execute its firmware program independently of the data transfers that occur to and from the SCSI bus; however, floppy transfers require direct control by the microprocessor.

The VH01 FIFO section buffers data between the SCSI bus and either the VMEbus or internal DRAM. The VH01 microprocessor transfers data between the VMEbus and internal DRAM. The internal DRAM is used for transferring floppy data and VH01 Command/Response Packets to and from the VMEbus. The FIFO is not used to transfer data between the VMEbus and the internal DRAM.

The VH01 uses a Page Memory to support scatter/gather on the VMEbus.

The SCSI interface is implemented with an Emulex SCSI Processor (ESP) IC, which contains single-ended SCSI drivers. For the differential SCSI interface, differential drivers and receivers on the VH01 board are connected externally to the ESP IC.

The following subsections describe in detail the various components of the VH01 shown in Figure 4-1.

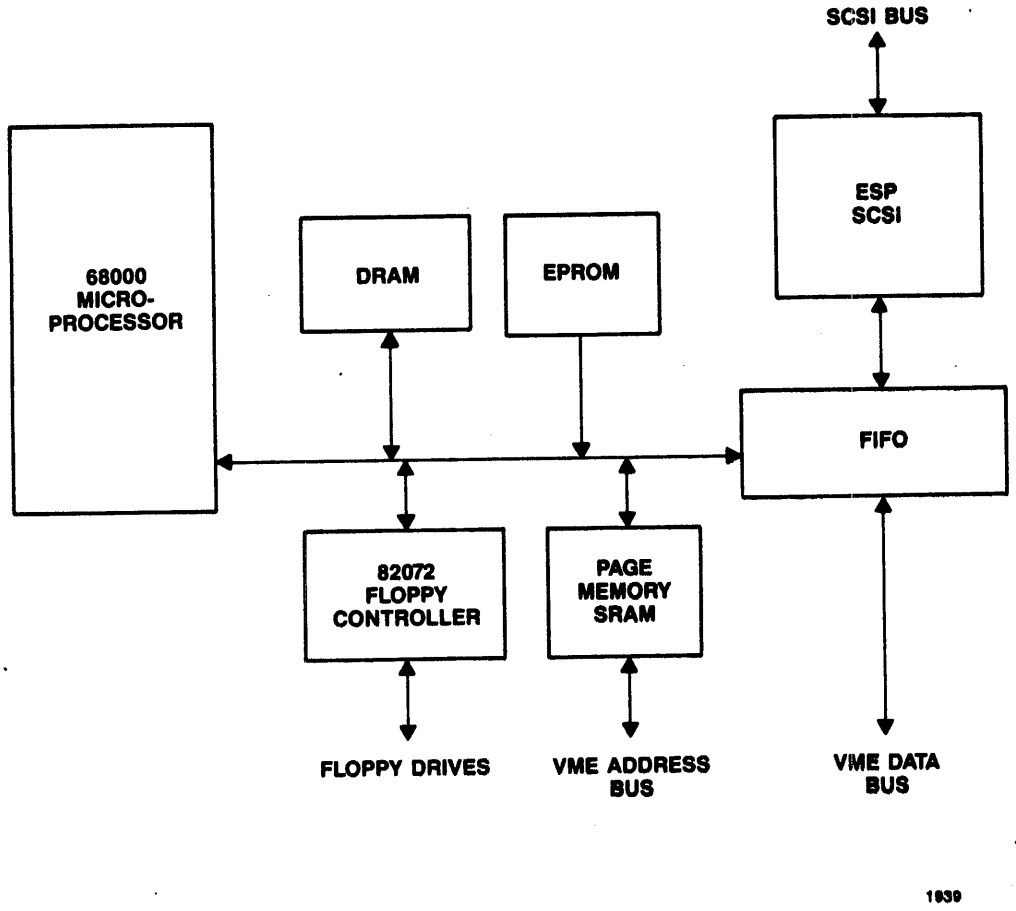


Figure 4-1. VH01 Hardware Block Diagram

4.3.1 Control Processor

The Control Processor, shown in Figure 4-1, is a Motorola 68000 microprocessor running at 12 MHz. Its duties include handling communications with the VMEbus host processor, controlling the SCSI interface, controlling the floppy interface, and instructing the FIFO sequencers. The microprocessor program executes independently from SCSI transfers, unless the destination or source of the transfer is internal VH01 DRAM, in which instance the microprocessor will share the DRAM equally with the FIFO.

4.3.2 DRAM

The Dynamic Random Access Memory (DRAM), shown in Figure 4-1, is composed of eight 64K x 4 bit ICs, yielding a total of 256K bytes. The DRAM contains both the actual executing firmware and general data for the firmware. Initially, after power-up, the VH01 copies the firmware from the EPROM to the DRAM.

DRAM accesses do not require any processor wait states, while EPROM accesses require seven wait states. Thus, using the DRAM for program execution ensures that instruction fetches run with no wait states.

4.3.2.1 VH01 - VMEbus DRAM Window

A 512-byte portion of the 256K byte DRAM is shared with the VMEbus, as a "window" for communications between the VH01 and the VMEbus host processor. The VH01 hardware automatically arbitrates for this DRAM window.

The address of the window in the VMEbus host processor's address space is the 512-byte block beginning at the base address "BA". Switches on the VH01 set the value of BA. See subsection 3.4.3.2 for switch setting information.

4.3.2.2 DRAM VMEbus Interrupt Vectors

The seven interrupts that the VH01 can issue to the VMEbus have seven corresponding 8-bit interrupt vectors. When a VMEbus host processor acknowledges an interrupt from the VH01, the VH01 will place one of these seven vectors onto the VMEbus. The vector indicates to the host which device (in this case the VH01) is responding to the VMEbus interrupt acknowledge cycle. The VH01 microprocessor initializes these vectors after it receives the vector in a Command/Response Packet.

4.3.3 EPROM

The 64K bytes of EPROM, shown in Figure 4-1, contain the firmware for the VH01, which is copied into the DRAM shortly after power-up.

4.3.4 Page Memory

The Page Memory, shown in the block diagram of Figure 4-1, provides the page address during FIFO access to either the VMEbus or to local DRAM, and provides the higher-order address bits when the VH01 microprocessor directly accesses VMEbus memory. The page memory also provides hardware assistance when the VH01 performs scatter/gather operations on the VMEbus.

4.3.5 Real Time Clock

The Real Time Clock (RTC) provides a time base for the VH01 firmware, for time-out functions.

4.3.6 FIFO

The FIFO is divided into two separate sections: the FIFO Input Section and the FIFO Output Section. Each section has its own data bus and control sequencer. Thus, the FIFO can perform simultaneous input and output. Each section can sustain an average 35M bytes per second transfer rate.

The FIFO is composed of two 64 x 8-bit FIFO ICs, arranged in parallel to form a 64 x 16-bit FIFO. Both the FIFO Input Section and the FIFO Output Section use latches and data buffers to allow overlapped operations on the VMEbus and to handle the odd byte starting and ending conditions which can arise on the VMEbus. The FIFO can support either 16-bit or 32-bit VMEbus transfers, and only 16-bit transfers to local DRAM.

4.3.7 Floppy Interface

This subsection describes the interface between the VH01 microprocessor data bus and the floppy disk drives. The floppy drives are controlled entirely by the VH01's 68000 microprocessor, including the actual data transfer. The VMEbus host processor accesses the floppy drives via SCSI-type Command/Response Packets, similar to the SCSI commands used to access hard disk drives on the SCSI bus. Section 5, PROGRAMMING THE VH01, describes the protocol used between the VMEbus host processor and the VH01.

4.3.7.1 Floppy Disk Controller IC

The floppy interface on the VH01 consists of the Intel 82072 Floppy Disk Controller IC, with internal phase-locked loop. This IC handles most floppy interface functions, including serial data stream handling, gap detection and generation, address mark detection, CRC checking, and all timing-critical functions for the floppy drives. Drivers and receivers on the VH01 board are connected externally to the 82072 IC to interface it to the floppy bus.

The 82072 is connected directly to the data bus of the VH01's on-board microprocessor. This means that only the microprocessor can transfer data to and from the 82072. These data transfers can occur simultaneously with a SCSI transfer.

4.3.7.2 Floppy Disk Interface Signals

For the connections between the VH01 and the floppy disk drives, refer to the table for Connector J1 in Appendix B. Note that the signal that the VH01 expects on pin 34 is the drive ready signal, and not the disk change signal. The VH01's floppy port will not function properly unless the ready signal is supplied. This is a jumper-settable option on most drives.

4.3.8 SCSI Interface

The Emulex SCSI Processor (ESP) IC is a VLSI chip designed by Emulex that contains all the electronics to form a single-ended SCSI port. For the differential SCSI option, differential drivers and receivers on the VH01 board are connected externally to the ESP IC. The VH01 has SIP sockets to allow the VH01 to be a SCSI termination node, for the single-ended and differential options.

4.3.8.1 Emulex SCSI Processor (ESP) IC

The ESP IC performs such SCSI functions as arbitration, target selection, and reselection of an initiator. It handles message, command, status, and data transfers between the SCSI bus and its own internal FIFO. The ESP IC is essentially a sophisticated sequencer that consists of two state machines. The high-level state machine manages the disconnect, target, and initiator sequences, and the low-level machine performs the actual interface operations. Unlike most other SCSI processors, the ESP IC does not require the intervention of a microprocessor to manage each of the standard SCSI bus phases (Bus Free, Arbitration, Selection, Reselection, Command, Data, Status, and Message). This results in very low SCSI phase change overhead and high SCSI data transfer rates.

4.3.8.2 SCSI Bus Termination

Both the single-ended and differential versions of the VH01 can optionally terminate the SCSI bus. The single-ended version uses two 220-ohm and two 330-ohm 10-pin SIP resistor packages. The differential version uses four 150-ohm and four 330-ohm 10-pin SIP resistor packages. Tables 4-1 and 4-2 respectively list the location of each of the SIP packages for the single-ended and differential versions of the VH01. Figure 3-1 illustrates the location of the SIP packages on the VH01 board.

Table 4-1. Single-ended SCSI Termination

220 SIP	330 SIP
RM2 RM4	RM1 RM3

Table 4-2. Differential SCSI Termination

150 SIP	330 SIP
RM4 RM6 RM5 RM3	RM2 RM1 RM8 RM7

4.3.8.3 SCSI Interface Signals

For the connections between the VH01 and the SCSI bus, refer to the tables for Connector J2 in Appendix B.

4.3.9 VMEbus Interface

The VH01's VMEbus interface allows either the FIFO or the on-board 68000 to become VMEbus bus master, accessing VMEbus host memory. In addition, the VH01's VMEbus interface responds, as a slave, to a 512-byte range in VMEbus host standard or short space, for communication with the VMEbus host processor.

The VH01 can issue multiple VMEbus interrupts simultaneously, allowing multiple VMEbus processors to communicate with the VH01 simultaneously.

4.3.9.1 VH01 - VMEbus DRAM Window

A 512-byte portion of the VH01's DRAM is mapped as a shared memory space, called the DRAM window. Both the VMEbus host processor and the VH01's microprocessor can access this shared memory. The DRAM window is available to the VMEbus in VMEbus host standard space, using address modifiers 0x39 or 0x3D, or in VMEbus host short space, using address modifiers 0x29 or 0x2D. The DRAM window serves four purposes for the VMEbus host:

1. Message Passing
2. Resetting the VH01
3. Interrupting the VH01
4. Clearing the VH01's VMEbus Interrupts

All four functions are memory-mapped into 512 bytes for access from the VMEbus. Figure 4-2 illustrates the DRAM window from the perspective of the VMEbus.

NOTE

The VMEbus host processor MUST perform only 8-bit wide or 16-bit wide accesses to data in the VMEbus DRAM window. Any longword write to the VH01's DRAM window will cause the VH01 to reset, as if a power-on reset occurred. If the address modifiers are incorrect, the VH01 will NOT respond with the VMEbus DTACK or BERR signals.

4.3.9.1.1 Message Passing (VH01 Message Area)

The first portion of the DRAM window is called the VH01 Message Area. The VH01 and the VMEbus host processor can pass messages between them in the message area, as shown in Figure 4-2. These bytes are read/write from either processor as a byte or word. Subsection 5.5 describes how to pass messages between the VH01 and the VMEbus host processor.

4.3.9.1.2 Resetting the VH01

When a longword read or write is performed to the VH01's DRAM window, the VH01 will reset itself in a manner identical to that following power on. This allows a host processor to reset the VH01 should it need to do so. Once the VH01 has been issued this "command," it will not be ready to perform further host communication until the VH01 completes its selftest, which is approximately 10 seconds.

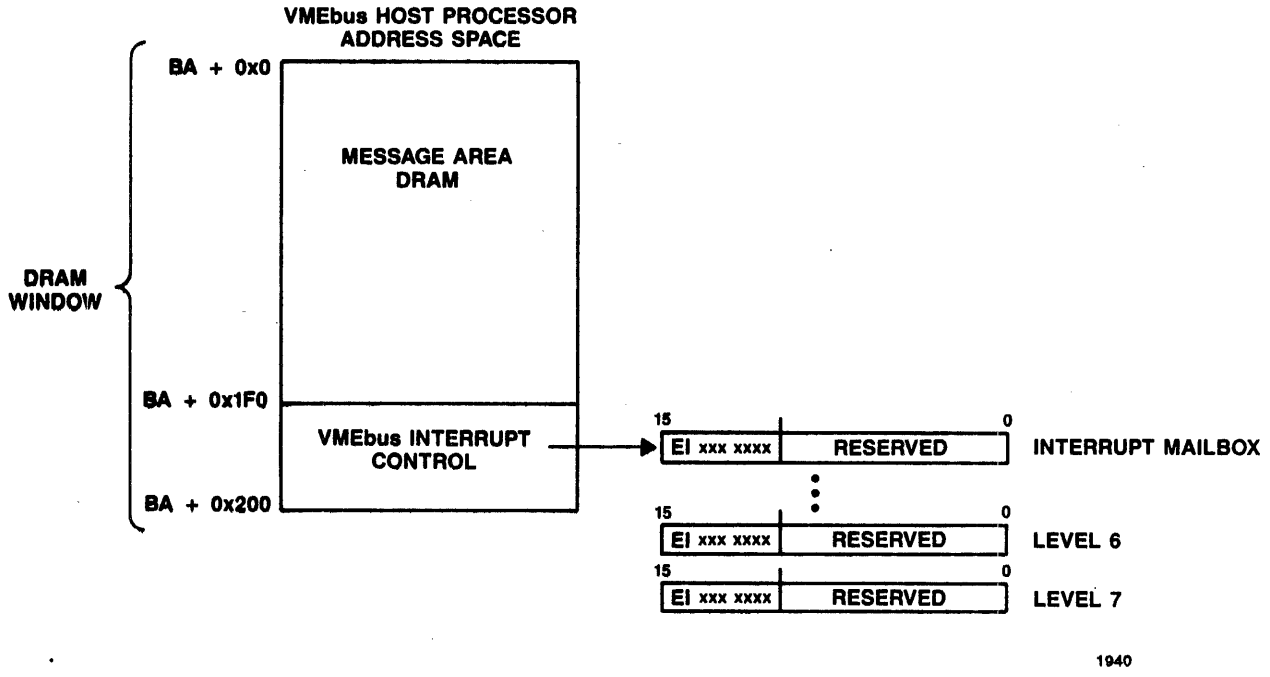


Figure 4-2. DRAM Window

4.3.9.1.3 Interrupting the VH01 (VH01 Interrupt Mailbox)

Word BA + 0x1F0 of the DRAM window is called the VH01 Interrupt Mailbox. The VMEbus host processor must write to this word to interrupt the VH01 microprocessor, in order to signal it that a message awaits in the message area of the DRAM window. Any write to this word, as long as bit 15 is set, will generate an interrupt. Bits 0 to 14 of the word are reserved and should be set to zeros.

4.3.9.1.4 Clearing VMEbus Interrupts

The VMEbus host processor clears a VMEbus interrupt from the VH01 by writing to the appropriate byte in the DRAM window. There are seven words, beginning at BA + 0x1F2, which represent each of the seven VMEbus interrupt levels. Writing a zero to the most-significant byte of the appropriate word will clear the associated interrupt on the VMEbus. The least-significant byte of each of the words is reserved and should never be written. Also, the seven words should never be read, and they will not indicate the status of any interrupts. The host must supply the VH01 with the desired interrupt vector for a particular interrupt level (see subsection 5.6.2).

4.3.9.2 VMEbus Burst Counter

The VMEbus burst counter ensures that the VH01 does not remain on the VMEbus for extended periods of time. The counter can be programmed to allow between 1 and 16 transfers to occur by the FIFO before the VH01 is forced off the VMEbus.

4.3.9.3 VMEbus BLT Option

The VH01 can perform Block Mode Transfers (BLT). BLT will only occur when the FIFO is transferring data on the VMEbus.

4.3.9.4 VMEbus Overlapped Arbitration (Early Release) Option

The VH01 can perform overlapped arbitration (early release) during data transfers on the VMEbus. This will only occur when the VH01 is performing BLT transfers on the VMEbus to or from the FIFO.

4.3.9.5 VMEbus ROR Option

The VH01 can perform Release On Request (ROR) transfers. ROR will only occur when the FIFO is transferring data on the VMEbus.

4.3.9.6 VH01 VMEbus Bus Request

The VH01 can request the VMEbus on one of four DMA priority levels, from 0 to 3. The DMA-priority jumper settings on the VH01 determine which level will be used. Level 3 is the highest priority if the VMEbus Arbiter is a Prioritized Arbiter (PRI Arbiter). If the VMEbus Arbiter is a Round-Robin-Select Arbiter (RRS Arbiter), then all levels have the same priority, since they are serviced in a round-robin fashion. If the VMEbus Arbiter is a Single-Level Arbiter (SGL Arbiter), then only level 3 is used; all other levels are ignored.

For information on setting the DMA-priority jumpers, see subsection 3.4.4.1.

4.3.9.7 VMEbus SYSFAIL* Signal

The VMEbus has a signal called SYSFAIL* that is common to all card slots. Any VMEbus card, including the VH01, can assert SYSFAIL*. The VH01 asserts SYSFAIL* whenever it detects either the VMEbus signal ACFAIL*, or a fatal hardware error. (See the description of the BERR* signal below.) Also, when the VH01 is asserting SYSFAIL*, it will light the SYSFAIL LED on the VH01 front panel.

4.3.9.8 VMEbus ACFAIL* Signal

The VMEbus has a signal called ACFAIL* that is common to all card slots. The VMEbus power supply asserts ACFAIL* whenever it detects that the AC line voltage has dropped to below 70% of its nominal value. This indicates that the DC voltage on the VMEbus will soon be below tolerance. When the VH01 detects ACFAIL*, it resets the SCSI bus, asserts SYSFAIL*, and then stops the VH01 microprocessor.

4.3.9.9 VMEbus Bus Error (BERR*) Signal

When the VH01 is a VMEbus bus master, which occurs when the FIFO is transferring data or when the VH01 microprocessor is accessing VMEbus memory, it might receive a BERR* signal from either the VMEbus slave memory module or from the VMEbus Bus Timer. Such an indication means that the present transfer is unsuccessful and should be aborted.

When the VH01 detects the BERR* signal, its actions will depend upon the nature of the transfer, as listed in Table 4-3.

Table 4-3. VH01 Bus Error Actions

Cause of Bus Error	Resulting VH01 Action(s)
FIFO transfer.	VH01 resets the SCSI Bus; returns a response of "Host Bus Error" to host.
Programmed transfer to Command/Response Packet; packet does not exist.	No response issued.
Packet exists but Response List Header does not exist.	No response issued. Command/Response Packet indicates that command is complete.
Packet and Response List Header exist, but fault occurred accessing the response list.	VH01 resets the SCSI Bus, then asserts SYSFAIL*.
Programmed transfer on data transfer for floppy.	VH01 returns a response of "Host Bus Error" to host.

4.3.9.10 User-defined External VMEbus Outputs

The VH01's VMEbus P2 connector provides seven output signals, EXT 0 through EXT 6. These eight outputs connect directly to the outputs of a register on the VH01, called the External Control Register. By sending commands to the VH01, the host can set the External Control Register bits to any desired value. Section 5, PROGRAMMING THE VH01, describes how to program the External Control Register.

4.3.9.11 VMEbus Interface Signals

For the connections between the VH01 and the VMEbus, refer to the tables for Connectors P1 and P2 in Appendix B.

5.1 Overview

This section contains the following subsections:

Subsection	Title
5.1	Overview
5.2	Format Conventions
5.3	Initialization
5.4	Command Execution
5.5	VH01 Message Area
5.6	Host Memory Data Structures

5.2 Format Conventions

Address and data values in this section are in hexadecimal notation, while bit numbering is in decimal. The diagrams and tables provided show all parameter field values in hexadecimal and bit numbering in decimal.

Except as otherwise noted, the terms "base," "base address," and "BA," which are used throughout this section, all refer to the value set for the VMEbus Base I/O Address of the VH01's VMEbus DRAM Window. The settings of switches SW1-2 through SW1-6 determine this base address. The base address is usually shown with an offset in bytes (Base+X). The offset is in hexadecimal.

The text paragraphs of this section show all hexadecimal values with a leading "0x".

Bits shown as "R" are reserved for the exclusive use of the VH01, and should be set to zero by the host processor.

5.3 Initialization

Before the VH01 can begin operation, the host processor must perform two levels of initialization: the host must initialize both the VH01 itself and the target devices. The host uses a command called the Initialize Adapter Command to initialize the VH01. This command is explained later, in subsection 5.5.1.7, "Adapter Commands".

To initialize a SCSI target, the host uses the Initialize Device Command. This command is also described later, in subsection 5.5.2.6, "Queue Commands".

Section 6, APPLICATION INFORMATION, gives additional information on VH01 and target initialization.

5.4 Command Execution

To send a command to the VH01, the host processor must first store the parameters for the command request in VMEbus memory. Next, the host must request exclusive access to one of the command entry queues. After the host has gained access to the queue, the host must modify the queue data, to tell the VH01 where the command data structure is located in VMEbus address space. When all of the information in the command entry queue is valid, the host should interrupt the VH01. The VH01 will then copy the queue information into its internal memory and begin processing the command.

After the command has either completed or timed out, the VH01 will store status information in the host processor's memory space. The VH01 will then interrupt the host processor. The host processor must then examine the status information, to determine what to do next.

Section 6, APPLICATION INFORMATION, gives additional information on VH01 command execution.

5.5 VH01 Message Area

The VH01 Message Area is the lowest-addressed portion of the 512-byte DRAM window, and starts at the VH01's base address (BA). Both the VMEbus host processor and the VH01's microprocessor can access the VH01 Message Area. The host processor uses this memory to send command requests to the VH01. The host can use either byte addressing or word addressing to access the shared memory, but it must not use longword (32-bit) addressing within this block of memory. Although some fields in this block of memory are longwords, they must be accessed either as two 16-bit words, or as four 8-bit bytes. The VH01 Message Area is divided into two sections, as shown in Figure 5-1: the Adapter Section and the Queue Section.

The Adapter Section (Figure 5-2) provides direct communication between the host and the VH01, as opposed to the SCSI bus or floppy port. It allows the host to set parameters (for example, burst size).

The Queue Section (Figure 5-3) contains the command queues. Each queue holds commands that the VH01 is to send to a SCSI controller or floppy drive. There are eight Command Entry Queues (numbered zero through seven).

The following subsections of the manual describe both the Adapter Section and the Queue Section in greater detail.

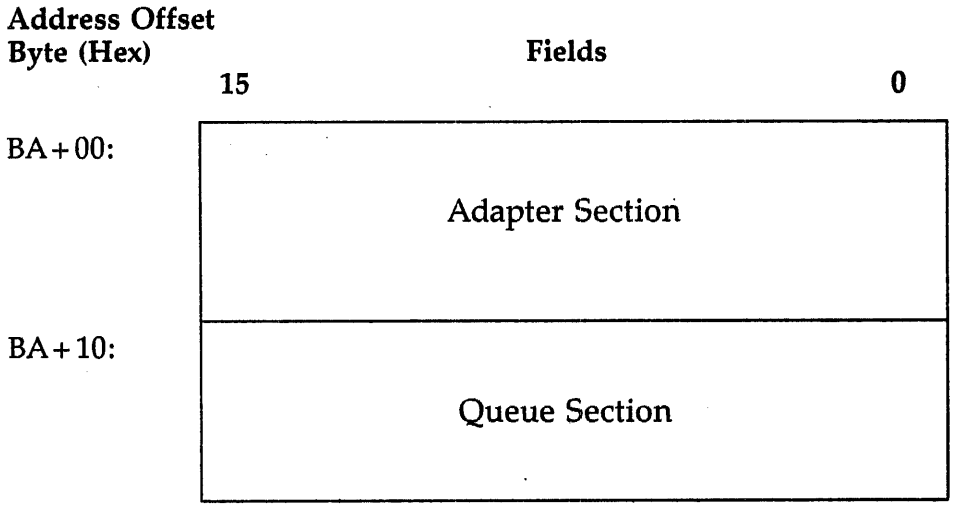


Figure 5-1. VH01 Message Area

NOTE

The VMEbus host processor MUST perform only 8-bit wide or 16-bit wide accesses to data in the VMEbus DRAM window. Any longword write to the VH01's DRAM window will cause the VH01 to reset, as if a power-on reset occurred. If the address modifiers are incorrect, the VH01 will NOT respond with the VMEbus DTACK or BERR signals.

Address Offset Byte (Hex)	Fields 15 — 8	7 — 0
BA+00:	Adapter State Byte	Adapter Command Byte
BA+02:	Parameter Byte 0	Parameter Byte 1
BA+04:	Parameter Byte 2	Parameter Byte 3
BA+06:	VH01 SCSI Bus ID	PROM Desig. (MS Byte)
	PROM Designator	PROM Designator
BA+0A:	PROM Desig. (LS Byte)	Firmware Rev. (MS Byte)
	Firmware Revision	Firmware Rev. (LS Byte)
BA+0E:	Reserved	

Figure 5-2. Adapter Section, VH01 Message Area

Address Offset Byte (Hex)	Fields 15 — 0
BA+10:	Command Entry Queue 0
BA+18:	Command Entry Queue 1
BA+20:	Command Entry Queue 2
BA+28:	Command Entry Queue 3
BA+30:	Command Entry Queue 4
BA+38:	Command Entry Queue 5
BA+40:	Command Entry Queue 6
BA+48:	Command Entry Queue 7

Figure 5-3. Queue Section, VH01 Message Area

5.5.1 Adapter Section

The Adapter Section of the VH01 Message Area provides direct communication between the host and the VH01, as opposed to the SCSI bus or floppy port. It allows the host to set parameters (for example, burst size). The Adapter Section also allows the host processor to determine if the VH01 is capable of accepting commands from the host. In addition, the Adapter Section allows the host to determine the VH01's SCSI Bus ID, PROM revision, and firmware revision.

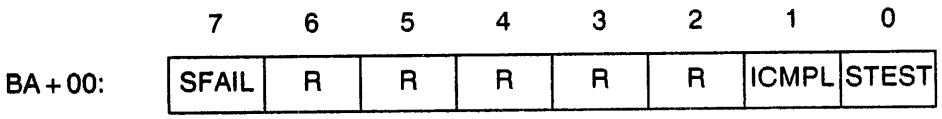
5.5.1.1 Adapter State Byte

The Adapter State Byte serves two purposes, depending on the setting of Bit 7 (SFAIL, the SYSFAIL bit). The host can read the Adapter State Byte at any time, but should treat this byte as "read-only" -- that is, the host should never attempt to alter it.

SYSFAIL (SFAIL)

When Bit 7 of the Adapter State Byte is a one, it means that the VH01 is asserting the SYSFAIL signal on the VMEbus. Under this condition, the bits of the Adapter State Byte must be interpreted according to Table A-3, "VH01 Error Codes," in Appendix A.

When Bit 7 of the Adapter State Byte is a zero, it means that the VH01 is *not* asserting SYSFAIL. Under this condition, the Adapter State Byte allows the host processor to determine if the VH01 is capable of accepting commands from the host. The bits of the Adapter State Byte then have the following interpretations:



Initialization Complete (ICMPL)

If this bit is a one, it means that no errors occurred during selftest, a host processor has initialized the VH01, and the VH01 is ready to accept commands from a host.

Selftest In Progress (STEST)

The VH01 will set this bit to a one before starting its selftest. After the selftest has completed with no errors, the VH01 will set this bit to zero.

5.5.1.2 Adapter Command Byte

	7	6	5	0
BA+01:	OWN	VAL	Command Field	

The Adapter Command Byte allows the host processor to gain exclusive access to the Command Field and Parameter Bytes, and to issue an Adapter Command. The state diagram in Figure 5-4 shows how ownership of the Command Field and Parameter Bytes passes from the free state, to the host, and to the VH01.

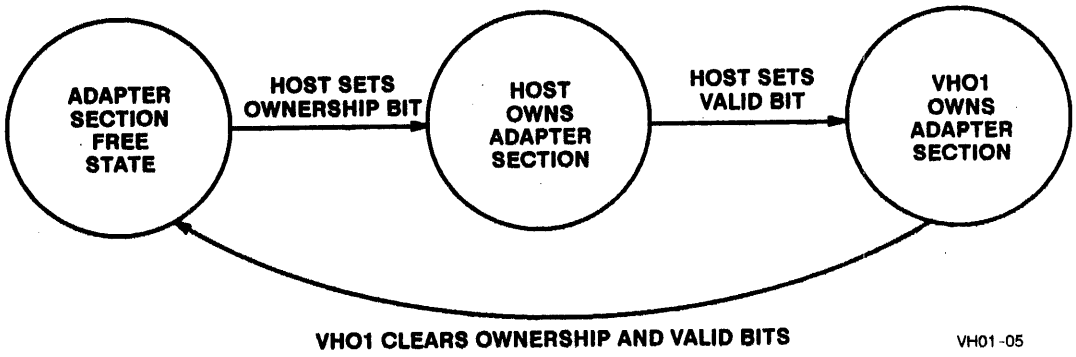


Figure 5-4. Adapter Section Ownership State Diagram

Ownership (OWN)

The Ownership Bit is a semaphore bit that the host uses to gain exclusive access to the Command Field and the adapter Parameter Bytes. The host must set this bit before changing any of these values. After the host has made the changes, the host must set the Valid Bit and then write the value 0x8000 to the VH01 Interrupt Mailbox, to interrupt the VH01.

NOTE

Some VMEbus systems have processors that use memory data caching. On these systems, the processor might need to disable data caching while it competes for ownership.

Valid (VAL)

The host sets the Valid Bit to indicate to the VH01 that the information in the Adapter Section of the VH01 Message Area is valid, and that the VH01 should act on it. After the host fills in the Adapter Section, it must set the Valid Bit to indicate to the VH01 that the information is complete.

No host processor should change any values in the Adapter Section of the Message Area until the VH01 clears both the Ownership Bit and the Valid Bit. When the VH01 has processed the changes in the Adapter Section of the Message Area, it will clear both the Ownership and Valid Bits.

Command Field

The Command Field contains a six-bit Adapter Command. The host places a value in this field, specifying an operation that the VH01 is to perform. Some commands have one or more associated parameters, which the host must place in Parameter Bytes 0 through 3 of the Adapter Section. Each command is described later, in subsection 5.5.1.7, "Adapter Commands".

5.5.1.3 Parameter Bytes

	7							0
BA + 02:	Parameter Byte 0							
	Parameter Byte 1							
	Parameter Byte 2							
	Parameter Byte 3							

The VH01 interprets the Parameter Bytes differently, depending on which Adapter Command the host issued. The Parameter Bytes for each Adapter Command are described later, in subsection 5.5.1.7, "Adapter Commands".

5.5.1.4 VH01 SCSI Bus ID

	7							0
BA + 06:	VH01 SCSI Bus ID							

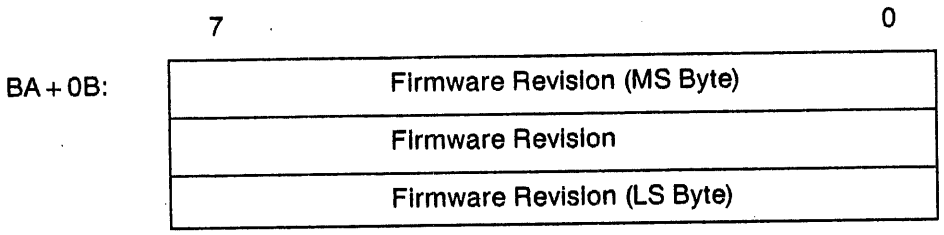
By reading this byte, the host processor can determine the VH01's SCSI Bus ID. Switches on the VH01 board set the SCSI Bus ID. This SCSI Bus ID byte should not be altered; it is only for the host processor's use, and writing a new value will not affect the SCSI Bus ID that is used by the VH01. To set the switches for the SCSI Bus ID, see subsection 3.4.3.3, "VH01 SCSI ID (SW1-7 through SW1-9)".

5.5.1.5 PROM Designator

	7							0
BA + 07:	PROM Designator (MS Byte)							
	PROM Designator							
	PROM Designator							
	PROM Designator (LS Byte)							

The PROM Designator is a group of four ASCII bytes that identify the functionality described in this manual (currently G130 or G142).

5.5.1.6 Firmware Revision



The Firmware Revision is a group of three ASCII bytes that specify the firmware revision.

5.5.1.7 Adapter Commands

Table 5-1 lists the Adapter Commands, along with the value of the six-bit command field and the Parameter Bytes applicable for each command. Where applicable, the Parameter Bytes for each command are described in the following subsections, under the individual description of each command. All command values not listed in the table are reserved; the host should not send these to the VH01.

Table 5-1. Adapter Commands

Command Name	Command Value	Parameter Bytes
Initialize Adapter	0	Byte 0: Burst Control Byte 1: DMA Control Bytes 2 and 3: Not used
Reset SCSI Bus	1	Bytes 0 through 3: Not used
Change External Register	2	Byte 0: Control Register Value Byte 1: Control Register Mask Bytes 2 and 3: Not used
Firmware Download	0X3E	Byte 0: Control Byte 1: Data Bytes 2 and 3: Not used
Firmware Reset	0X3F	Bytes 0 through 3: Not used

5.5.1.7.1 Initialize Adapter (Command 0x0)

If the host specifies this Adapter Command, it indicates to the VH01 that the host has set the values of the burst control and DMA control parameters in the parameter area of the Adapter Section. When the VH01 decodes this command, it will begin using the new values for these two parameters. Parameter Byte 0 specifies the burst control information, and Parameter Byte 1 specifies the DMA control information. Parameter Bytes 2 and 3 are not used with this command.

Following are the descriptions of the Parameter Bytes for the Initialize Adapter Command.

Burst Control Parameter (Parameter Byte 0)

	7	6	5	4	3	0
BA+02:	R	R	R	R	BSIZE	

Burst Size (BSIZE)

The host can specify a value from 1 to 16 for the VMEbus burst length (number of transfers). For a burst size of 16, the host should set the four burst size bits to zero. For burst sizes of 1 through 15, the host should set the desired burst size directly into the four burst size bits.

DMA Control Parameter (Parameter Byte 1)

	7	6	5	4	3	2	1	0
BA+03:	R	R	R	R	R	R	ERLSE	ROR

Early Release (ERLSE)

If the host sets this bit to a one, the VH01 will use early release (overlapped arbitration) on the VMEbus.

Release-On-Request (ROR)

This bit is the Release-On-Request bit. When the host sets this bit to a one, it causes the VH01 to release the VMEbus when it detects another VMEbus bus request (Release-On-Request). When this bit is zero, the VH01 will release the VMEbus when either the Burst Counter expires or when the FIFO is empty (Release-When-Done).

5.5.1.7.2 Reset SCSI Bus (Command 0x1)

If the host specifies this Adapter Command, the VH01 will immediately perform a SCSI bus reset. After the SCSI bus reset, the VH01 will retry all outstanding commands, with the following exceptions:

1. The VH01 will not retry commands having the Disable Retry Bit set to a one in the Command Flags Byte.
2. The VH01 will not retry linked commands, unless it is executing the first command in a group of linked commands.

Parameter Bytes 0 through 3 are not used with this command.

5.5.1.7.3 Change External Control Register (Command 0x2)

If the host specifies this Adapter Command, the VH01 will change the 7-bit External Control Register. The outputs of this register connect directly to seven pins on the P2 VMEbus connector. For the pin connections of the External Control Register on P2, see Appendix B.

Parameter Byte 0 in the VH01 Message Area specifies the control register value, and Parameter Byte 1 specifies a control register mask. Parameter Bytes 2 and 3 are not used with this command.

To change the External Control Register, the VH01 uses the Control Register Value and Control Register Mask parameter bytes, in addition to the old value of the External Control Register, to obtain the new value of the External Control Register as follows:

Temp = Ext_Reg

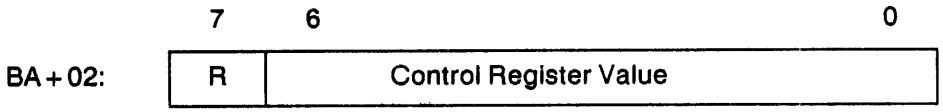
Temp = Temp AND (NOT(Control_Reg_Mask))

Temp = Temp OR (Control_Reg_Value AND Control_Reg_Mask)

Ext_Reg = Temp

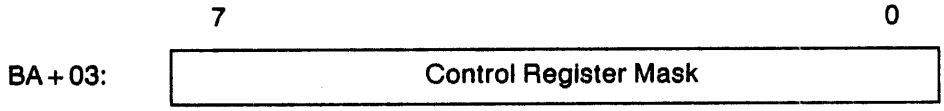
Following are the descriptions of the Parameter Bytes for the Change External Control Register Command.

Control Register Value (Parameter Byte 0)



This is the value of the bits that the VH01 will use to update the External Control Register.

Control Register Mask (Parameter Byte 1)



If the host places a zero in a particular bit of the Control Register Mask, the VH01 will not alter the corresponding bit in the External Control Register. Placing a one in a particular bit of the mask allows the VH01 to change the same bit in the External Control Register.

5.5.1.7.4 Reserved Adapter Commands (Commands 0x3 through 0x3D)

Adapter Commands 0x3 through 0x3D are reserved; the host should not use these values.

5.5.1.7.5 Firmware Download (Command 0x3E)

If this Adapter Command is specified, the host system will download firmware to the VH01's RAM memory, and cause the VH01 to execute the new firmware instead of the onboard firmware.

Parameter Byte 0 is used as a handshake byte to synchronize the data transfer between the host system and the VH01. Parameter Byte 1 is the actual data transferred. Parameter Bytes 2 and 3 are not used.

Following are descriptions of the parameter bytes for the Firmware Download Command:

Handshake Parameter (Parameter Byte 0)

7	6	5	4	3	2	1	0
H	H	H	H	H	H	H	H

Valid handshake byte values are:

1. 0x00 - No data is present in the data byte.
2. 0x01 - Valid data is present in the data byte.
3. 0xFF - Host sets the FF value to indicate the end of download.

Data Parameter (Parameter Byte 1)

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

This byte is used to download firmware from the host to the VH01. Listing 5-1 shows an example of a firmware download routine, in the C programming language:

```
Ptr          = Download_Code_Buffer_Address;
Xfer_Cnt     = Length_of_Download;
while (Xfer_Cnt)
{
  Adapter.Data_Byte = *Ptr++;
  Adapter.Handshake = 1;
  while (Adapter.Handshake)
  {
            /* WAIT FOR VH01 TO TAKE DATA */
  }
}
Adapter.Handshake = 0xFF;
```

Listing 5-1. Firmware Download Example

NOTE

The program fragment shown above should be regarded as an example only, not as a complete program.

Table 5-2, following, shows the basic structure of the RAM-resident jump table, including ROM-based functions that are callable from RAM. This table is loaded into RAM from ROM and begins at location 0x100000.

The user-downloadable firmware must start with the local copy of the jump table at address 0x100018. The jump table is used by the firmware to get from RAM to common ROM-based routines, and back again. The jump table begins with a jump instruction to the new Main routine, and the rest of the firmware code follows the jump table.

Table 5-2. RAM-Resident Jump Table

Address	Function	Comments
0x100000	**Firmware_Reset()** Firmware_Reset: jump.1 VH01_main	Refer to Section 5.5.1.7.6
0x100006	**Firmware_Download()** Firmware_Download: jmp.1 DLoadFW	Refer to Section 5.5.1.7.5
0x10000C	**Halt_VH01_Without_LEDS()** Halt_VH01_Without_LEDS: jmp.1 Stop_VH01	Refer to Section 5.5.1.7.5
0x100012	**Halt_VH01_With_LEDS(Error_Code)**	Halts the VH01 firmware and displays a hex byte error code as 3 octal digits starting with the most significant digit (MSD). The uppermost bit is ignored. To get here from 'C', call: `SysFail_Die(Fail_Reason)`[1] This in turn calls: `Halt_VH01_With_LEDS:` `jmp StopWLed`[2]
0x100018	**VH01_Strt:** starts the Main routine jmp.1 Firmwares_main_routine	Start of downloadable jump table.
0x10001E	**Bus_error:** reports an error in the bus jmp.1 Bus_Error_Handler	
	Auto Vector Interrupt Handlers	
0x100024	ISPL1: jmp.1 SPL1Int	used for internal Fifo status
0x10002A	ISPL2: jmp.1 SPL2Int	used to call host interrupt service routine
0x100030	ISPL3: jmp.1 SPL3Int	used to call SCSI interrupt service routine
0x100036	ISPL4: jmp.1 SPL4Int	used to call floppy interrupt service routine
0x10003C	ISPL5: jmp.1 SPL5Int	used to call timer interrupt service routine
0x100042	ISPL6: jmp.1 SPL6Int	used to call bus error routine (Bus error during DMA)

[1]Fail Reason is the hex byte.
[2]This entry jumps to the ROM-based routine to display the LEDs.

5.5.1.7.6 Firmware Reset (Command 0x3F)

If the host specifies this Adapter Command, the VH01 will reset. When this occurs, the VH01 runs the self-test, uploads ROM code to RAM, and executes the operational firmware. All VH01 adapter parameters must be reloaded after this command is executed. Firmware reset has the same effect as a power-up reset.

Parameter Bytes 0 through 3 are not used with this command.

5.5.2 Command Entry Queues

There are eight Command Entry Queues that are sequential in memory. All eight queues have the same organization. Also, the VH01 treats all of the Command Entry Queues alike. For this reason, the VMEbus host processor can use them in any way that it wants. For some suggested ways to use the eight queues, see the application information in Section 6. Figure 5-5 gives a detailed view of the entire Queue Section, while Figure 5-6 and the following subsections describe one of the Command Entry Queues.

The following subsections of the manual show address offsets within a queue that are relative to $QBase_n$. $QBase_n$ represents the base address of any one of the Command Entry Queues. The subscript n determines a particular queue -- n = 0 for Queue 0, n = 1 for Queue 1, and so on.

As an example, suppose that we wish to find the address of the Device Byte for Queue 3. Figure 5-6 shows that the Device Byte is located at $QBase_n$ + 0x02. Thus, for Queue 3, n = 3, and the Device is at location $QBase_3$ + 0x02. Referring back to Figure 5-3, we see that Queue 3 starts at location BA + 0x28. Thus, for Queue 3, $QBase_3$ would have a value of BA + 0x28. Therefore, the Device Byte is located at address
$QBase_3$ + 0x02 = BA + 0x28+ 0x02 =BA + 0x2A.

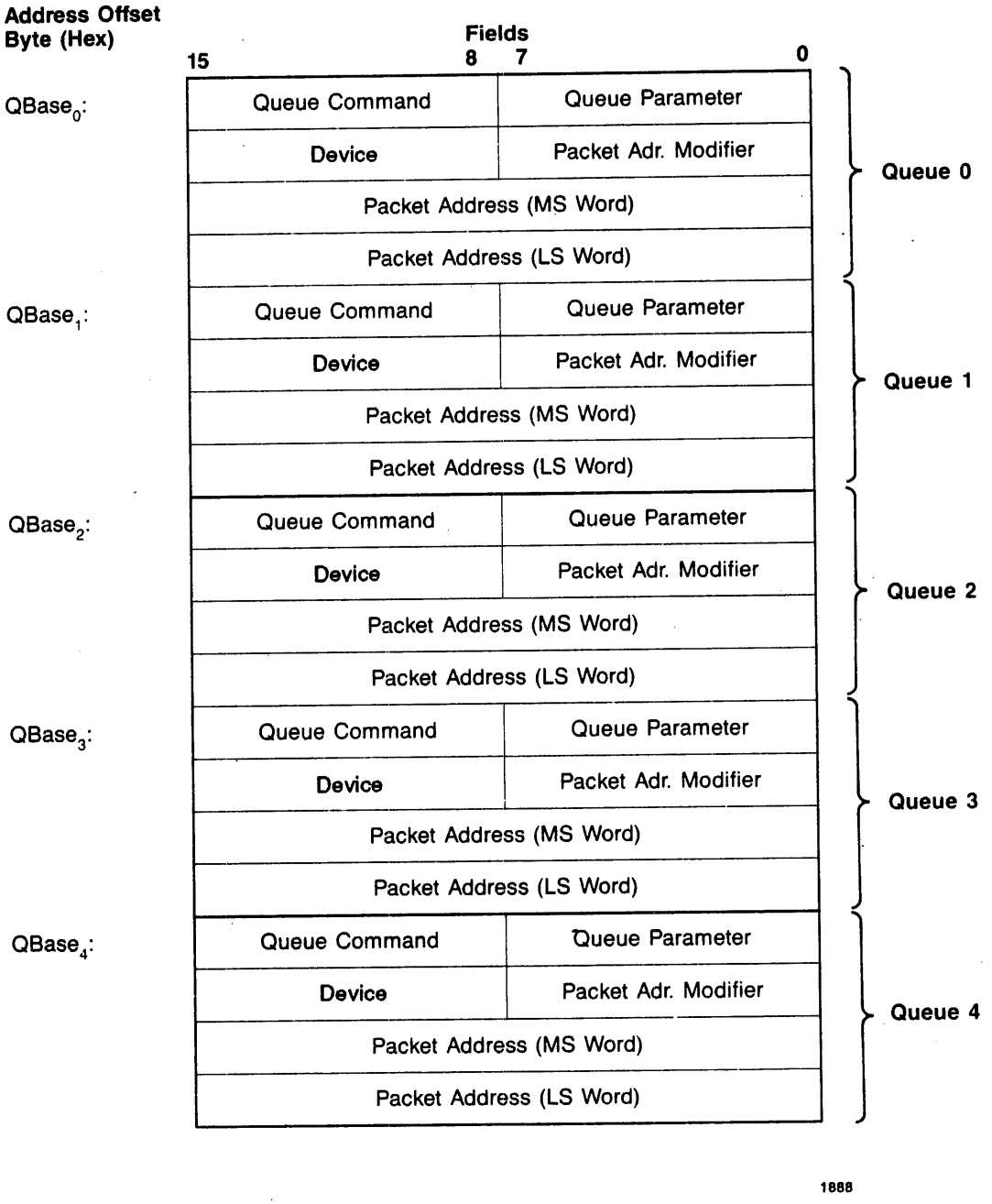


Figure 5-5. Queue Section (Detailed) (page 1 of 2)

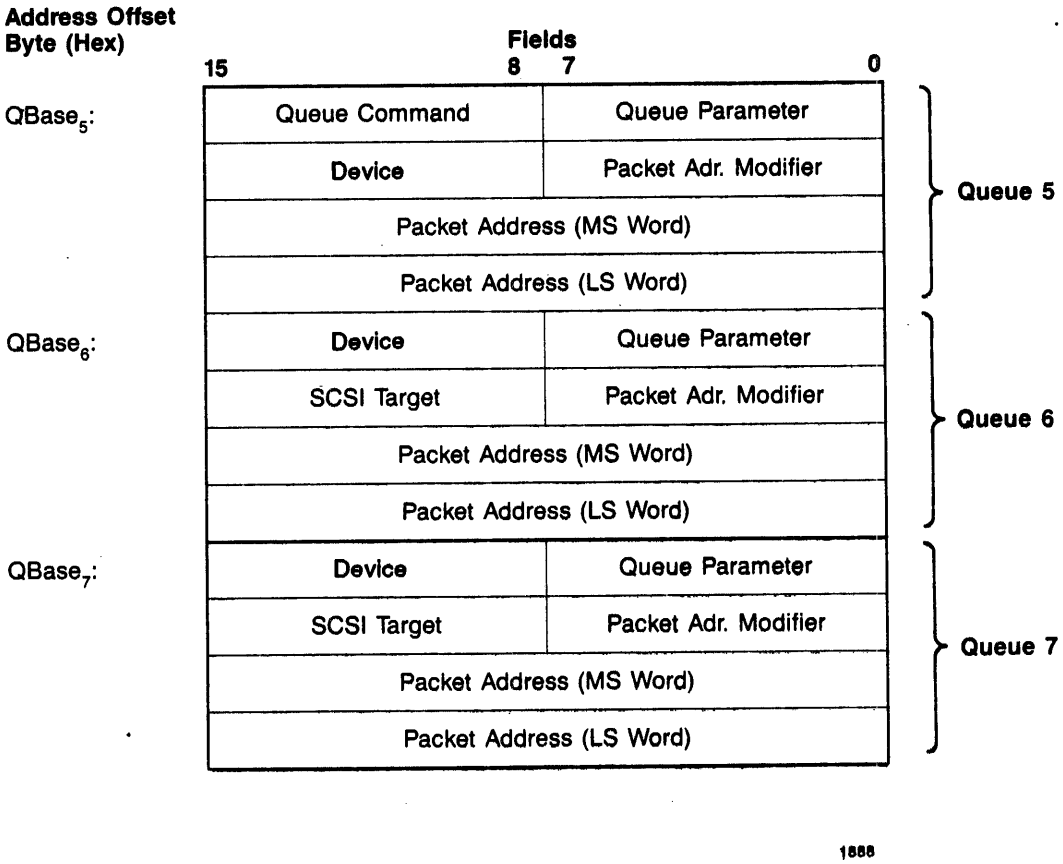


Figure 5-5. Queue Section (Detailed) (page 2 of 2)

Address Offset Byte (Hex)	Fields 15	8 7	0
$QBase_n + 00$:	Queue Command	Queue Parameter	
$QBase_n + 02$:	Device	Packet Adr. Modifier	
$QBase_n + 04$:	Packet Address (MS Word)		
	Packet Address (LS Word)		

Figure 5-6. Command Entry Queue

5.5.2.1 Queue Command

	7	6	5	0
$QBase_n + 00$:	QOWN	QVAL	Command Field	

The Queue Command Byte allows the host processor to gain exclusive access to the Command Field and Queue Parameter of a particular Command Entry Queue, and to issue Queue Commands. The state diagram in Figure 5-7 shows how queue ownership passes from the free state, to the host, and to the VH01.

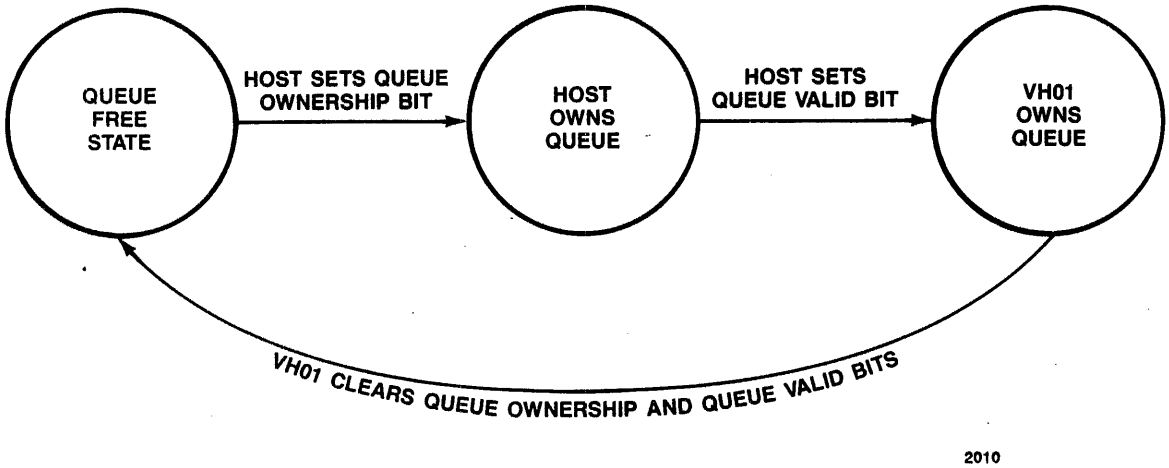


Figure 5-7. Queue Ownership State Diagram

Queue Ownership (QOWN)

The Queue Ownership Bit is a semaphore bit that the host uses to gain exclusive access to a particular Command Entry Queue. The host processor must set this bit before issuing a request. After the host has issued the request, the host must set the Queue Valid Bit and then write the value 0x8000 to the VH01 Interrupt Mailbox, to interrupt the VH01.

NOTE

- Some VMEbus systems have processors that use memory data caching. On these systems, the processor might need to disable data caching while it competes for ownership of a Command Entry Queue.

NOTE

A host can immediately reuse a queue, as soon as the VH01 has cleared that queue's Queue Ownership (QOWN) Bit.

Queue Valid (QVAL)

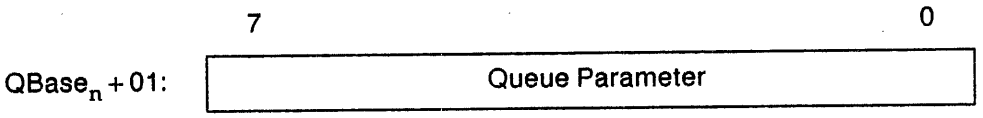
The host sets the Queue Valid Bit to indicate to the VH01 that the Queue Entry information is valid, and that the VH01 should act on it. After the host fills in this Command Entry Queue, it must set the Queue Valid Bit to indicate to the VH01 that the information is complete.

No host processor should change any values corresponding to this Command Entry Queue until the VH01 clears both the Queue Ownership Bit and the Queue Valid Bit. When the VH01 has processed the changes in this Command Entry Queue, it will clear both the Queue Ownership and Queue Valid Bits.

Command Field

The Command Field contains a six-bit Queue Command. The host places a value in this field, specifying an operation that the VH01 is to perform. Some commands have associated parameters, which the host must place in the Queue Parameter byte of the queue. Each command is described later, in subsection 5.5.2.6, "Queue Commands".

5.5.2.2 Queue Parameter



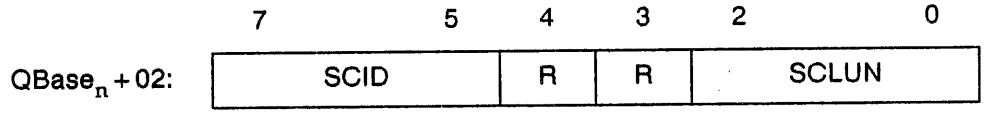
The VH01 interprets this byte differently, depending on which Queue Command the host issued. The Queue Parameter for each Queue Command is described later, in subsection 5.5.2.6, "Queue Commands".

For the floppy port, this byte is unused.

5.5.2.3 Device

The Device Byte specifies the SCSI or floppy device for which the Queue Command is intended.

Device Byte for SCSI Initiator Commands



If the command field in the Queue Command specifies a SCSI initiator command, then the Device Byte contains two fields, the SCSI ID and the SCSI LUN, defined as follows:

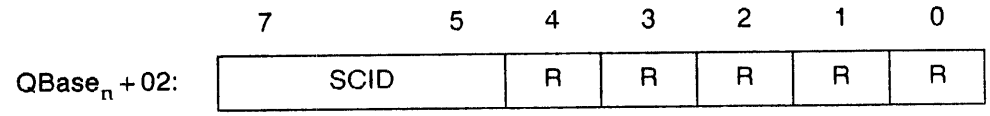
SCSI ID (SCID)

Bits 5 through 7 contain the 3-bit SCSI target ID.

SCSI LUN (SCLUN)

Bits 0 through 2 contain the 3-bit SCSI Logical Unit Number.

Device Byte for SCSI Target Commands



If the command field in the Queue Command specifies a SCSI target command, then bits 5 through 7 of the Device Byte specify the SCSI ID of the SCSI initiator, and all other bits in the Device Byte must be zero. The SCSI ID field is defined as follows:

SCSI ID (SCID)

Bits 5 through 7 contain the 3-bit SCSI initiator ID.

Device Byte for Floppy Commands

	7	6	5	4	3	2	1	0
$QBase_n + 02$:	R	R	R	R	R	R	R	FDRV

If the command field in the Queue Command specifies a floppy command, then bit 0 of the Device Byte selects which floppy disk drive will receive the floppy command, and all other bits in the Device Byte must be zero. The FDRV bit is defined as follows:

Floppy Drive (FDRV)

Bit 0 contains the floppy drive number. A zero selects Floppy Drive 0, and a one selects Floppy Drive 1.

5.5.2.4 Packet Address Modifier

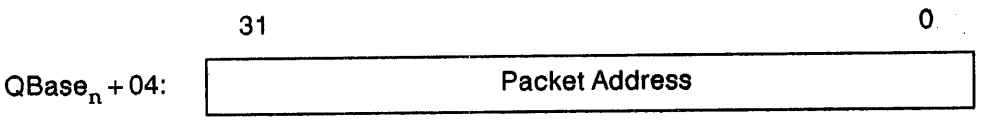
	7	0
$QBase_n + 03$:	Packet Address Modifier	

This is a VMEbus address modifier that the VH01 will use when referencing the Command/Response Packet, the Linked Command/Response Packet, and the Response List Header. Any valid VMEbus address modifier can be used. Table 5-3 lists the defined address modifiers.

Table 5-3. Defined VMEbus Address Modifiers

Hex Code	Function
09	Extended Non-Privileged Data Access
0A	Extended Non-Privileged Program Access
0B	Extended Non-Privileged Block Transfer
0D	Extended Supervisory Data Access
0E	Extended Supervisory Program Access
0F	Extended Supervisory Block Transfer
29	Short Non-Privileged Access
2D	Short Supervisory Access
39	Standard Non-Privileged Data Access
3A	Standard Non-Privileged Program Access
3B	Standard Non-Privileged Block Transfer
3D	Standard Supervisory Data Access
3E	Standard Supervisory Program Access
3F	Standard Supervisory Block Transfer

5.5.2.5 Packet Address



This is the VMEbus address of a Command/Response Packet that the VH01 is to execute. Subsection 5.6.1 describes the Command/Response Packet.

5.5.2.6 Queue Commands

Table 5-4 lists the Queue Commands, along with the value of the six-bit command field, the Queue Parameter, and other values applicable for each command. The Queue Parameter for each command is described in the following subsections, under the individual description of each command. Where applicable, the Device Byte, Packet Address Modifier, and Packet Address function as described previously. All command values not listed in the table are reserved; the host should not send these to the VH01.

Table 5-4. Queue Commands

SCSI Initiator Commands					
Command Name	**Command Value**	**Queue Parameter**	**Target Device**	**Packet Adr. Modifier**	**Packet Address**
Pass-through	0x0	See text	ID and LUN	Mandatory	Mandatory
Initialize Device	0x1	See text	ID and LUN	Optional	Optional[1]
Combine Commands	0x2	Not used	ID and LUN	Not used	Block size[2]
IOPB Cancel; Reset or Abort Device	0x3	See text	ID and LUN	Optional	Optional
SCSI Target Commands[3]					
Command Name	**Command Value**	**Queue Parameter**	**Initiator Device**	**Packet Adr. Modifier**	**Packet Address**
Target Buffer or Target Data	0x10	See text	SCSI ID	Mandatory	Mandatory
Target Mode Initialize Device	0x11	See text	SCSI ID	Not used	Not used
Target IOPB Cancel	0x13	See text	SCSI ID	Optional	Optional
Floppy Commands					
Command Name	**Command Value**	**Queue Parameter**	**Target Device**	**Packet Adr. Modifier**	**Packet Address**
Pass-through	0x20	Not used	Floppy Drive	Mandatory	Mandatory

[1]For the Initialize Device Command, the Packet Address field is used only if the host also issues a packet containing a SCSI Group 0 MODE SELECT Command.

[2]For the Command Combine Command, the host must specify the number of bytes in a block in the Packet Address field, rather than the address of a Command/Response Packet. To disable command combining, the host should zero the Packet Address field.

[3]For an explanation of the SCSI target commands, see subsection 6.10, "Using the VH01 in Target Mode". For a list of the target mode commands that the VH01 supports, see subsection C.1.1, "VH01 Target Mode SCSI Commands," in Appendix C.

5.5.2.6.1 Pass-through (Command 0x0) (SCSI Initiator)

This command allows the host to send a SCSI command to a SCSI device.

Queue Parameter for Pass-through Command

	7	6	5	4	3	2	1	0
$QBase_n + 01$:	R	R	R	R	DRTRY	DARS	DDISC	DMAW

Following are the individual bit descriptions for the Queue Parameter of the Pass-through Command.

Disable Retry (DRTRY)

The host sets this bit to prevent the VH01 from retrying SCSI Unit Attention and Bus Reset conditions. If this bit is set, retries are disabled.

Disable Auto-Request Sense (DARS)

The host sets this bit to prevent the VH01 from issuing a SCSI REQUEST SENSE command on a SCSI Check Condition. If this bit is set and a Check Condition occurs, the VH01 will return a Check Condition internal error.

Disable Disconnects (DDISC)

If the host sets this bit to a one, the VH01 will inhibit the SCSI target from sending a disconnect message.

DMA Width (DMAW)

If the host sets this bit to a one, the VMEbus DMA transfers will be 16 bits wide. If the host sets this bit to a zero, the VMEbus DMA transfers will be 32 bits wide.

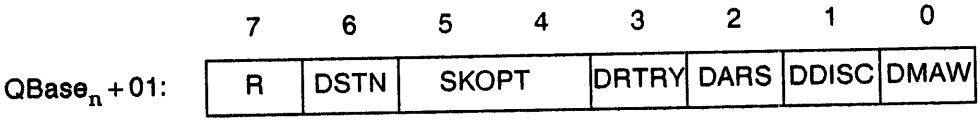
5.5.2.6.2 Initialize Device (Command 0x1) (SCSI Initiator)

The first command that the host sends to each SCSI device must be an Initialize Device command. This command allows the host to specify whether SCSI synchronous data transfer request negotiation will be enabled or disabled for a particular SCSI device. For SCSI disk drives, the Initialize Device command also allows the host to select the type of seek optimization that the VH01 will use.

Additionally, the Initialize Device command allows the host the option of sending a six-byte SCSI Group 0 MODE SELECT command, along with up to 255 bytes of information for the SCSI Data Out Phase. In addition to issuing the MODE SELECT command to the SCSI device, the VH01 will also store the MODE SELECT command and its data and re-send them in the event of a SCSI Bus Reset or Unit Attention condition. This relieves the host from having to re-send the MODE SELECT command whenever a SCSI Bus Reset or Unit Attention occurs. If the host wants to issue an Initialize Device command without mode select information, it must set the Packet Address in the queue to zero.

The host can reissue an Initialize Device command to the VH01 at any time (for example, to change the settings of the DSTN and SKOPT fields) without issuing any further SCSI MODE SELECT commands. In this case, if a SCSI Bus Reset or Unit Attention occurs, the VH01 would use the previously-issued mode select information.

Queue Parameter for Initialize Device Command



Following are the individual bit descriptions for the Queue Parameter of the Initialize Device Command.

Disable Synchronous Transfer Negotiation (DSTN)

The host uses this bit to disable or enable SCSI synchronous data transfer request negotiation during initialization. Negotiation will be disabled if the bit is set, and enabled if the bit is cleared.

Seek Optimization Technique (SKOPT)

The VH01 can optimize head movement on a SCSI disk drive by rearranging the order of the SCSI commands before sending them to the drive. Two bits, the SKOPT bits, allow the host to control the optimization method, or to disable optimization altogether. Table 5-5 shows the functions of the Seek Optimization Technique Bits.

If the host sets both SKOPT bits to zero, then the VH01 will use first in, first out command ordering, and seek optimization is effectively disabled.

If the host sets the SKOPT bits to codes 1, 2, or 3, then the VH01 will use one of the command ordering techniques (closest, sawtooth, or elevator) as shown in Table 5-5. In this case, the VH01 will optimize head movement by reordering all read and write commands that reference nonoverlapping data areas of the media.

Table 5-5. Seek Optimization Codes

Code	Function
0	First in, first out (no optimization)
1	Closest
2	Sawtooth
3	Elevator

CAUTION

Seek optimization should be used with SCSI disk devices only. For all other device types, the host must disable seek optimization by setting both of the SKOPT bits to zero.

NOTE

The VH01 performs seek optimization using an insertion sort that occurs only during periods when the VH01 would otherwise be idle. Thus, seek optimization will never degrade the performance of the SCSI bus.

DRTRY, DARS, DDISC, and DMAW Bits

These bits function the same for the Initialize Device Command as described previously for the Pass-through Command. These bits only affect mode select initialize commands.

5.5.2.6.3 Command Combine (Command 0x2) (SCSI Initiator)

This command allows the VH01 to combine sequential operations of the same type (reads or writes) for the same SCSI device. To enable command combining, the host must specify the number of bytes per block in the Packet Address field, rather than the address of a Command/Response Packet. To disable command combining, the host should set the Packet Address field to zero. Command combining is disabled by default.

The Queue Parameter byte is not used with this command.

NOTE

We recommend that the Command Combine Queue Command be used only with disk devices, otherwise performance could be reduced.

5.5.2.6.4 IOPB Cancel; Reset or Abort Device (Command 0x3) (SCSI Initiator)

The host specifies this Queue Command to cause the VH01 to perform either a SCSI Bus Device Reset or a SCSI Abort. The host must set the Abort or Reset (A/R) bit in the Queue Parameter byte to a one to cause a SCSI Bus Device Reset, or to a zero to cause a SCSI Abort. The actions that the VH01 performs depend upon whether the Packet Address is zero or nonzero, and if nonzero, whether the command in the referenced packet is active or inactive. There are three possible cases, as follows:

Case 1: The Packet Address is nonzero and the command is inactive.

Case 2: The Packet Address is nonzero and the command is active.

Case 3: The Packet Address is zero.

For Case 1, the VH01 will set the Command Aborted Bit in the VH01 Error Word of the packet and move the packet to the response list.

For Case 2, the VH01 will send a SCSI Bus Device Reset message (A/R Bit = 1) or a SCSI Abort message (A/R Bit = 0) to the active SCSI LUN. If the active SCSI LUN is disconnected, the VH01 will wait until either the LUN reconnects or the command times out. If the LUN reconnects before the command times out, the VH01 then will send the Bus Device Reset message or Abort message to the LUN. Finally, the VH01 will set the Command Aborted Bit in the VH01 Error Word of the packet and move the packet to the response list.

For Case 3, the VH01 will perform the same actions as for Case 2, except that it will operate on the **currently-active** packet, instead of the packet at the Packet Address.

Queue Parameter for IOPB Cancel Command

	7	6	5	4	3	2	1	0
$QBase_n + 01$:	R	R	R	R	R	R	R	A/R

The A/R bit in the Queue Parameter for the Reset Device IOPB Cancel Command is defined as follows:

Abort or Reset (A/R)

If the host sets this bit to a one, the VH01 will perform a SCSI Bus Device Reset. If the host sets this bit to a zero, the VH01 will perform a SCSI Abort.

5.5.2.6.5 Target Buffer or Target Data (Command 0x10) (SCSI Target)

This command causes buffers and pointers to be established in host memory so that the VH01 can respond as a SCSI target device to SCSI SEND and RECEIVE commands. This command uses the Queue Parameter byte to determine which type of buffer to create.

Queue Parameter for Target Buffer or Target Data Command

	7	6	5	4	3	2	1	0
$QBase_n + 01$:	R	R	R	R	R	R	D/B	DMAW

Following are the individual bit descriptions for the Queue Parameter of the Target Buffer or Target Data Command.

Buffer Type (D/B)

If the host sets this bit to a one, it specifies target data (a full buffer for the SCSI RECEIVE command). If the host sets this bit to a zero, it specifies a target buffer (an empty buffer for the SCSI SEND command).

DMA Width (DMAW)

If the host sets this bit to a one, the VMEbus DMA transfers will be 16 bits wide. If the host sets this bit to a zero, the VMEbus DMA transfers will be 32 bits wide.

5.5.2.6.6 Target Mode Initialize Device (Command 0x11) (SCSI Target)

The Target Mode Initialize Device command must be executed prior to executing any other target mode commands, and prior to receiving any SCSI commands from another initiator by a VH01 acting as a target. This command allows the host to specify whether SCSI synchronous data transfer request negotiation will be enabled or disabled for a particular SCSI device. The Target Mode Initialize Device command also allows the host to specify the type of Medium Not Available error handling, and whether to support SCSI RECEIVE commands.

Queue Parameter for Target Mode Initialize Device Command

	7	6	5	4	3	2	1	0
$QBase_n + 01$:	R	DSTN	ERR		R	R	REC	R

Following are the individual bit descriptions for the Queue Parameter of the Target Mode Initialize Device Command.

Disable Synchronous Transfer Negotiation (DSTN)

The host uses this bit to disable or enable SCSI synchronous data transfer request negotiation during initialization. Negotiation will be disabled if the bit is set, and enabled if the bit is cleared.

Medium Not Available Error Handling (ERR)

In target mode, a "Medium Not Available" error means that the target has not allocated a buffer (via a target buffer or target data VH01 queue command) corresponding to an initiator's SCSI SEND or SCSI RECEIVE command. Bits 4 and 5 specify the Medium Not Available error handling, as follows:

00 - On error, Check Condition
01 - On error, Busy
10 - On error, Disconnect
11 - Reserved

Receive Command Supported (REC)

The host uses this bit to specify whether or not SCSI RECEIVE commands are supported. A one signifies that both SEND and RECEIVE commands are supported, while a zero signifies that only SEND commands are supported.

NOTE

The SCSI RECEIVE command is optional for a SCSI processor-type device. If the host does not support it, use only the SCSI SEND command, to prevent errors during the SCSI TEST UNIT READY command.

5.5.2.6.7 Target IOPB Cancel (Command 0x13) (SCSI Target)

The host specifies this Queue Command to cancel a buffer created by a previous Target Buffer or Target Data Command. This command uses the Queue Parameter byte to determine which type of buffer command to abort.

Queue Parameter for Target IOPB Cancel Command

	7	6	5	4	3	2	1	0
$QBase_n + 01$:	R	R	R	R	R	R	D/B	R

The D/B bit in the Queue Parameter for the Target IOPB Cancel Command is defined as follows:

Buffer Type (D/B)

If the host sets this bit to a one, it specifies that a Target Data Command is to be aborted. If the host sets this bit to a zero, it specifies that a Target Buffer Command is to be aborted.

5.5.2.6.8 Pass-through (Command 0x20) (Floppy)

This command allows the host to send a floppy command to the VH01's floppy port.

The Queue Parameter byte is not used with this command.

5.5.2.6.9 Reserved Queue Commands

Queue Commands 0x4 through 0x0F, 0x13 through 0x1F, and 0x21 through 0x3F are reserved; the host should not use these values.

5.6 Host Memory Data Structures

The VMEbus host processor must create two different kinds of data structures in host memory: the Command/Response Packet and the Response List Header. The VH01 and the host use Command/Response Packets to transfer commands and status between them; the host uses the Response List Header to keep track of completed packets. Figure 5-8 shows an example of how these data structures relate to each other and to the queues in the VH01 Message Area. The following subsections of the manual describe the Command/Response Packet and Response List Header data structures in detail.

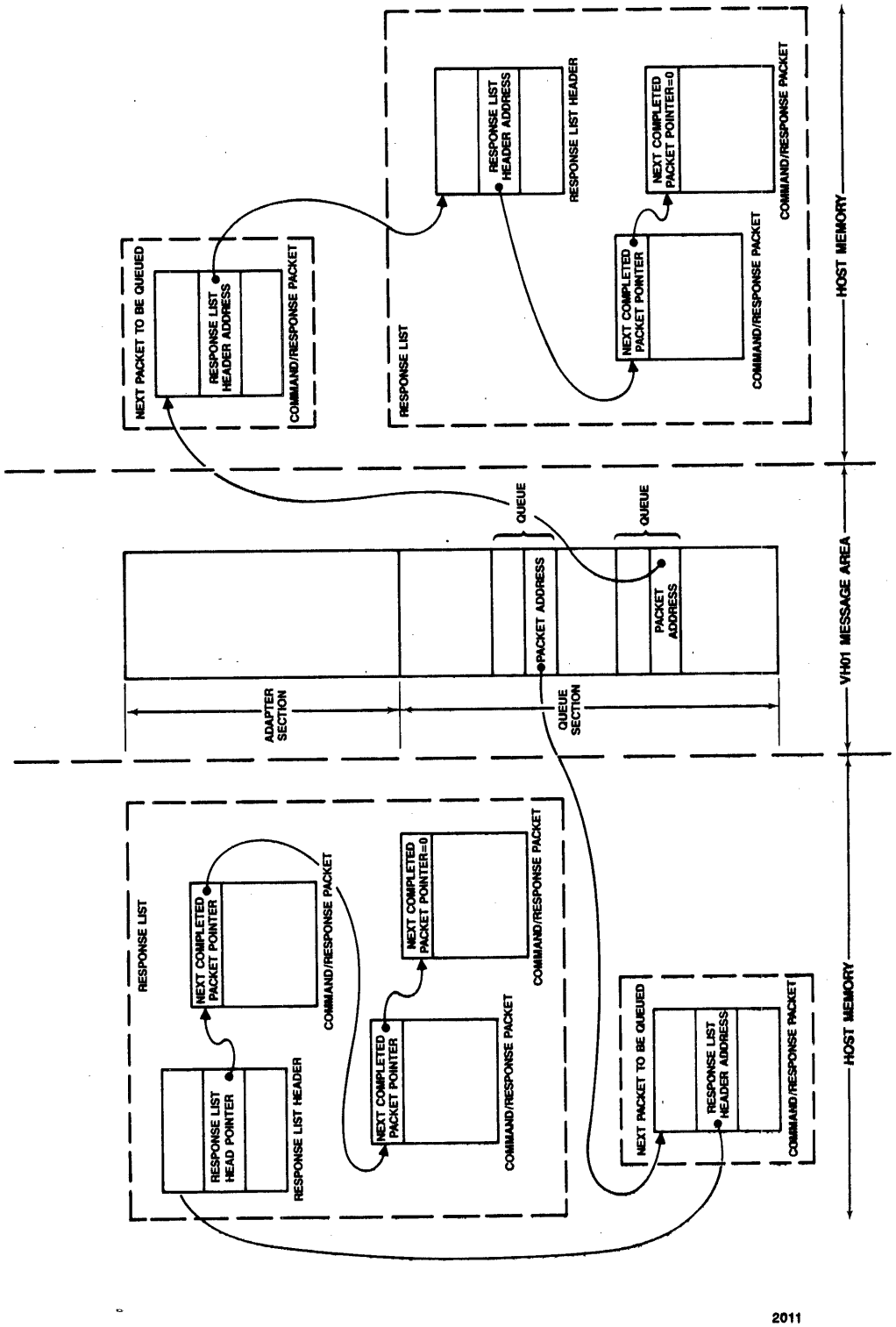


Figure 5-8. Overview of VH01 Data Structures

5.6.1 Command/Response Packet

The Command/Response Packet is a data structure that the host creates in the VMEbus host memory, containing SCSI or floppy commands and responses. The Command/Response Packet has three main sections, as shown in Figure 5-9: the Link Section, the Command Section, and the Response Section.

The VH01 uses the Link Section to link the completed packet into a response linked list.

The host uses the Command Section to send commands and related information to the SCSI or floppy port.

The VH01 uses the Response Section to return error information, SCSI messages, SCSI status, and related information to the host.

Figure 5-10 and the following subsections of the manual describe the Command/Response Packet in greater detail.

NOTE

Command/Response Packets must not reside at address zero.

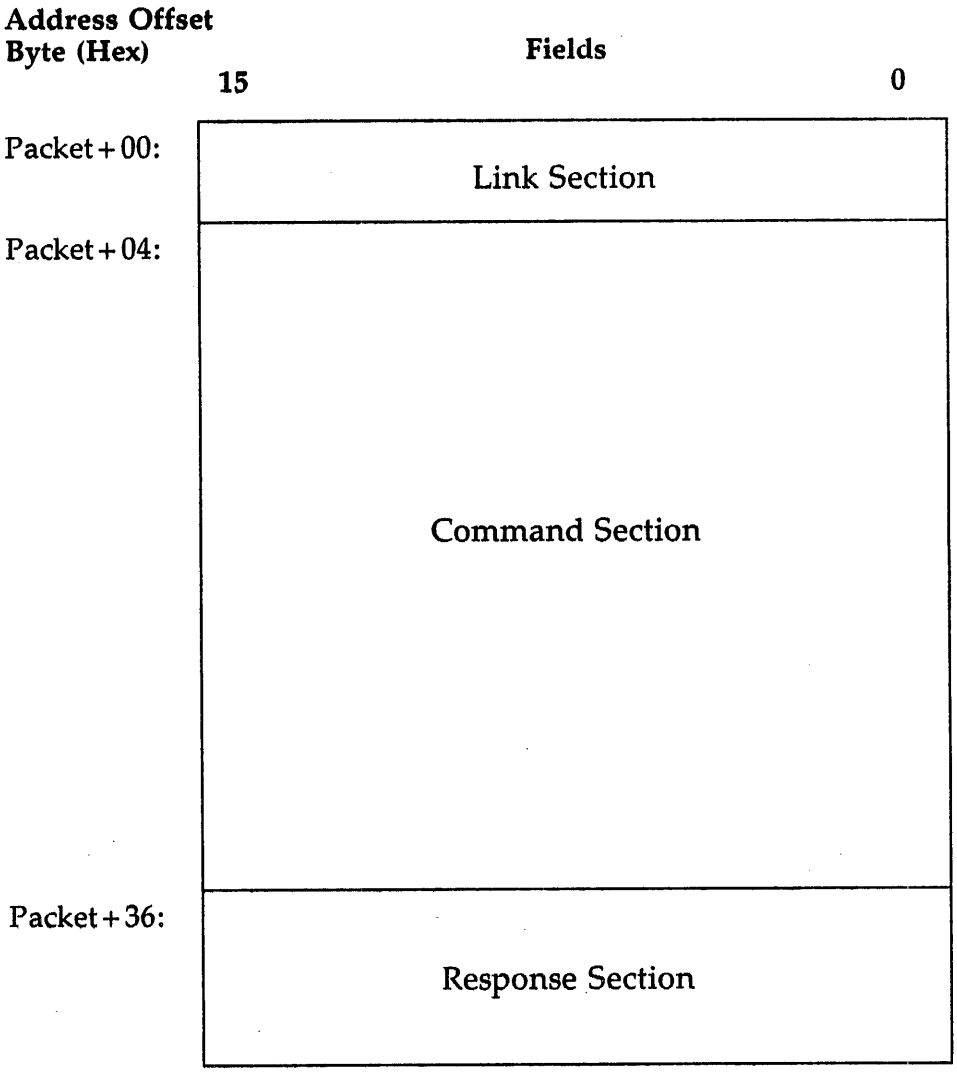
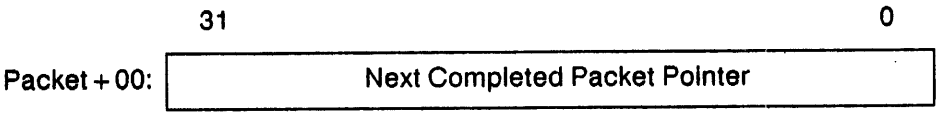


Figure 5-9. Command/Response Packet

Address Offset Byte (Hex)	Fields 15 — 8 7 — 0	
Packet + 00:	Next Completed Packet Pointer (MS Word)	
	Next Completed Packet Pointer (LS Word)	
Packet + 04:	Reference Number (MS Word)	
	Reference Number (LS Word)	
Packet + 08:	Command Timeout Count (MS Word)	
	Command Timeout Count (LS Word)	
Packet + 0C:	SCSI or Floppy Command (12 bytes)	
Packet + 18:	SCSI Linked Packet Address (MS Word)	
	SCSI Linked Packet Address (LS Word)	
Packet + 1C:	Data Transfer Byte Allocation (MS Word)	
	Data Transfer Byte Allocation (LS Word)	
Packet + 20:	Buffer/Scatter-table Address (MS Word)	
	Buffer/Scatter-table Address (LS Word)	
Packet + 24:	Scatter-table Entry Count (MS Word)	
	Scatter-table Entry Count (LS Word)	
Packet + 28:	Buffer Address Modifier	Not Used
Packet + 2A:	Not Used	
Packet + 2C:	Not Used	
	Not Used	
Packet + 30:	Response List Header Address (MS Word)	
	Response List Header Address (LS Word)	
Packet + 34:	Request Sense Byte Allocation	
Packet + 36:	Request Sense Byte Count	
Packet + 38:	Data Transfer Byte Count (MS Word)	
	Data Transfer Byte Count (LS Word)	
Packet + 3C:	VH01 Internally-defined Error Word	
Packet + 3E:	SCSI Last Message Info.	SCSI Last Status Info.
Packet + 40:	SCSI Request Sense Information (255 bytes maximum)	

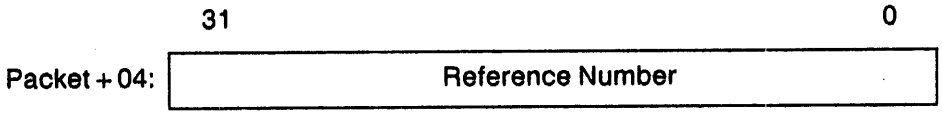
Figure 5-10. Command/Response Packet (Detailed)

5.6.1.1 Next Completed Packet Pointer



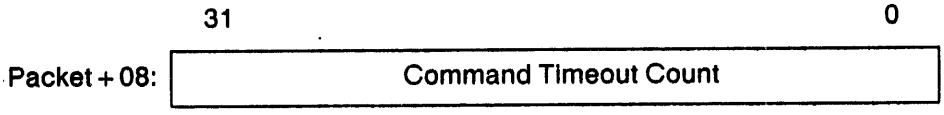
When a Command/Response Packet completes, the VH01 uses this address to link the completed packet into a response list. See subsection 5.6.1.10, "Response List Header Address".

5.6.1.2 Reference Number



The Reference Number is for the host processor. The VH01 will not use this information in any way.

5.6.1.3 Command Timeout Count

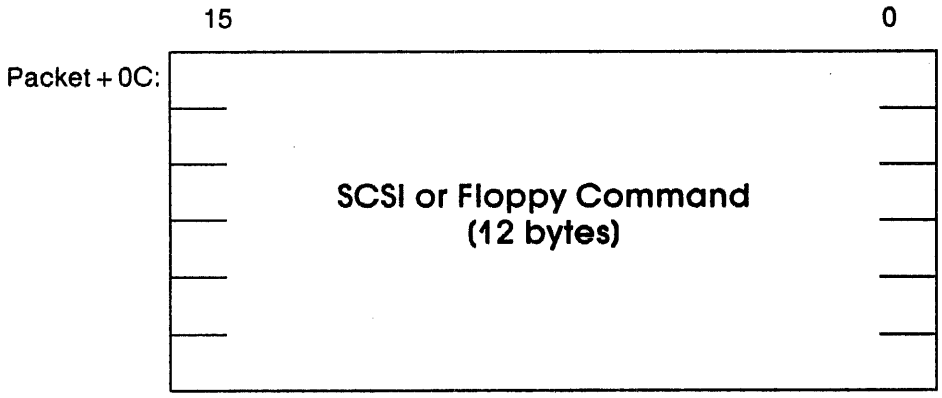


The Command Timeout Count allows the VH01 to detect devices that are not operating as they should. The granularity of this count is 6.8 ms. If the host sets the Command Timeout Count to zero, the VH01 will use a timeout value of approximately 15 seconds. The timeout on a command does not start until the command has been started.

NOTE

The Command Timeout Count is not applicable to the VH01's floppy port.

5.6.1.4 SCSI or Floppy Command



This is the SCSI command for the VH01's SCSI or floppy port. The host must always allocate 12 bytes in the Command/Response Packet for the SCSI command, even though some SCSI commands require less than 12 bytes. In these cases, some of the higher bytes will be unused.

5.6.1.4.1 SCSI Command Set

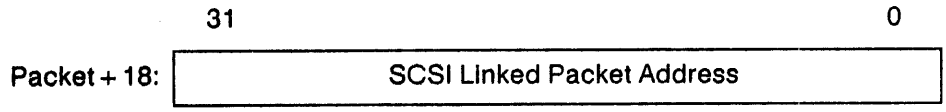
A detailed discussion of all SCSI commands is beyond the scope of this manual. However, Appendix C contains a summary of the SCSI commands that are common to all SCSI device types, as well as the SCSI Message Codes and SCSI Status Byte Codes.

For a detailed description of the SCSI commands, refer to the SCSI standard, as listed in the Preface.

5.6.1.4.2 Floppy Command Set

The floppy interface on the VH01 executes a subset of the SCSI command set. Appendix D gives the commands for the VH01 floppy port. It also details the parameters for the SCSI MODE SELECT command for the floppy port.

5.6.1.5 SCSI Linked Packet Address

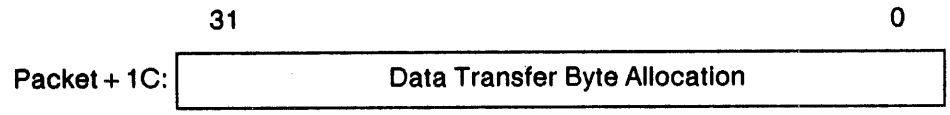


This is the VMEbus address of another Command/Response Packet data structure that is attached for a SCSI Linked Command Option. If the SCSI Link Bit has been set in the last byte of the command, then the host processor must make sure that there is a valid VH01 Command/Response Packet Address here. The VH01's floppy port ignores this address.

When each of the Command/Response Packets has completed, the VH01 will return the packet to the host by placing it on a response list.

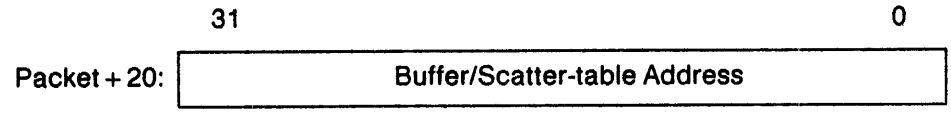
If an error occurs, the VH01 will terminate the links and return the erroneous packet to the host processor. The host must then recover the remaining linked packets in the chain by walking through the chain of linked packets, starting from the erroneous packet.

5.6.1.6 Data Transfer Byte Allocation



The Data Transfer Byte Allocation is the actual maximum number of bytes that can be sent or received.

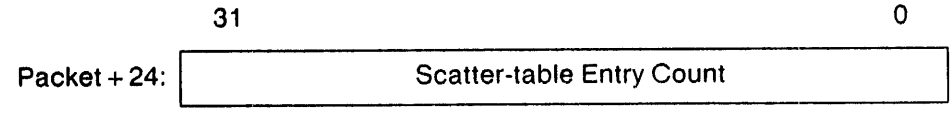
5.6.1.7 Buffer/Scatter-table Address



For sequential data transfers, the Buffer/Scatter-table Address is the starting location of the data to transfer. For scatter/gather transfers, the Buffer/Scatter-table Address is the address of the scatter/gather table.

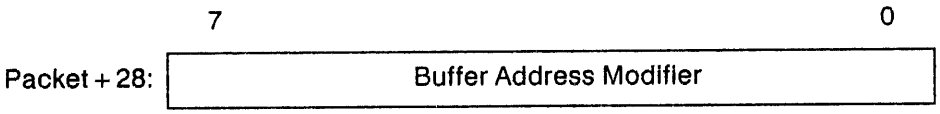
Subsection 5.6.1.18 describes in detail the different types of DMA data transfers that the VH01 can perform.

5.6.1.8 Scatter-table Entry Count



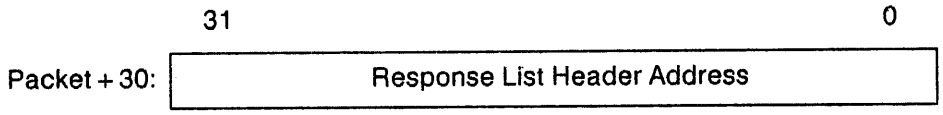
The VH01 uses the Scatter-table Entry Count to determine how many scatter/gather table entries exist for a transfer. A Scatter-table Entry Count of zero indicates a sequential transfer.

5.6.1.9 Buffer Address Modifier



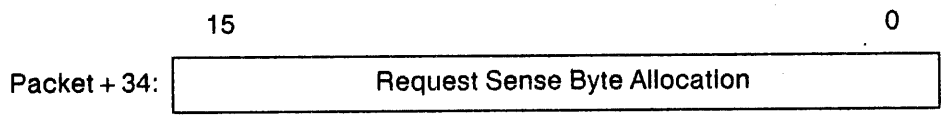
For sequential transfers, the VH01 uses the Buffer Address Modifier as the VMEbus address modifier when it accesses the data. For scatter/gather transfers, the VH01 uses the Buffer Address Modifier as the VMEbus address modifier when it accesses the scatter/gather table entries.

5.6.1.10 Response List Header Address



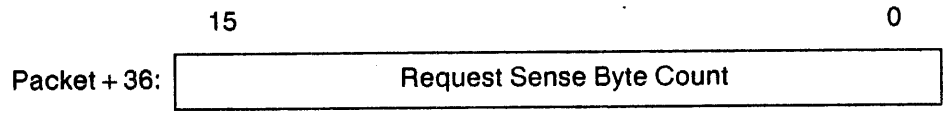
The Response List Header Address is the VMEbus address of the host's data structure that maintains responses (completed Command/Response Packets) as they complete. Subsection 5.6.2 describes the Response List Header in detail.

5.6.1.11 Request Sense Byte Allocation



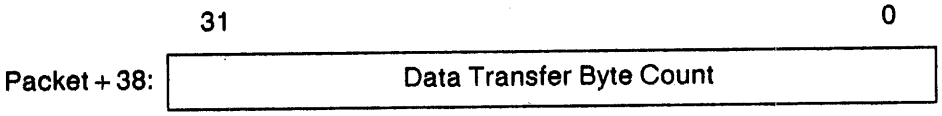
The Request Sense Byte Allocation is the maximum number of bytes that the VH01 can receive for a SCSI REQUEST SENSE command. The allocation is limited to a maximum of 255 bytes.

5.6.1.12 Request Sense Byte Count



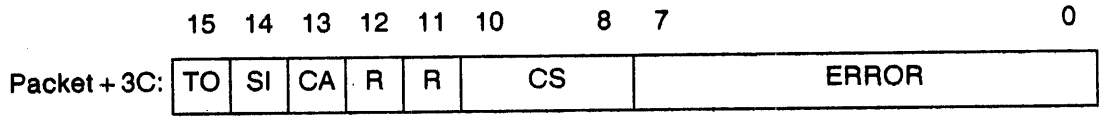
The Request Sense Byte Count is the number of bytes that the SCSI target wanted to send for a SCSI REQUEST SENSE command. If this value is less than or equal to the Request Sense Byte Allocation, then this is the number of bytes that were actually transferred. However, if this number is larger than the allocation, then the VH01 truncated the transfer to the allocation length.

5.6.1.13 Data Transfer Byte Count



The Data Transfer Byte Count is the number of bytes that the SCSI target wanted to transfer during a data phase. If this value is less than or equal to the Data Transfer Byte Allocation, then this is the number of bytes that were actually transferred. However, if this number is larger than the Allocation, then the transfer was truncated to the Allocation length.

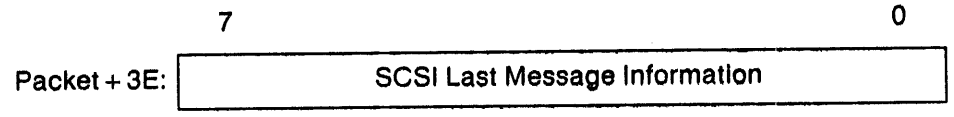
5.6.1.14 VH01 Internally-defined Error Word



The VH01 Internally-defined Error Word is a sixteen-bit location in the Command/Response Packet that contains error information for the host processor. Bits 0 through 7 of the error word form a numeric error code value, bits 8 through 10 give the VH01 processing state of the command in this Command/Response Packet, and bits 13 through 15 of the error word are individual error flags. The error flags can occur individually, in combination with each other, or in combination with the numeric error code value.

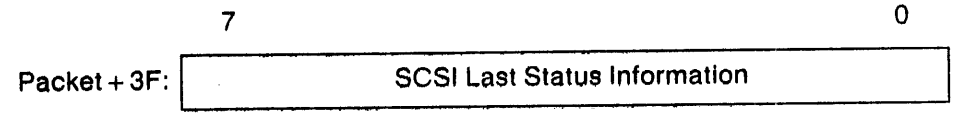
See Appendix E for a detailed description of the numeric error codes, the command state bits, and the error flag bits.

5.6.1.15 SCSI Last Message Information



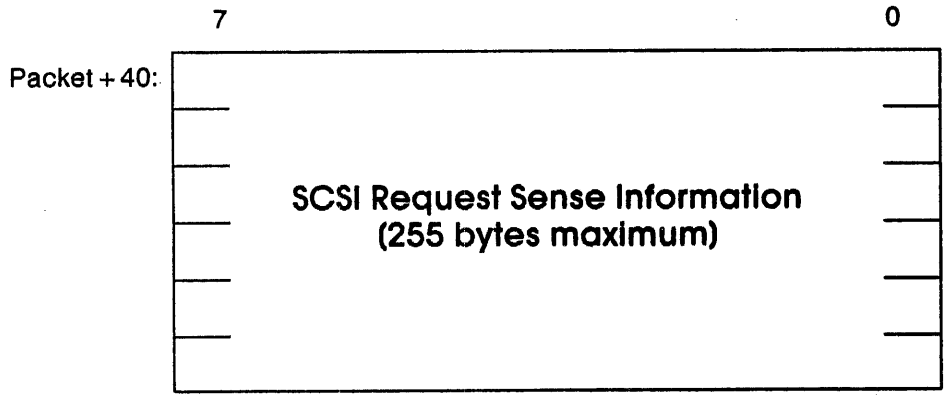
SCSI Last Message Information contains the first byte of the last message that was sent from a selected SCSI target. See Appendix C for the SCSI Message Codes that can be returned.

5.6.1.16 SCSI Last Status Information



SCSI Last Status Information contains the SCSI command completion status byte. See Appendix C for the SCSI Status Byte Codes that the VH01 can return.

5.6.1.17 SCSI Request Sense Information



When a command does not complete and the VH01 Error Word Sense Information Valid (SI) Bit is set, the SCSI Request Sense Information will contain detailed information on the reason. The value of Request Sense Byte Allocation (255 bytes maximum) specifies the length of this data area.

5.6.1.18 VH01 DMA Data Transfers

The VH01 has two methods for transferring data to or from host memory using DMA: sequential transfers and scatter/gather transfers.

A sequential transfer is a simple method that is useful when all of the data is in one contiguous block in host memory.

A scatter/gather transfer provides the maximum memory-management flexibility, allowing the transfer of segments that are both noncontiguous and variable in length.

The following subsections of the manual describe each of these two DMA methods in detail.

5.6.1.18.1 Sequential Transfers

For sequential transfers, the following fields of the Command/Response Packet have significance:

Scatter Table Entry Count

The host must set the Scatter Table Entry Count field to zero, to select a sequential data transfer.

Buffer/Scatter-table Address

The host must place the starting location for the sequential data transfer into the Buffer/Scatter-table Address field. When it accesses the data, the VH01 uses the Buffer Address Modifier as the VMEbus address modifier.

5.6.1.18.2 Scatter/Gather Transfers

For scatter/gather transfers, the following fields of the Command/Response Packet have significance:

Scatter Table Entry Count

The host must set the Scatter Table Entry Count field to a nonzero value, to select a scatter/gather data transfer.

Buffer/Scatter-table Address

The host must place the address of a scatter/gather table into the Buffer/Scatter-table Address field. A scatter/gather table consists of one or more 64-bit entries. As shown in Figure 5-11, each table entry contains a 32-bit segment address, an 8-bit VMEbus address modifier, and a 24-bit segment length. When it accesses the scatter/gather table entries, the VH01 uses the Buffer Address Modifier as the VMEbus address modifier.

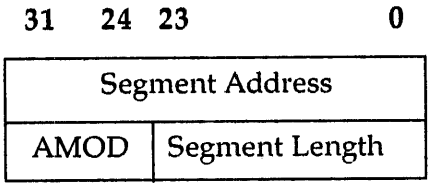


Figure 5-11. Scatter/Gather Table Entry

5.6.2 Response List Header

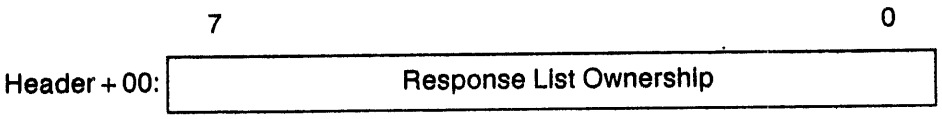
The Response List Header is a data structure that the host must create in the VMEbus host memory, in order to keep track of the completed Command/Response Packets. The Response List Header forms the header, or beginning, of a response list. A response list is a linked list in host memory of completed packets, together with their Response List Header. In this linked list, the Response List Head Pointer in the header points to the start of the first completed Command/Response Packet in the list. The Next Completed Packet Pointer field in the first completed packet points to the second completed packet, the Next Completed Packet Pointer in the second packet points to the third packet, and so on. The Next Completed Packet Pointer in the last completed packet contains zero, to mark the end of the linked list.

If desired, the host can maintain more than one response list, each list having its own Response List Header. This would allow each response list to have its own associated interrupt level and interrupt vector, for example. Figure 5-12 and the following subsections describe the Response List Header.

Address Offset Byte (Hex)	Fields 15 — 8	7 — 0
Header + 00:	Resp. List Ownership	Adapter Altering Resp. List
Header + 02:	Resp. Complete Int. Level	Resp. Cmpl. Int. Vector #
Header + 04:	Response List Head Pointer (MS Word)	
	Response List Head Pointer (LS Word)	
Header + 08:	Reserved (MS Word)	
	Reserved (LS Word)	

Figure 5-12. Response List Header

5.6.2.1 Response List Ownership

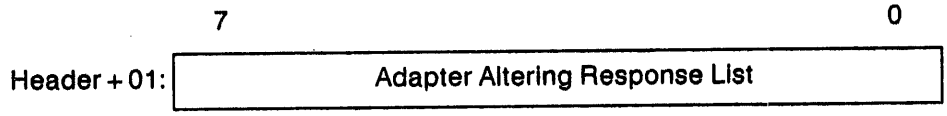


Before a host processor changes the Response List Header, the host must exclusively set the Response List Ownership Byte to 0x80. This is illustrated in the application information in Section 6. When the host initializes this list header, the host must set this byte to zero.

NOTE

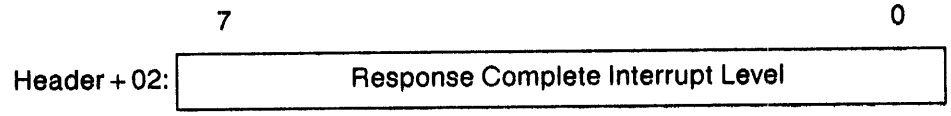
Some VMEbus systems have processors that use memory data caching. On these systems, the processor might need to disable data caching while it competes for ownership of a response list.

5.6.2.2 Adapter Altering Response List



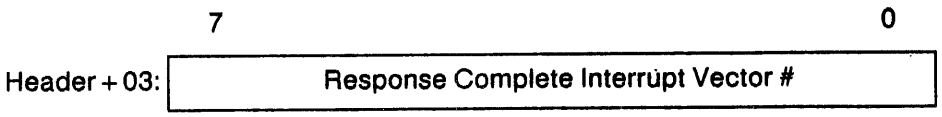
The Adapter Altering Response List will tell the VMEbus host processor that the VH01 has control of the response list. This is illustrated in the application information in Section 6. When the host processor initializes this Response List Header, the host must set this byte to zero.

5.6.2.3 Response Complete Interrupt Level



The Response Complete Interrupt Level is a one-byte location that contains the interrupt request priority level that the VH01 will send to the host when a command completes. If the host sets the Response Complete Interrupt Level to zero, the VH01 will not send an interrupt to the host. Otherwise, the host should set the Response Complete Interrupt Level byte to a value corresponding to the desired interrupt priority level. The VH01 can defer interrupts and send them at its convenience.

5.6.2.4 Response Complete Interrupt Vector Number

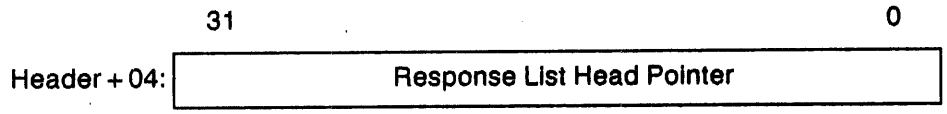


The Response Complete Interrupt Vector Number is a one-byte location that contains the interrupt vector number that the VH01 will send to the host when a command completes. The host must place a value in this location corresponding to the desired interrupt vector number. Later, when the VH01 generates an interrupt to the VMEbus, the VH01 will send this interrupt vector number to the host. The host then uses bits 0 through 7 of this vector number to form bits A2 through A10 respectively, of an address, setting the remaining address bits to zero. Next, the host uses this resulting address as the address of a 4-byte value in the host's vector table. Finally, the host transfers this 4-byte value into its program counter. Thus, the host then begins executing an interrupt processing routine at this new program counter value.

NOTE

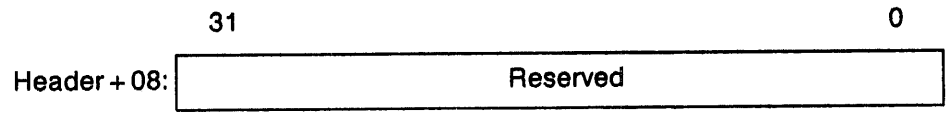
Interrupts are only sent on empty-to-not-empty transitions of the response list.

5.6.2.5 Response List Head Pointer



The Response List Head Pointer is the VMEbus address of the beginning of the completed response list. The host must only remove entries from the beginning of the linked list. When the host initializes the Response List Header, the host must set this address to zero.

5.6.2.6 Reserved



This longword is reserved and should NEVER be altered by a host processor.

Section 6
APPLICATION INFORMATION

6.1 Overview

This section provides information related to writing a program, such as a diagnostic or device driver, to control the VH01. This section contains the following subsections:

Subsection	Title
6.1	Overview
6.2	Initializing the VH01
6.3	Initializing the SCSI Target
6.4	VH01 Command Execution
6.5	Using the Command Entry Queues
6.6	Using the Reference Number Field
6.7	Using Linked SCSI Commands
6.8	Systems with Multiple VMEbus Host Processors
6.9	VMEbus Host Device Driver
6.10	Using the VH01 in Target Mode

6.2 Initializing the VH01

The VMEbus host processor must initialize the VH01 Host Adapter before the VH01 will honor any command requests. The host must not, however, initialize the VH01 until the VH01 self-test has completed successfully. The host can reinitialize the VH01 at any time.

The following list of numbered steps gives the VH01 initialization sequence:

1. The host must first check the Self-test In Progress Bit in the Adapter State Byte to determine if the VH01 is currently performing its internal self-test. If this bit is a one, then the VH01 is still performing its self-test, and the host should not attempt any communication with the VH01. The host should either delay or perform other tasks until this bit is a zero.

2. If the Self-test In Progress Bit is a zero, indicating that the VH01 is not in self-test, the host must then check the Initialization Complete Bit in the Adapter State Byte to determine if another host has previously initialized the VH01. If the Initialization Complete Bit is set and the host does not want to change the VH01's initialized values, then no further initialization steps are necessary.

3. However, if the Initialization Complete Bit is not set, or if the host wants to change the VH01's initialized values, then the host must now gain ownership of the VH01 by setting the Ownership Bit in the Adapter Command Byte in an exclusive manner.

4. The host must then write the new Burst Control value to Parameter Byte 0 and the new DMA Control value to Parameter Byte 1 in the Adapter Section.

5. The host must write an Initialize Adapter Command to the Command Field of the Adapter Command Byte.

6. The host must next set the Valid Bit in the Adapter Command Byte.

7. Finally, the host must write the value 0x8000 to the VH01 Interrupt Mailbox, to interrupt the VH01.

6.3 Initializing the SCSI Target

After the host has initialized the VH01, the host must initialize the SCSI target. The host processor's device driver must do this initialization during the first Device Open operation that it performs on the SCSI target.

The SCSI target initialization serves three purposes:

1. It allows the VH01 to recover from SCSI Unit Attention and SCSI Bus Reset conditions,
2. It allows the VH01 to determine, for each SCSI target, if the host will allow the SCSI synchronous data transfer mode, and
3. It allows the host to specify the seek optimization technique.

The following list of numbered steps gives the SCSI target initialization sequence:

1. To initialize a SCSI target, the host must first gain ownership of a Command Entry Queue by setting the associated Queue Ownership Bit in the Queue Command Byte in an exclusive manner.
2. Once the host gains ownership of the queue, it must then write an Initialize Device Command to the Command Field of the Queue Command Byte.
3. The host now has the option of allowing or disallowing SCSI synchronous data transfer request negotiation. If the host wants to allow negotiation, it must now clear the Disable Synchronous Transfer Negotiation (DSTN) Bit in the Queue Parameter Byte. If the host does not want to allow negotiation, it must now set the DSTN Bit. If the host sets the DSTN Bit, the VH01 will reject all attempts to change the current data transfer mode to synchronous. However, if the VH01 is already in the synchronous data transfer mode, it will stay in this mode until until a condition occurs that returns it to the asynchronous data transfer mode.
4. The host now has the option of sending a six-byte SCSI Group 0 MODE SELECT command, along with up to 255 bytes of information for the SCSI Data Out Phase. To prevent the VH01 from sending a MODE SELECT command in the event of a SCSI Bus Reset or Unit Attention condition, or if a mode select is not desired, the host must now set the Packet Address in the queue to zero, and then skip to Step 7.

5. If the host does want to send a MODE SELECT command, it must now build a Command/Response Packet in host memory containing a six-byte SCSI Group 0 MODE SELECT command, and optionally, up to 255 bytes of information for the SCSI Data Out Phase.

6. The host must now set the Packet Address in the queue to the address of the packet.

7. When the SCSI target initialization information is complete, the host must set the Queue Valid Bit in the Queue Command Byte.

8. Finally, the host must write the value 0x8000 to the VH01 Interrupt Mailbox, to interrupt the VH01.

NOTE

SCSI target initialization does not apply to the VH01's floppy disk port or to the floppy disk drives connected to it.

6.4 VH01 Command Execution

The following list of numbered steps gives a typical sequence for executing a VH01 command:

1. The host must first gain ownership of one of the Command Entry Queues by setting the associated Queue Ownership Bit in the Queue Command Byte in an exclusive manner.

2. Once the host gains ownership of the queue, it should then modify the queue data as required.

3. The host must now build a Command/Response Packet in host memory, containing the SCSI or floppy command.

4. The host must now set the Packet Address in the queue to the address of the packet.

5. When all of the queue information is complete, the host must set the Queue Valid Bit in the Queue Command Byte.

6. Finally, the host must write the value 0x8000 to the VH01 Interrupt Mailbox, to interrupt the VH01.

6.5 Using the Command Entry Queues

The eight Command Entry Queues all have the same organization. Also, the VH01 treats all of the queues alike. For this reason, the VMEbus host processor can use them in any way that it wants. Following are three possible ways to use the eight Command Entry Queues:

1. Use only the first Entry Queue. This method would provide a simple implementation.

2. Use one Entry Queue to represent each of the SCSI target IDs. This method would allow multiple commands to be simultaneously issued for targets with different IDs.

3. Use one Entry Queue for each VMEbus host processor or task using the VH01. This method would provide a simple implementation for multiple host usage.

6.6 Using the Reference Number Field

The Command/Response Packet contains a field called the Reference Number. The VH01 does not use this field; it is available for any purpose that the host desires. Here are some suggestions for possible ways that the host might use this field:

1. Sequential numbering of the packets, in the order that the host sends them to the VH01.

2. Keeping track of which command queue upon which the host places each packet.

3. Keeping track of which response list upon which the VH01 returns each packet.

6.7 Using Linked SCSI Commands

The advantage of using linked SCSI commands is that the host can force the VH01 to process a series of two or more SCSI commands in the desired order. The VH01 will only process a sequence of linked commands in the order that they are linked. Furthermore, the VH01 will process no other commands for the same SCSI peripheral until either the VH01 reaches the end of the linked list, or an error occurs.

For example, if the host wanted to send a SCSI MODE SELECT command to a SCSI disk drive, followed by a SCSI FORMAT UNIT command to that same drive, the drive would need to perform the MODE SELECT command first. To accomplish this, the host must build the command/response packets for both the MODE SELECT command and the FORMAT UNIT command before sending either packet to the command queue. When the host builds the packet for the MODE SELECT command, it must set the SCSI Link Bit in the last byte of the command. The host must also place the address of the FORMAT UNIT packet into the SCSI Linked Packet Address field of the MODE SELECT packet. When the host builds the FORMAT UNIT packet, it must put a zero in the SCSI Link Bit, to indicate that the FORMAT UNIT packet is the last packet in this linked command list.

Note that although two SCSI linked commands are consecutively linked, when the same two commands are completed and put on a response list, there might be other unrelated commands between them in the response list.

Note further that the VH01 will continue to execute commands in a linked list until either the VH01 has executed the last command in the list, or a Command/Response Packet receives an error. Therefore, if a command in a linked list receives an error, the VH01 will execute none of the following commands on the linked list, and will not put those commands on a response list.

6.8 Systems with Multiple VMEbus Host Processors

Some VMEbus processor boards do not implement the Test And Set (TAS) instruction correctly over the VMEbus. Therefore, in a multiple host system, we suggest that you use a mechanism external to the VH01 to resolve contention for the VH01 by two or more hosts. For example, a semaphore in an area of VMEbus memory that is accessible by two or more hosts could be used.

6.9 VMEbus Host Device Driver

The device driver on the VMEbus host computer must perform a variety of different operations. The following subsections describe some of these operations, illustrating them with program fragments. The programming examples in C show all hexadecimal values with a leading "0x".

NOTE

The program fragments shown in the following two subsections should be regarded as examples only, not as complete programs.

6.9.1 Initiating a VH01 Command

Listing 6-1 shows, in the C programming language, how the device driver on the host processor should initiate a command:

```
while (! Test_And_Set(Entry_Queue[Q_Number].Ownership))
{
    /* Continue trying to set the bit until success.          */
}

/* Set up flags and other information here.  This is not      */
/* complete.                                                  */
/*                                                            */
/*                          .                                 */
/*                          .                                 */
/*                          .                                 */
/*                                                            */

Entry_Queue[Q_Number].CMD_ADDR      = Packet_Address;
Entry_Queue[Q_Number].CMD_ADDR_MODE = VMEbus_Address_Mode;
Entry_Queue[Q_Number].CMD_VALID     = SET;

/* Interrupt the VH01.                                        */

*Mailbox_Interrupt                  = 0x8000;
```

Listing 6-1. Initiating a VH01 Command

6.9.2 Servicing a VH01 Interrupt

Listing 6-2 shows a "C" example of the steps that the host processor should take when it services an interrupt from the VH01:

```
while (RSP->List_Header)
{
    while (RSP->Alter == 0x80)
    {
        /* Wait for the VH01 to finish with the update.        */
    }

    while (! Test_And_Set(RSP->Owner))
    {
        /* Continue setting until success.                     */
    }

    if (RSP->Alter == 0)
    {
        Ptr = RSP->List_Header;
        RSP->List_Header = RSP->List_Header->Next;


        RSP->Owner = 0;
        Process_Response(Ptr);
    }
    else
    {
       RSP->Owner = 0;
    }
}
RETURN_FROM_INTERRUPT;
```

Listing 6-2. Servicing a VH01 Interrupt

6.10 Using the VH01 in Target Mode

Normally, the VH01 acts as a SCSI initiator device. Initiators request that I/O operations be performed by another SCSI device known as a target device. However, the VH01 can also act as a target device and accept SCSI commands from another initiator.

In target mode, the VH01 acts as a "peripheral" device, and host memory is its "medium". In target mode, the VH01 responds as a SCSI "processor" device type. An initiator uses the SCSI SEND and RECEIVE commands to transfer data to and from the target VH01. The target VH01 must in turn be able to transfer data to and from its host memory. This is accomplished via the target buffer and target data VH01 queue commands. These commands are sent from the host to the VH01, acting as a target device to direct the flow of data in and out of the host's memory. The VH01 does not issue these commands as SCSI bus commands. Instead, they only cause buffers and pointers to be established in host memory, to facilitate target mode responses to SCSI SEND and RECEIVE commands. The host will send a target data command to the VH01, in case an initiator issues a SCSI RECEIVE command to the VH01. This target data command describes where the full data buffer is for the SCSI RECEIVE command sent from an initiator. On the other hand, the host will send a target buffer command to the target VH01, in case an initiator issues a SCSI SEND command to the VH01. This target buffer command describes where the data is to be stored for the SCSI SEND command.

For example, if the initiator issues a SEND command to a target VH01, the VH01 must accept the command and be ready to begin taking in the data as the initiator supplies it. In order to do this, the VH01 must have a place to put the data. Thus, the host can issue a target buffer command to the VH01 that will cause the VH01 to allocate a buffer and await a SEND command from an initiator.

A similar situation exists for the RECEIVE command. The host would set up a target data command to allocate and fill a buffer to use upon receipt of a RECEIVE command.

Refer to Figure 6-1 for an example using the target mode SEND command. In this example, the empty buffer in the VH01 Host System "A" system memory was set up via a target buffer command. This command allocates a buffer whose size must be determined by a previously agreed upon protocol. This buffer exists as a place to put data received from a SEND command when the VH01 is in target mode.

Since, in this example, both host adapters are VH01s, the typical method for bidirectional communication would be for Host System "B", acting as an initiator, to transmit a request for data, using the SEND command, as shown in Figure 6-1. Then, referring to Figure 6-2, Host System "A" would become the initiator and would select Host System "B" as the target device, and Host System "A" would use the SEND command to supply the requested data.

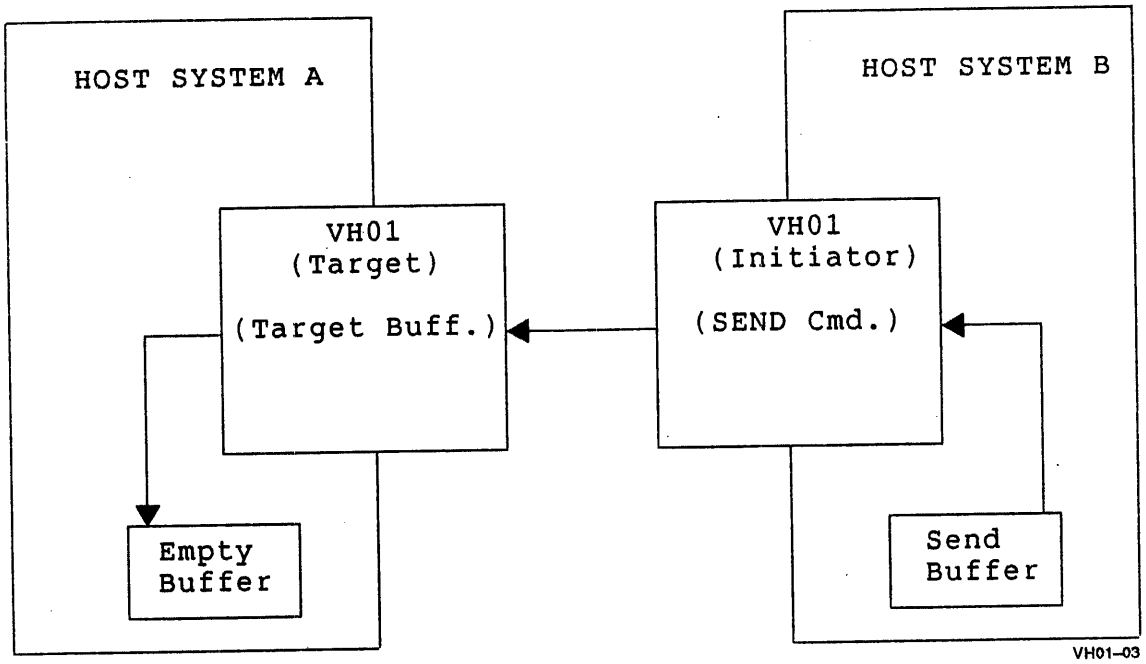


Figure 6-1. Target Mode SEND Command (Example 1)

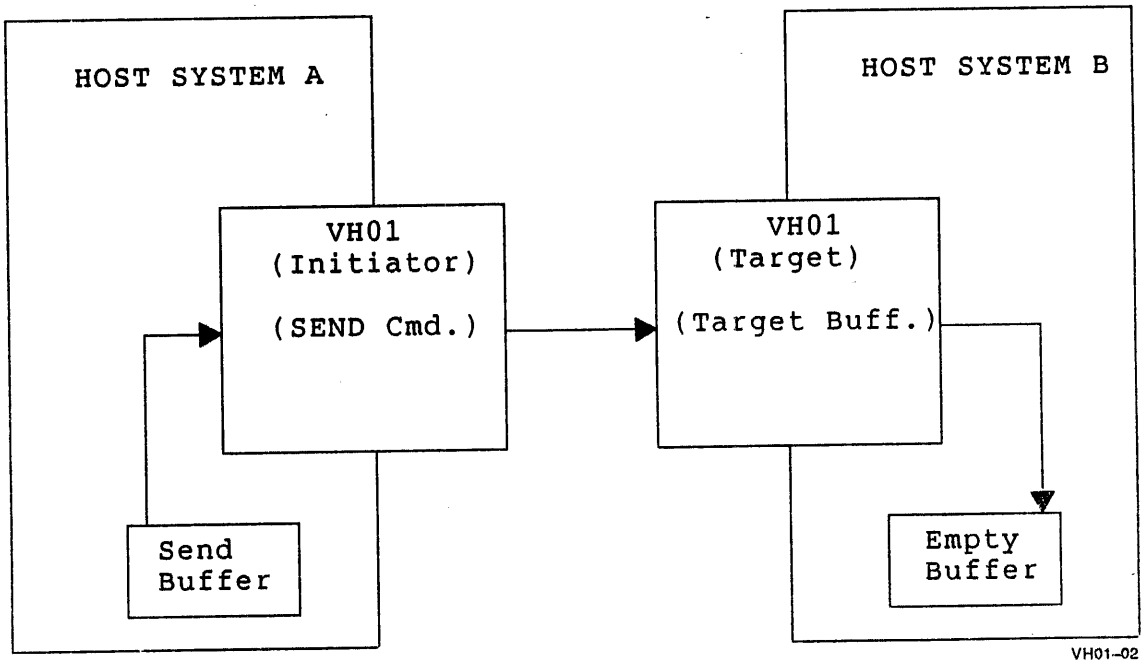


Figure 6-2. Target Mode SEND Command (Example 2)

The example in Figure 6-3 shows a situation where a host adapter that does not support target mode is used with a VH01. In this case, the VH01 must support both target mode SEND and RECEIVE commands, since the host adapter in Host System "B" does not support target mode and can not be issued a SEND command from Host System "A". This example shows the relationship of target data and target buffer commands in the VH01 target to the SCSI SEND and RECEIVE commands sent from a SCSI initiator. The target data and target buffer commands are simply a way to create buffers (either full data buffers for target data, or empty buffers for target buffer) to handle commands sent from an initiator. These commands do not translate into any SCSI commands, and can be thought of as "virtual" commands.

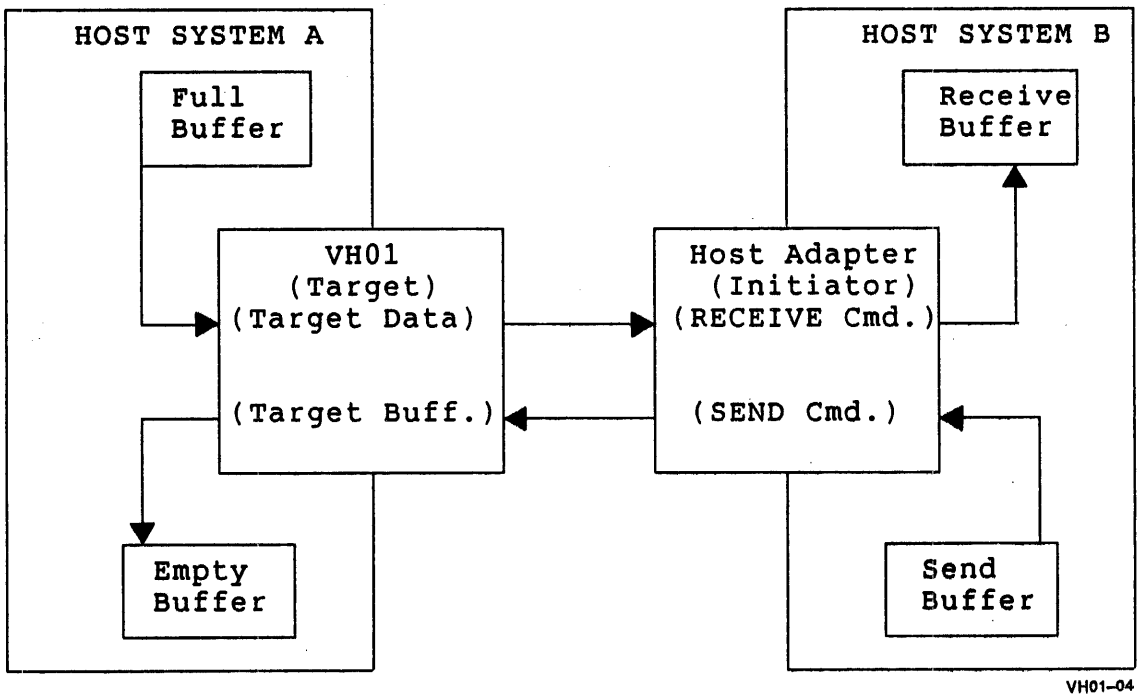


Figure 6-3. Target Mode SEND and RECEIVE Commands

BLANK

Appendix A
TROUBLESHOOTING

A.1 Overview

The installation of the VH01 Host Adapter, when used with the recommended devices, should run smoothly and problem-free. The diagnostic procedures described in this manual are intended to help you identify and resolve any problems you might encounter. However, because of the wide variety of VMEbus systems, floppy disk drives, and SCSI devices to which the VH01 could be connected, diagnostic procedures cannot be specific or all-inclusive. The following subsections explain how to isolate problems, and how to obtain technical assistance or service for problems that you cannot resolve.

A.2 Problem Identification

The following subsections explain how to interpret the error codes that can appear in the VH01's LEDs, how to isolate problems that might occur in the VH01 or its attached peripherals, and how to check the cables that connect the VH01 to its peripherals.

A.2.1 Reading VH01 Error Codes

When the VH01's SYSFAIL LED (LED4) is on, it indicates that an error condition has halted the VH01. There are two groups of errors that accompany the SYSFAIL LED: Nonflashing LED errors and flashing LED errors.

The VH01 displays nonflashing LED errors by setting a fixed, or steady-state, 3-bit binary code in LEDs 1 through 3. Nonflashing LED errors indicate VH01 selftest errors, as shown in Table A-1:

Table A-1. VH01 Selftest Error Codes

LED 1 2 3	Error Code
1 0 0	S1
0 1 0	S2
1 1 0	S3
0 0 1	S4
1 0 1	S5
0 1 1	S6

NOTE: In this table, a binary 1 means that the corresponding LED is ON; a binary 0 means the LED is OFF.

The VH01 displays flashing LED errors as a blinking pattern in LEDs 1 through 3. The pattern is displayed as a repeating sequence of four 3-bit values, called sequence 1 through sequence 4. Sequence 1 is displayed first, and consists of all three LEDs on, to mark the start of the sequence. To further differentiate sequence 1 from the other three sequences, sequence 1 is displayed for a shorter time period, giving it the appearance of a brief flash in the LEDs. After displaying sequence 1, the VH01 then displays sequence 2, sequence 3, and sequence 4, in order. After displaying sequence 4, the VH01 will repeat the entire sequence, starting once again with sequence 1.

Flashing LED errors indicate operational (runtime) errors, as shown in Table A-2:

Table A-2. VH01 Operational Error Codes

Seq. 1 LED 1 2 3	Seq. 2 LED 1 2 3	Seq. 3 LED 1 2 3	Seq. 4 LED 1 2 3	Error Code
1 1 1	0 1 0	0 0 0	0 1 0	01
1 1 1	0 1 0	0 0 0	0 0 1	02
1 1 1	0 1 0	0 0 0	0 1 1	03
1 1 1	0 1 0	1 0 0	0 0 0	04
1 1 1	0 1 0	1 0 0	0 1 0	05
1 1 1	0 1 0	1 0 0	0 0 1	06
1 1 1	0 1 0	1 0 0	0 1 1	07
1 1 1	0 1 0	0 1 0	0 0 0	08
1 1 1	0 1 0	0 1 0	0 1 0	09
1 1 1	0 1 0	0 1 0	0 0 1	10
1 1 1	0 1 0	0 1 0	0 1 1	11
1 1 1	0 1 0	1 1 0	0 0 0	12
1 1 1	0 1 0	1 1 0	0 1 0	13
1 1 1	0 1 0	1 1 0	0 0 1	14
1 1 1	0 1 0	1 1 0	0 1 1	15
1 1 1	0 1 0	0 0 1	0 0 0	16
1 1 1	0 1 0	0 0 1	0 1 0	17
1 1 1	0 1 0	0 0 1	0 0 1	18
1 1 1	0 1 0	0 0 1	0 1 1	19
1 1 1	0 1 0	1 0 1	0 0 0	20
1 1 1	0 1 0	1 0 1	0 1 0	21
1 1 1	0 1 0	1 0 1	0 0 1	22
1 1 1	0 1 0	1 0 1	0 1 1	23
1 1 1	0 1 0	0 1 1	0 0 0	24
1 1 1	0 1 0	0 1 1	0 1 0	25
1 1 1	0 1 0	0 1 1	0 0 1	26
1 1 1	0 1 0	0 1 1	0 1 1	27
1 1 1	0 1 0	1 1 1	0 0 0	28
1 1 1	0 1 0	1 1 1	0 1 0	29
1 1 1	0 1 0	1 1 1	0 0 1	30
1 1 1	0 1 1	1 1 1	0 1 0	F1
1 1 1	0 1 1	1 1 1	0 0 1	F2
1 1 1	0 1 1	1 1 1	0 1 1	F3

NOTE: In this table, a binary 1 means that the corresponding LED is ON; a binary 0 means the LED is OFF.

To interpret both flashing and nonflashing LED errors, locate the binary value for the error in the appropriate table. Use Table A-1 for nonflashing (selftest) errors and Table A-2 for flashing (operational) errors. Read across the table and find the corresponding error code (for example, error code F1). Next, look up this error code in Table A-3, VH01 Error Codes, following. In Table A-3, find both the error category and the error description corresponding to the error code. (For example, error code F1 lists an error category of "Fatal" and an error description of "Address error".) Finally, look up the error category in Table A-4, Error Category Explanation. In Table A-4, find the explanation and suggested corrective action for the error. (For example, error category "Fatal" gives the explanation as a "Fatal firmware problem".)

Table A-3. VH01 Error Codes

Error Code	Adapter State Byte Bits 7 0	Error Category	Error Description
S1	10010001	ST*	CPU selftest has failed
S2	10100001	ST*	ROM selftest has failed
S3	10110001	ST*	DRAM selftest has failed
S4	11000001	ST*	ESP selftest has failed
S5	11010001	ST*	DMA selftest has failed
S6	11100001	ST*	Floppy selftest has failed
01	10000010	EorT	Reselect without target ID bits set
02	10000100	EorT	Select without Phase or Bus Free
03	10000110	Berr	Response List Header bus error
04	10001000	Berr	Response List Header bus error
05	10001010	Berr	Response List Header bus error
06	10001100	Berr	Response List Header bus error
07	10001110	Berr	Response List Header bus error
08	10010000	Berr	Response List Header bus error
09	10010010	Berr	Response List Header bus error
10	10010100	Berr	Response List Header bus error
11	10010110	Berr	Response List Header bus error
12	10011000	Target	Target selected incorrectly
13	10011010	Target	ESP failure
14	10011100	Target	ESP failure
15	10011110	Target	ESP failure
16	10100000	Target	ESP failure
17	10100010	Target	ESP failure
18	10100100	Target	ESP failure
19	10100110	Target	ESP failure
20	10101000	EorT	Disconnecting--did not go Bus Free
21	10101010	Berr	Command packet (status area)
22	10101100	Driver	Invalid command from host in entry
23	10101110	Driver	Queued command had address of zero
24	10110000	Target	ESP problem
25	10110010	Target	ESP problem
26	10110100	Target	ESP problem
27	10110110	EorT	Disconnecting--did not receive disconnect interrupt
28	10111000	EorT	Disconnecting--did not go Bus Free
29	10111010	EorT	Disconnecting--did not receive disconnect interrupt
30	10111100	EorT	Bad SCSI bus phase
F1	11111010	Fatal	Address error
F2	11111100	Fatal	Trap error
F3	11111110	Ext	AC fail

*The LEDs will not flash for these errors.

Table A-4. Error Category Explanation

Error Category	Explanation and Corrective Action Needed
ST	Selftest failure. The VH01 hardware has malfunctioned. The VH01 might be defective. Refer to subsection A.2.2, "Problem Isolation," following.
Berr	Bus error. This usually indicates a device driver error. Carefully examine the specified address and address modifier. Also, ensure that the entire data structure being referenced is in memory, and that any reserved fields that the driver does not directly use are in valid memory.
Driver	The device driver sent invalid information to the VH01. Carefully examine the information from the driver, and correct it if necessary.
EorT	ESP or target device error. Replacing the VH01 might or might not help. An error in this category could indicate an unusual target SCSI implementation. It also could indicate that the VH01 is defective. Refer to subsection A.2.2, "Problem Isolation," following.
Target	The VH01 is executing in target mode and the ESP is not performing correctly. The VH01 might be defective. Refer to subsection A.2.2, "Problem Isolation," following.
Fatal	Fatal firmware problem. The hardware has corrupted the firmware. The VH01 might be defective. Refer to subsection A.2.2, "Problem Isolation," following.
Ext	External event error. The system has crashed due to an event beyond the control of the VH01 and external to it, such as a power failure. Restart the system.

A.2.2 Problem Isolation

The selftest, described in subsection 3.7, "Power-Up and System Verification," diagnoses problems within the VH01 itself. It does not diagnose problems with the VMEbus system, floppy disk drives, or SCSI devices to which the VH01 is connected.

If the VH01 does not pass the selftest when it is connected to the floppy disk drives and SCSI devices in your system, then disconnect the peripherals and try the selftest again. If the VH01 passes the selftest, then it is likely that the problem is elsewhere in the system. Refer to subsections 3.4.4.3, "SCSI Termination Power Jumper" and 3.6, "Cabling," and check that the termination is correct for the SCSI bus and the floppy drives. Also, refer to subsection 3.6 and check that the cables are installed correctly from the VH01 to the floppy drives and to the SCSI bus devices. Finally, refer to subsection A.2.3, "Checking the Peripheral Cables," following. If none of these checks uncovers a problem, refer to the directions for obtaining help from Emulex's technical support personnel in subsection A.3, "Technical Assistance".

If the VH01 does not pass the selftest when it is tested apart from the peripherals, then it could be defective and should be returned to Emulex for repair or replacement. See the directions for returning the unit in subsection A.4, "Service".

A.2.3 Checking the Peripheral Cables

Before returning the VH01 board for service, check the cables between the VH01 and its peripherals (SCSI bus and floppy drives). Verify that the connectors on the ends of the cables are fully seated in the mating connectors on the VH01 board and on the peripherals. For each cable connector, verify that it is oriented correctly with respect to its mating connector--that is, verify that pin 1 of the cable connector connects to pin 1 of the board connector. Check the condition of the cables, especially the connectors on the cables. Verify that the connectors appear to be properly crimped onto the cable wires. If another set of cables is available, you might want to try substituting them for the cables between the VH01 and its peripherals. If the symptoms then disappear, the original cables are probably bad and should be repaired or permanently replaced.

A.3 Technical Assistance

If the VH01 Host Adapter passed the selftest, but you believe it is not performing as expected, you can obtain assistance by contacting Emulex's technical support personnel at the address or telephone number listed in the subsection on Service, following.

A.4 Service

Your Emulex VH01 Host Adapter has been designed to give years of trouble-free service, and it was thoroughly tested before leaving the factory.

Should one of the diagnostic procedures described in this manual indicate that the VH01 Host Adapter is not working properly, the VH01 must be returned to Emulex for service. Emulex products are not designed to be repaired in the field.

Before returning the VH01 to Emulex, whether the product is under warranty or not, we prefer that you contact Emulex Technical Support.

In the continental United States, Alaska, and Hawaii, contact:

Emulex Technical Support
3545 Harbor Boulevard
Costa Mesa, CA 92626

(714) 662-5600

TWX (910) 595-2521

Outside the United States, contact the distributor from whom the VH01 Host Adapter was initially purchased.

To help you efficiently, Emulex or its representative requires certain information about our product and the environment in which it is installed. Before you call, be sure to have ready the parameters you have chosen for the VH01.

After you have contacted Emulex, package the VH01 (preferably using the original packing material) and send the VH01 **POSTAGE PAID** to the address given you by the Emulex representative. The sender must also insure the package.

Appendix B
CONNECTOR PIN ASSIGNMENTS

B.1 VMEbus Connectors and Interface Signals

Tables B-1 through B-3 show the connections between the VH01 and the VMEbus, and the connections of the seven user-defined external VMEbus signals, EXT 0 through EXT 6. Two P2 connectors are tabulated, Table B-2 for the single-ended SCSI version of the VH01, and Table B-3 for the differential SCSI version of the VH01.

For further information on the VMEbus interface signals, refer to the VMEbus standard, as listed in the Preface.

For further information on the SCSI interface signals, refer to the SCSI standard, as listed in the Preface.

Table B-1. VMEbus Interface Signals (Connector P1)

PIN NUMBER	ROW a SIGNAL MNEMONIC	ROW b SIGNAL MNEMONIC	ROW c SIGNAL MNEMONIC
1	D00	BBSY*	D08
2	D01	BCLR*	D09
3	D02	ACFAIL*	D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK[1]	BG3IN*	SYSFAIL*
11	GND	BG3OUT*	BERR*
12	DS1*	BR0*	SYSRESET*
13	DS0*	BR1*	LWORD*
14	WRITE*	BR2*	AM5
15	GND	BR3*	A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20	IACK*	GND	A18
21	IACKIN*	SERCLK[1]	A17
22	IACKOUT*	SERDAT*[1]	A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	IRQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31	-12[1]	+5STDBY[1]	+12V
32	+5V	+5V	+5V

[1]The VH01 does not use these signals.

Table B-2. Single-ended VH01 VMEbus Interface Signals (Connector P2)

PIN NUMBER	ROW a SIGNAL MNEMONIC	ROW b SIGNAL MNEMONIC	ROW c SIGNAL MNEMONIC
1	GND	+5V	SCSI D0
2	GND	GND	SCSI D1
3	GND	RESERVED	SCSI D2
4	GND	A24	SCSI D3
5	GND	A25	SCSI D4
6	GND	A26	SCSI D5
7	GND	A27	SCSI D6
8	GND	A28	SCSI D7
9	GND	A29	SCSI DP
10	--	A30	--
11	GND	A31	GND
12	--	GND	--
13	--	+5V	--
14	GND	D16	GND
15	--	D17	--
16	GND	D18	SCSI ATN
17	--	D19	--
18	GND	D20	SCSI BSY
19	GND	D21	SCSI ACK
20	GND	D22	SCSI RST
21	GND	D23	SCSI MSG
22	GND	GND	SCSI SEL
23	GND	D24	SCSI C/D
24	GND	D25	SCSI REQ
25	GND	D26	SCSI I/O
26	EXT 0	D27	EXT 4
27	EXT 1	D28	EXT 5
28	EXT 2	D29	EXT 6
29	EXT 3	D30	--
30	--	D31	--
31	--	GND	--
32	--	+5V	--

Table B-3. Differential VH01 VMEbus Interface Signals (Connector P2)

PIN NUMBER	ROW a SIGNAL MNEMONIC	ROW b SIGNAL MNEMONIC	ROW c SIGNAL MNEMONIC
1	GND	+5V	GND
2	+SCSI D0	GND	-SCSI D0
3	+SCSI D1	RESERVED	-SCSI D1
4	+SCSI D2	A24	-SCSI D2
5	+SCSI D3	A25	-SCSI D3
6	+SCSI D4	A26	-SCSI D4
7	+SCSI D5	A27	-SCSI D5
8	+SCSI D6	A28	-SCSI D6
9	+SCSI D7	A29	-SCSI D7
10	+SCSI DP	A30	-SCSI DP
11	--	A31	GND
12	GND	GND	GND
13	SCSI TERMPWR	+5V	SCSI TERMPWR
14	GND	D16	GND
15	+SCSI ATN	D17	-SCSI ATN
16	GND	D18	GND
17	+SCSI BSY	D19	-SCSI BSY
18	+SCSI ACK	D20	-SCSI ACK
19	+SCSI RST	D21	-SCSI RST
20	+SCSI MSG	D22	-SCSI MSG
21	+SCSI SEL	D23	-SCSI SEL
22	+SCSI C/D	GND	-SCSI C/D
23	+SCSI REQ	D24	-SCSI REQ
24	+SCSI I/O	D25	-SCSI I/O
25	GND	D26	GND
26	EXT 0	D27	EXT 4
27	EXT 1	D28	EXT 5
28	EXT 2	D29	EXT 6
29	EXT 3	D30	--
30	--	D31	--
31	--	GND	--
32	--	+5V	--

B.2 SCSI Connector and Interface Signals

Tables B-4 and B-5 show the connections between the VH01 and the SCSI single-ended or differential bus provided by connector J2. For the single-ended version of the VH01, J2 is the SCSI single-ended connector; for the differential version, J2 is the SCSI differential connector. For a pin-out of the SCSI single-ended signals on the P2 VMEbus connector, see Table B-2; for the SCSI differential signals on P2, see Table B-3.

For further information on the SCSI single-ended and differential interface signals, refer to the SCSI standard, as listed in the Preface.

Table B-4. SCSI Single-Ended Interface Signals (Connector J2)

Pin Number	Signal Name
1-49 (Odd numbers, except 25)	GROUND
25	Open
2	DB0 L
4	DB1 L
6	DB2 L
8	DB3 L
10	DB4 L
12	DB5 L
14	DB6 L
16	DB7 L
18	DBP L
20	GROUND
22	GROUND
24	GROUND
26	TERMPWR
28	GROUND
30	GROUND
32	ATN L
34	GROUND
36	BSY L
38	ACK L
40	RESET L
42	MSG L
44	SEL L
46	-C/D L
48	REQ L
50	-I/O L

Table B-5. SCSI Differential Interface Signals (Connector J2)

Signal Name	Pin Number		Signal Name
SHIELD GROUND	1	2	GROUND
+DB0	3	4	-DB0
+DB1	5	6	-DB1
+DB2	7	8	-DB2
+DB3	9	10	-DB3
+DB4	11	12	-DB4
+DB5	13	14	-DB5
+DB6	15	16	-DB6
+DB7	17	18	-DB7
+DBP	19	20	-DBP
DIFFSENS	21	22	GROUND
GROUND	23	24	GROUND
TERMPWR	25	26	TERMPWR
GROUND	27	28	GROUND
+ATN	29	30	-ATN
GROUND	31	32	GROUND
+BSY	33	34	-BSY
+ACK	35	36	-ACK
+RESET	37	38	-RESET
+MSG	39	40	-MSG
+SEL	41	42	-SEL
+C/D	43	44	-C/D
+REQ	45	46	-REQ
+I/O	47	48	-I/O
GROUND	49	50	GROUND

B.3 Floppy Connector and Interface Signals

Table B-6 shows the connections between the VH01 and the floppy disk drives.

Table B-6. Floppy Disk Interface Signals (Connector J1)

Pin Number	Source	Signal Name
1-33 (odd numbers, except 5)	--	Ground
5	--	Reserved for polarization (KEY)
2	Controller	Reduced Write Current L
4	----	Reserved
6	Controller	
8	Floppy Disk	Index L
10	Controller	Motor Enable 1 L
12	Controller	Drive Select 2 L
14	Controller	Drive Select 1 L
16	Controller	Motor Enable 2 L
18	Controller	Direction
20	Controller	Step
22	Controller	Write Data
24	Controller	Write Gate L
26	Floppy Disk	Track 00 L
28	Floppy Disk	Write Protect L
30	Floppy Disk	Read Data
32	Controller	Side 1 Select
34	Floppy Disk	Diskette Ready L

CAUTION

For proper operation, the VH01 floppy port requires that the floppy drives provide the DISKETTE READY signal, *not* the DISKETTE CHANGE signal. Many newer floppy drives provide a jumper for selecting which signal will be output. For operation with the VH01 floppy port, these drives *must* be jumpered to output the DISKETTE READY signal. For further information, refer to subsection 3.4.4.4, "Floppy Jumper".

BLANK

Appendix C
SCSI COMMAND SUMMARY

C.1 VH01 SCSI Commands

The VH01 normally issues SCSI commands to the SCSI bus separately, and in the order that they were received from the host. However, under certain circumstances, the VH01 will resequence some SCSI commands received from the host or combine them with other commands. Also, there is one SCSI command (REQUEST SENSE) that the VH01 internally synthesizes and issues to the SCSI bus under certain conditions. In addition, there is one SCSI command (MODE SELECT) that the VH01 stores and reissues to the SCSI bus under certain conditions. Table C-1 summarizes the SCSI commands that fall into the above categories.

For further information on these SCSI commands, or for information on SCSI commands not listed, refer to the SCSI standard, as listed in the Preface.

Table C-1. VH01 SCSI Commands

Command Name	Hex Operation Code	Notes
READ	08	1,2
WRITE	0A	1,2
READ EXTENDED	28	1,2
WRITE EXTENDED	2A	1,2
REQUEST SENSE	03	3
MODE SELECT	15	4

[1]The VH01 will combine these SCSI commands when it performs command combining.

[2]The VH01 will resequence these SCSI commands when it performs seek optimization.

[3]When a SCSI CHECK CONDITION occurs, the VH01 will generate a SCSI REQUEST SENSE command. If the target does not support REQUEST SENSE, then the host must disable Autorequest Sense.

[4]The VH01 Mode Select Initialization Command uses the SCSI MODE SELECT command.

C.1.1 VH01 Target Mode SCSI Commands

When the VH01 responds as a SCSI target device to another SCSI initiator, the VH01 supports the commands shown in Table C-2:

Table C-2. VH01 Target Mode SCSI Commands

Command Name	Hex Operation Code
TEST UNIT READY	00
REQUEST SENSE	03
RECEIVE	08
SEND	0A
INQUIRY	12
SEND DIAGNOSTIC	1D

C.2 SCSI Message Codes

Table C-3 summarizes for convenience the SCSI message codes.

For further information on SCSI message codes and their meanings, refer to the SCSI standard, as listed in the Preface.

Table C-3. SCSI Message Codes

Hex Message Code	Description
00	COMMAND COMPLETE
01	EXTENDED MESSAGE
02	SAVE DATA POINTER
03	RESTORE POINTERS
04	DISCONNECT
05	INITIATOR DETECTED ERROR
06	ABORT
07	MESSAGE REJECT
08	NO OPERATION
09	MESSAGE PARITY ERROR
0A	LINKED COMMAND COMPLETE
0B	LINKED COMMAND COMPLETE (WITH FLAG)
0C	BUS DEVICE RESET
0D-7F	Reserved Codes
80-FF	IDENTIFY

C.3 SCSI Status Byte Codes

Table C-4 summarizes for convenience the SCSI status byte codes.

For further information on SCSI status byte codes and their meanings, refer to the SCSI standard, as listed in the Preface.

Table C-4. SCSI Status Byte Codes

Hex Code	Status(es) Represented
00	GOOD
02	CHECK CONDITION
04	CONDITION MET/GOOD
08	BUSY
10	INTERMEDIATE/GOOD
14	INTERMEDIATE/CONDITION MET/GOOD
18	RESERVATION CONFLICT

D.1 Floppy Command Set

This appendix gives the commands for the VH01 floppy port. It also details the parameters for the SCSI MODE SELECT command for the floppy port.

D.2 VH01 Floppy Commands

The floppy interface on the VH01 executes a subset of the SCSI command set. Table D-1 lists the floppy commands.

Table D-1. VH01 Floppy Commands

SCSI Group	Command Name	Hex Operation Code	Comments
0	REZERO UNIT	01	Flag and Link command fields not supported.
0	FORMAT UNIT	04	FmtDat, CmpLst, Defect List Format, Flag, and Link command fields not supported. Media defect lists not supported.
0	MODE SELECT	15	SMP, Flag, and Link command fields not supported. Parameter list length must be 0 or 44; the VH01 treats any other value as an invalid request.
0	MODE SENSE	1A	PCF, Flag, and Link command fields not supported. SCSI Page Code must be 5; the VH01 treats any other value as an invalid request.
1	READ	28	RelAdr, Flag, and Link command fields not supported.
1	WRITE	2A	RelAdr, Flag, and Link command fields not supported.
1	VERIFY	2F	BytChk, RelAdr, Flag, and Link command fields not supported. The VH01 does not support data comparison. The VH01 checks the medium only for uncorrectable data errors.

The VH01 floppy port will treat any SCSI command other than the ones listed in Table D-1 as an invalid request.

All fields listed as unsupported in Table D-1 and all SCSI reserved and vendor-unique fields must be zero, otherwise the VH01 will reject the command as an invalid request.

The VH01 floppy interface does not support the SCSI REQUEST SENSE command directly. The VH01 automatically returns the SCSI Extended Sense data at the completion of each SCSI command, and supports all fields except FMark, EOM, ILI, FRU, BPV, and Bit Pointer.

D.3 SCSI Mode Select Parameters (Floppy Interface)

The VH01 floppy interface supports the following Mode Select Header Parameter Field. (All other parameter fields must be zero):

- Block Descriptor Length (must equal 8)

(The VH01 does not use the Medium Type Field, as the VH01 determines this automatically by interrogating the drive.)

The VH01 floppy interface supports the following Mode Select Block Descriptor Parameter Field. (All other parameter fields must be zero):

- Logical Block Length (3 bytes)

NOTE

If a logical block length of zero is specified, the VH01 will use the physical sector size as the default block length.

The VH01 floppy interface supports the Extended Mode Select Flexible Disk Drive Parameter Fields listed in Table D-2. All other Extended Mode Select Parameter Fields must be zero.

The parameter fields transfer rate, sectors per track, physical sector size, and number of cylinders must be specified. All other parameter fields listed in Table D-2 can be defaulted by setting the parameter value to zero.

For specific parameter values for IBM-compatible drives and media, see Table D-3 in the next subsection.

Table D-2. Extended Mode Select Parameter Fields

Parameter	Units	Byte Offset (decimal)	Byte Length	Notes
Page code	0x05	0	1	
Page length	0x1E	1	1	
Transfer rate	KBPS	2	2	1
Sectors per Track	sectors	5	1	
Physical sector size	bytes	6	2	2
Number of cylinders	cylinders	8	2	
Write precomp cylinder	cylinder	10	2	
Gap length for formatting	bytes	12	1	3
Gap length for read/write	bytes	13	1	3
Drive step rate	0.1 mS	14	2	4
Motor on delay	0.1 S	19	1	5
Motor off delay	0.1 S	20	1	6,7
Head load delay	1.0 mS	24	1	8,9
Head unload delay	1.0 mS	25	1	10

[1]The VH01 supports only 250, 300 and 500 KBPS data rates.

2The VH01 supports only 256, 512 and 1024 bytes per sector.

3A zero value indicates that the VH01 will use the default gap lengths, depending on drive type and sector size. Note that these fields replace the reduced write current starting cylinder.

4A zero value indicates that the VH01 will use the default step rate:
6 mS at 250 KBPS
5 mS at 300 KBPS
3 mS at 500 KBPS

5If zero is specified, the motor on delay defaults to 1.0 seconds.

6If zero is specified, the motor off delay defaults to 25.5 seconds.

7The VH01 interprets a motor off delay value of 255 as 25.5 seconds, rather than as "motor on signal shall not be de-asserted".

8Head load delay includes head settle time.

9A zero value indicates that the VH01 will use the default head load delay:
36 mS at 250 KBPS
30 mS at 300 KBPS
18 mS at 500 KBPS

A value of 255 indicates that the VH01 will use the maximum head load delay:
512 mS at 250 KBPS
427 mS at 300 KBPS
256 mS at 500 KBPS

[10]A zero value or a value of 255 indicates that the VH01 will use the default (maximum) head unload delay:
512 mS at 250 KBPS
427 mS at 300 KBPS
256 mS at 500 KBPS

The VH01 supports only the standard 125 nanosecond write precompensation value for floppy diskettes.

After a power-up sequence, the VMEbus host processor must issue a SCSI MODE SELECT command before formatting the first diskette. In addition, the host should issue the MODE SELECT command each time different formatting parameters are required. Otherwise, the VH01 will use the current Mode Select parameters for formatting. We recommend that the host issue the MODE SELECT command prior to formatting each diskette.

D.4 IBM-Compatible Mode Select Parameters (Floppy Interface)

Table D-3 lists the required Mode Select parameters for IBM-compatible floppy drive and media configurations:

Table D-3. IBM-Compatible Floppy Mode Select Parameters

Drive Type and Media Type	Transfer Rate (KBPS)	Bytes Per Sector	Sectors Per Track	No. of Cylinders
5-1/4 inch Dual Density, with 360 KB Media	250	512	9	40
5-1/4 inch Quad Density, with 360 KB Media	300	512	9	40
5-1/4 inch Quad Density, with 1.2 MB Media	500	512	15	80
3-1/2 inch, with 720 KB Media	250	512	9	80
3-1/2 inch, with 1.44 MB Media	500	512	18	80

D.5 SCSI Mode Sense Parameters (Floppy Interface)

The floppy MODE SENSE command returns all parameter fields that can be set with the MODE SELECT command. In addition, MODE SENSE returns the following:

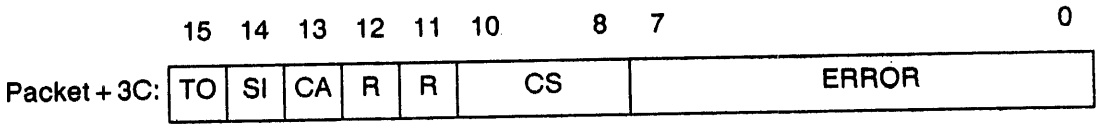
1. SCSI Medium Type Code (0x12, 0x1A, 0x1E)
2. Density code
3. Write protect status
4. Number of logical blocks
5. Number of heads (always 2)

BLANK

Appendix E
VH01 ERROR CODES

E.1 VH01 Internally-defined Error Word

This appendix summarizes the values and meanings of the VH01 Internally-defined Error Word.



The VH01 Internally-defined Error Word is a sixteen-bit location in the Command/Response Packet that contains error information for the host processor. Bits 0 through 7 of the error word form a numeric error code value, bits 8 through 10 give the VH01 processing state of the command in this Command/Response Packet, and bits 13 through 15 of the error word are individual error flags. The error flags can occur individually, in combination with each other, or in combination with the numeric error code value.

The following subsections describe the numeric error codes, the command state bits, and the error flag bits.

E.1.1 Error Codes (ERROR)

Error Code 0x0 - Normal Command Complete

The VH01 successfully completed the command request.

Error Code 0x1 - Auto Configuration Error

A condition occurred that required that the VH01 re-send the SCSI mode select/initialization command. When the VH01 re-sent the command, an error occurred.

Error Code 0x2 - Bus Reset

The SCSI Bus was reset and retries were disabled.

Error Code 0x3 - Check Condition

The SCSI target had an error of some sort, but auto-request sense has been disabled.

Error Code 0x4 - Unit Attention

The SCSI target sent a SCSI Check Condition status code and the VH01 did a SCSI Request Sense Command, but retries are disabled.

Error Code 0x5 - Device Busy

The VH01 made 20 attempts to select a SCSI device, but the device was busy.

Error Code 0x6 - Board Initialization Required

The host processor must perform a VH01 Host Adapter Initialization command before the VH01 will accept any commands.

Error Code 0x7 - Device Initialization Required

A SCSI Device Initialization must be sent for this SCSI device before it can accept a command.

Error Code 0x8 - Invalid Request

The request parameters are invalid. Verify that the parameters are in the proper format.

Error Code 0x9 - Parity Error

A SCSI target will typically retry parity errors one time. If a parity error remains during the retry, the VH01 will return this error code, and the VH01 will not attempt internally to retry the command.

Error Code 0xA - Hardware Error

Normally, this error will never occur. If this error does occur, it indicates a serious problem in the hardware of the VH01.

Error Code 0xB - Device Not Ready

The SCSI target did not acknowledge the VH01's selection. The VH01 attempts the selection two times.

Error Code 0xC - Message Rejected

The SCSI target did not allow the VH01 to send a standard mandatory SCSI message. The VH01 instantly aborted the command.

Error Code 0xD - Host Bus Error

While it was processing this command, the VH01 detected the VMEbus BERR* (Bus Error) signal true.

Command State 0x7 - Complete

The VH01 has finished processing this command.

E.1.2 Error Flags (TO, SI, CA)

Command Aborted (CA)

The VH01 aborted the command at the host's request. (The host set the Abort Bit or the Reset Device Bit in the Command State Byte of the Command Queue.)

ERROR, Sense Information Valid (SI)

The VH01 performed a SCSI Request Sense command for the host processor, and stored the sense data in the SCSI Request Sense Packet, at the end of the Command/Response Packet.

Timeout (TO)

In addition to any other error information given, a timeout occurred.

BLANK

address modifier [VMEbus term] - An address modifier, or address modifier code, is a code that allows a bus MASTER to pass additional information to a bus SLAVE during Data Transfer Bus (DTB) cycles. The MASTER sends the address modifier code over the 6 VMEbus address modifier lines, AM0 through AM5. The address modifier codes are used to broadcast address size, cycle type, and privilege information.

address pipelining [VMEbus term] - The ability of a bus MASTER to broadcast the address for the next cycle while the data transfer for the previous cycle is still in progress.

ARBITER [VMEbus term] - A VMEbus functional module that accepts bus requests from REQUESTER modules and grants control of the Data Transfer Bus to one REQUESTER at a time.

ARBITER type [VMEbus term] - See **VMEbus ARBITER type.**

BG (bus grant) [VMEbus term] - On the VMEbus, there are eight totem-pole driven bus grant signals. Four of these, BG0IN* through BG3IN*, are the "bus grant in" signals, and the remaining four signals, BG0OUT* through BG3OUT*, are the "bus grant out" signals. The "bus grant in" and the "bus grant out" signals form four bus grant daisy chains. The first "bus grant in" signals are generated by the ARBITER; the "bus grant out" signals are generated by REQUESTERS. The "bus grant in" signal indicates to the board receiving it that it has been granted use of the Data Transfer Bus (DTB), while the "bus grant out" signal indicates to the next board in the daisy chain that it may use the DTB. (See also **BR.**)

block scatter/gather - See **scatter/gather.**

BLT (Block Mode Transfers) [VMEbus term] - A Data Transfer Bus cycle used to transfer a block of 1 to 256 bytes between a SLAVE and a MASTER. This transfer is done using a string of 1, 2, or 4 byte data transfers. Once the block transfer is started, the MASTER does not release the Data Transfer Bus until all of the bytes have been transferred. BLT differs from a string of read cycles or write cycles in that the MASTER broadcasts only one address and address modifier (at the beginning of the cycle). Then the SLAVE increments this address on each transfer so that the data for the next cycle is transferred to or from the next higher location.

BR (bus request) [VMEbus term] - On the VMEbus, there are four open-collector driven signals, BR0* through BR3*, that are generated by REQUESTERS. A low level on one of these lines indicates that some MASTER needs to use the Data Transfer Bus (DTB). (See also **BG**.)

burst [VMEbus term] - A series of transfers performed after an initial bus arbitration sequence grants control of the bus to a bus MASTER. No intervening arbitrations occur between transfers.

bus MASTER [VMEbus term] - See **MASTER**.

bus SLAVE [VMEbus term] - See **SLAVE**.

byte - A byte is 8 bits.

CCS (Common Command Set), SCSI CCS [SCSI term] - An early version of the upcoming ANSI SCSI-2 document, except intended for random-access devices only.

CDB (Command Descriptor Block), SCSI CDB [SCSI term] - The data structure used to communicate requests from an initiator to a target.

command entry queue - A VH01 queue, located in the VH01 Message Area, into which the VMEbus host processor places Command/Response Packets. The Command/Response Packets contain VH01 commands and command-related information. The VH01 has eight command entry queues. (See also **internal queue**.)

Command/Response Packet - A data structure that is used for communications between the VMEbus host processor and the VH01. The host creates the Command/Response Packet in the host memory. The Command/Response Packet contains SCSI or floppy commands and responses.

command queuing - The method whereby one or more commands can be issued to a peripheral controller before a previous command has completed.

connect [SCSI term] - The function that occurs when an initiator selects a target to start an operation. (See also **initiator**, **target**, **disconnect**, and **reconnect**.)

controller, peripheral controller - A device (usually composed of one or more printed circuit boards) that connects to a computer bus and controls one or more peripherals. (See also **driver** and **host adapter**.)

device driver - See **driver.**

disconnect [SCSI term] - The function that occurs when a target releases control of the SCSI bus, allowing it to go to the BUS FREE phase. (See also **initiator**, **target**, **connect**, and **reconnect.**)

DMA (Direct Memory Access) - A method of transferring data between memory and a peripheral controller or host adapter without involving the host processor.

DRAM window - See **VH01 VMEbus DRAM window.**

driver, device driver - A part of an operating system that directly interfaces with, and controls, a peripheral controller or host adapter and its attached peripherals. (See also **controller** and **host adapter.**)

early release [VMEbus term] - The condition where a VMEbus MASTER releases the BSY* signal before its last data transfer is completed, to allow overlapped bus arbitration to occur while the transfer takes place. (See also **overlapped arbitration.**)

ESP (Emulex SCSI Processor) - A VLSI device, designed by Emulex, that implements the detailed protocol of the SCSI bus. The ESP performs such functions as SCSI bus arbitration, selection of a target, or reselection of an initiator. It handles message, command, status, and data transfers between the SCSI bus and the VH01's internal memory. On the VH01, the ESP provides the interface between the SCSI bus and the other logic on the VH01 Host Adapter board. (See also **VLSI.**)

FIFO (First In, First Out) - A method of organizing a memory or storage device such that the first data item stored will be the first one removed. A FIFO can be implemented in hardware, software, or a combination of hardware and software. The FIFO on the VH01 is implemented in hardware.

host, host processor, VMEbus host processor [VMEbus term] - Within the context of the VH01, the term host processor refers to any VMEbus processor or CPU (central processing unit) that can be a VMEbus MASTER and can issue commands to the VH01. A VMEbus processor generally consists of one or more printed circuit boards that plug into the VMEbus.

host adapter, host computer adapter - A device (usually composed of one or more printed circut boards) that connects to a computer bus and allows the use on that computer of another, different, bus. Thus, the host adapter "adapts" this different bus to the bus of the host computer. For example, the VH01 Host Adapter adapts the SCSI bus to the VMEbus. (See also **controller** and **driver**.)

IC (Integrated Circuit) - A solid-state electronic device containing the equivalent of a number of discrete transistors. (See also **VLSI**.)

initiator, initiator device [SCSI term] - A SCSI device (usually a host system) that requests an operation to be performed by another SCSI device. (See also **target**.)

internal queue - A queue, located within the VH01, that is only used internally by the VH01, and is not directly accessible to either the SCSI bus or the VMEbus. (See also **command entry queue**.)

INTERRUPTER [VMEbus term] - A VMEbus functional module that generates an interrupt request on the Priority Interrupt Bus, and then provides STATUS/ID information when the INTERRUPT HANDLER requests it.

INTERRUPT HANDLER [VMEbus term] - A VMEbus functional module that detects interrupt requests generated by INTERRUPTERS and responds to those requests by asking for STATUS/ID information.

LED (Light-Emitting Diode) - A solid-state electronic device that emits light, commonly used as an indicator in electronic equipment.

logical unit [SCSI term] - A physical or virtual device addressable through a SCSI target.

longword - Within the context of the VH01, the 68000 microprocessor, and the VMEbus, a longword is 32 bits.

LUN (logical unit number), SCSI LUN [SCSI term] - An encoded three-bit identifier for a SCSI logical unit. (See also **logical unit**.)

mailbox - See **VH01 mailbox.**

mailbox interrupt, VMEbus mailbox interrupt - A means by which a host processor can interrupt a peripheral controller to indicate that a command and/or message awaits it. (See also **VH01 mailbox.**)

MASTER, bus MASTER [VMEbus term] - A VMEbus functional module that initiates Data Transfer Bus cycles in order to transfer data between itself and a SLAVE module. (See also **SLAVE.**)

Message Area - See **VH01 Message Area.**

multi-threaded [SCSI term] - Indicates the ability to simultaneously process several tasks or commands to different peripherals, rather than processing them sequentially.

overlapped arbitration [VMEbus term] - Occurs in conjunction with early release. Indicates that bus arbitration occurs simultaneously with the last data transfer of a VMEbus MASTER. (See also **early release.**)

packet - See **Command/Response Packet.**

page - Within the context of the VH01, a page is a block of 512 consecutive bytes of memory. A page always begins and ends on physical 512-byte boundaries.

peripheral, peripheral device [SCSI term] - A piece of computer input/output equipment, such as a printer, magnetic tape drive, magnetic disk drive, or optical disk drive, that can be attached to a SCSI device. (See also **SCSI device.**)

peripheral controller - See **controller.**

PRI ARBITER (prioritized ARBITER) [VMEbus term] - See **VMEbus ARBITER type.**

priority ARBITER [VMEbus term] - See **VMEbus ARBITER type.**

reconnect [SCSI term] - The function that occurs when a target selects an initiator to continue an operation after a disconnect. (See also **initiator, target, connect,** and **disconnect.**)

REQUESTER [VMEbus term] - A VMEbus functional module that resides on the same board as an INTERRUPT HANDLER or a MASTER and requests use of the Data Transfer Bus whenever its INTERRUPT HANDLER or MASTER needs it.

response list - A linked list data structure in VMEbus host memory consisting of a Response List Header and one or more completed Command/Response Packets. In this linked list, the Response List Head Pointer in the header points to the start of the first completed Command/Response Packet in the list. The Next Completed Packet Pointer field in the first completed packet points to the second completed packet, the Next Completed Packet Pointer in the second packet points to the third packet, and so on. The Next Completed Packet Pointer in the last completed packet contains zero, to mark the end of the linked list.

ROR (Release-On-Request) [VMEbus term] - Pertains to VMEbus MASTERS that perform burst transfers. Such MASTERS will continue to transfer data until a bus request is detected from another VMEbus MASTER. (See also **burst**, **RWD**, and **transfer limit**.)

round-robin-select ARBITER [VMEbus term] - See **VMEbus ARBITER type**.

RRS ARBITER (round-robin-select ARBITER) [VMEbus term] - See **VMEbus ARBITER type**.

RWD (Release-When-Done) [VMEbus term] - Pertains to VMEbus MASTERS that perform burst transfers. Such MASTERS will continue to transfer data until all the data has been transferred, or the transfer limit is reached, whichever occurs first. During the burst transfer, the MASTER will ignore bus requests from other VMEbus MASTERS. (See also **burst**, **ROR**, and **transfer limit**.)

scatter/gather, block scatter/gather - A method of transferring noncontiguous data blocks between system memory and a peripheral controller. Data blocks are **scattered** in memory by the operating system, due to memory-management schemes, to be **gathered** by the peripheral controller; or, data blocks are **scattered** by the peripheral controller, to be **gathered** by the operating system's memory management unit.

SCSI CCS [SCSI term] - See **CCS**.

SCSI CDB [SCSI term] - See **CDB**.

SCSI device [SCSI term] - A host computer adapter or a peripheral controller or an intelligent peripheral that can be attached to the SCSI bus. (See also **host adapter**, **controller**, **initiator**, and **target**.)

SCSI LUN [SCSI term] - See **LUN**.

SGL ARBITER (single-level ARBITER) [VMEbus term] - See **VMEbus ARBITER type**.

shared memory, shared memory space - See **VH01 VMEbus DRAM window**.

short space [VMEbus term] - The portion of the VMEbus address space that is addressable using address modifier codes that specify only 16-bit addressing. (See also **address modifier**.)

single ARBITER, single-level ARBITER [VMEbus term] - See **VMEbus ARBITER type**.

SLAVE, bus SLAVE [VMEbus term] - A VMEbus functional module that detects Data Transfer Bus cycles initiated by a MASTER and, when those cycles specify its participation, transfers data between itself and the MASTER. (See also **MASTER**.)

target, target device [SCSI term] - A SCSI device that performs an operation requested by an initiator. (See also **initiator**.)

target mode - A mode in which the VH01 acts as a SCSI target device, and accepts SCSI commands sent to it by another SCSI device acting as a SCSI initiator.

transfer limit - An integer indicating the maximum number of burst transfers a VMEbus MASTER can perform before releasing the bus. (The term **transfer limit** is a general term, which is not specified or used in the VMEbus specification; however, in the VH01, transfer limit applies only to the VMEbus.)

unaligned transfer, unaligned bus transfer - A transfer that starts and/or ends on an odd byte or odd word boundary.

VH01 mailbox - A word in the DRAM window that is used for communications from a VMEbus host processor to the VH01. The host must write to this word to interrupt the VH01 microprocessor, in order to signal it that a message awaits in the message area of the DRAM window. (See also **mailbox interrupt**.)

VH01 Message Area - A portion of the VH01 VMEbus DRAM window used for passing messages between the VH01 and the VMEbus host processor.

VH01 VMEbus DRAM window - The common 512-byte region of the VH01 DRAM that is accessible both by the VH01 microprocessor and by a VMEbus host processor.

VLSI (Very Large Scale Integration) - An Integrated Circuit (IC) containing a very large number of gates, usually more than 1000. (See also **IC**.)

VMEbus ARBITER type -- PRI, RRS, SGL [VMEbus term] - The VMEbus contains one bus ARBITER which is configured in one of three ways: Prioritized (PRI), Round-Robin-Select (RRS), or Single level (SGL).

Prioritized arbitration assigns the bus according to a fixed priority scheme where each of four bus request lines has a priority from highest (BR3*) to lowest (BR0*).

Round-robin arbitration assigns the bus on a rotating priority basis. When the bus is granted to the REQUESTER on bus request line BR(n)*, then the highest priority for the next arbitration is assigned to bus request line BR(n-1)*.

Single level arbitration only accepts requests on BR3*, and relies on BR3*'s bus grant daisy chain to arbitrate the requests.

VMEbus DRAM window - See **VH01 VMEbus DRAM window**.

VMEbus host processor [VMEbus term] - See **host**.

VMEbus mailbox interrupt - See **mailbox interrupt**.

VMEbus window - See **VH01 VMEbus DRAM window**.

window, DRAM - See **VH01 VMEbus DRAM window**.

word - Within the context of the VH01, the 68000 microprocessor, and the VMEbus, a word is 16 bits.

Rec 12-18-89

ENGINEERING CHANGE ORDER

EMULEX

ECO NO. 05742 ISSUE DATE 11/16/89

ECR NO. None SHEET 1 OF 1

REQUESTER: B. A. Brown | DATE: 11/16/89 | APPROVAL | DATE

MODEL/EQUIPMENT AFFECTED: VH01/S, VH01/SF, VH01/D

EFFECTIVITY
- [X] ALL UNITS IN HOUSE
- [ ] ALL UNITS IN PROCESS
- [ ] NEXT BUY/BUILD
- [ ] RECORD

EFFECTIVE DATE: __/__/__

AFFECTED
- [X] MANUALS
- [ ] FIRMWARE
- [ ] SOFTWARE
- [ ] DIAGNOSTICS
- [ ] PRODUCTION TOOLING
- [ ] CONFIGURATOR

MATERIAL DISPOSITION
1. USE AS IS
2. REWORK
3. PHASE-IN
4. NOT APPLICABLE
5. SCRAP

TYPE
- [ ] CORRECTIVE
- [ ] MAINTAINABILITY
- [ ] RECORD CHANGE
- [ ] DEVIATION

PRIORITY
- [ ] 1 CRITICAL
- [ ] 2 URGENT
- [X] 3 ROUTINE

REASON FOR CHANGE
Initial Release

WHERE USED: LIST BELOW ALL DOCUMENTS, PARTS AND ASSEMBLIES THAT ARE AFFECTED BY THIS CHANGE ORDER.

DOC/PART/ASSY NUMBER	CUR REV	NEW REV	TITLE/DESCRIPTION	MATERIAL DISPOSITION CODE: NEXT ORDER	ON ORDER	RECEIVING/ INSPECTION	STOCKROOM	WORK IN PROGRESS	FINISHED GOODS		
VH0151002-00	-	A	VH01 Programmer's Ref. Man.	3	4	4	4	4	4		

Instructions:

Print and distribute VH0151002-00 Rev. A

Reproduced in house.

VH0151002-00 Rev. A supersedes VH0151001-00 Rev. B.

The reference cards that were a part of VH0151001-00 Rev. B will not be included with VH0151002-00 Rev. A but will be released at a later date.

VH0151002-00 Rev. A includes the differential version of the VH01 (VH01/D).

COMMENTS
Please note: VH0151001-00 Rev. B is now obsolete.

FIELD CHANGE NOTICE
- [ ] YES
- [X] NO

ENGR SERVICES	ENGINEERING	MANUFACTURING	PURCHASING	QUALITY	CUST. SUPPORT	MARKETING
BC						

EF271A (12/88)

EMULEX

Reader's Comments

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publication.

Manual Part Number _____ **Rev.** _____

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? _____

_____

_____

_____

What features are most useful? _____

_____

_____

_____

What faults or errors have you found in the manual? _____

_____

_____

_____

Does this manual satisfy the need you think it was intended to satisfy? _____

Does it satisfy *your* needs? _____ Why? _____

_____

_____

_____

☐ Please send me the current copy of the *Controller Handbook*, which contains the information on the remainder of EMULEX's controller products.

Name _____ Street _____

Title _____ City _____

Company _____ State/Country _____

Department _____ Zip _____

Additional copies of this document are available from:

Emulex Corporation
3545 Harbor Boulevard
P.O. Box 6725
Costa Mesa, CA 92626
Attention: Customer Services

Fold Here and Staple

Do Not Tear—Fold Here

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 202 COSTA MESA, CA

POSTAGE WILL BE PAID BY ADDRESSEE

EMULEX CORPORATION
ATTN: TECHNICAL PUBLICATIONS
3545 HARBOR BOULEVARD
P.O. BOX 6725
COSTA MESA, CA 92628-9928