CONVEX VMEbus Async/Printer Controller Service Guide

Document No. 081-001430-000

First Edition
October 1989

CONVEX Computer Corporation
Richardson, Texas USA

CONVEX VMEbus Async/Printer Controller
Service Guide
Order No. DHW-057
First Edition

Printed in the United States of America

Revision Sheet

CONVEX VMEbus Async/Printer Controller
Service Guide

Edition	Document No.	Date	Description
First	081-001430-000	October 1989	First release, Version 1.0.

FCC NOTICE

Warning: This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in strict accordance with the instructions manual, may 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 FCC Rules, which are designed to provide reasonable protection against such interference when operated 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 may be required to correct the interference.

Do not connect external equipment to the utility outlets in CONVEX equipment cabinets. Unauthorized connection voids all agencies' emissions certifications.

Warning: This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in strict accordance with the instructions manual, may cause interference to radio communications.

Table of Contents

Appendixes

List of Tables

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

Preface

Purpose and Intended Audience

The *CONVEX VMEbus Async/Printer Controller Service Guide* provides a general overview of the VMEbus Async/Printer controller and related hardware and how to:

- Install the controller and related hardware
- Integrate the controller into the CONVEX Operating System (ConvexOS)
- Test the controller and related hardware
- Remove and replace the controller and related hardware

Primary Audience

This document is intended for:

- CONVEX Customer Support Engineers and CONVEX manufacturing personnel
- Customers who need to install a VMEbus Async/Printer controller and related hardware

Limitations

- Fault isolation to the single component level is not provided

Distribution

The *CONVEX VMEbus Async/Printer Controller Service Guide* is part of the *CONVEX VMEbus Service Documentation* kit.

This document is primarily for CONVEX customer support engineers use. It may be supplied to those customers who wish to install or maintain their own equipment.

Organization

The document consists of the following sections:

- **Chapter 1, Description and Specifications**—Describes the VMEbus Async/Printer controller and related hardware at the block diagram level. Defines and lists the electromechanical and environmental specifications.
- **Chapter 2, Unpacking and Installation**—Provides guidelines on how to unpack and install the VMEbus Async/Printer controller and related hardware.

- **Chapter 3, Integration and Test**—Explains how to integrate the VMEbus Async/Printer controller into the ConvexOS operating system. Explains how to test the VMEbus Async/Printer controller and related hardware.
- **Chapter 4, Maintenance Procedures and IPB**—Provides removal and replacement instructions for the controller and related hardware.
- **Appendix A, Async/Printer Configurator Document**—Contains a copy of the VMEbus Async/Printer Controller Configurator document.
- **Appendix B, Problem Reporting**—Provides an example of the CONVEX ***contact*** utility for reporting minor software and hardware problems.

Notational Conventions

The following are examples of warnings, cautions, and notes and their typical content as used in CONVEX documents:

WARNING

Warnings highlight procedures or information necessary to avoid injury to personnel. A warning immediately precedes the critical information and includes a description of the hazard.

CAUTION

Cautions highlight procedures or information necessary to avoid damage to equipment, loss of data, or invalid test results. A caution immediately precedes the critical information and includes a description of the possible damage.

NOTE

Notes highlight useful information that is supplemental in nature. A note may immediately precede or follow the information that is being highlighted.

Associated Documents

The following is a partial list of other manuals or books that may provide more detailed information on the topics presented in this manual:

- *CONVEX VIOP/VBCU Service Guide*, Order No. DHW-051

- *CONVEX PBUS I/O System Diagnostics Manual,* Order No. DHW-008
- *VME Async/Printer Controller Configurator,* CONVEX Part No. 410-001193-600
- *CONVEX System Manager's Guide,* Product No. DSW-004
- *CONVEX VMEbus Reference Manual,* Order No. DHW-061
- *CONVEX VMEbus Service Kit,* Order No. DHW-050
- *The VMEbus SPECIFICATION C.1,* Motorola Inc.
- *VMEarray Programmer's Manual,* CONVEX Part No. 081-001830-300
- *VMEarray Hardware Designer's Manual,* CONVEX Part No. 081-001930-300
- *VMEarray Reference Manual,* CONVEX Part No. 081-002030-300
- *VMEarray Test Vector Description,* CONVEX Part No. 081-002130-300
- *VMEarray Schematics,* CONVEX Part No. 081-002230-300
- *CONVEX Processor Operation Guide,* Order No. DHW-015
- *CONVEX Diagnostic Utility Manual,* Order No. DHW-082
- *RS-232-C User Reference Manual,* CONVEX Part No. 900-000334-001
- *Electrostatic Discharge Failures of Semiconductor Devices.* Unger, B.A. 1981. Bell Laboratories

Ordering Documentation

To order the most current version of this or any other CONVEX document, use the CONVEX order number. If the order number is not known, order by exact title. In some situations the most current version may not be desired. To receive a specific version of a manual, order the manual by its document, or part, number, which can be obtained by contacting the local CONVEX office or by calling the Technical Assistance Center.

The order number for this document is DHW-057
The part number for this document is 081-001430-000

CONVEX documents can be ordered by mail by sending a request to:

CONVEX Computer Corporation
Customer Service
PO Box 833851
Richardson Texas 75083-3851 USA

Technical Assistance

Hardware and software support can be obtained through the CONVEX Technical Assistance Center (TAC):

- From all locations in the continental United States, call 1(800)952-0379
- From locations in Alaska, Hawaii, and Canada, call 1(214)497-4379
- From all other locations, contact the nearest CONVEX office.

Electronic Mail

The Hardware Documentation Group has an email address for documentation comments. Use this service to give us a quick response mechanism if you have special documentation questions that you would like addressed immediately. If you have a technical question, you should still contact the Technical Assistance Center, as described previously. To use email response service, just send mail addressed to:

```
cnvxhwdoc@convex.COM
```

We will read your comments and give you a personal reply.

What to Include in an Email Message

When you use the electronic mail service, please provide the following information:

- The reader's name and company name
- A return email address in INTERNET notation or UUCP (bang) notation
- The manual that is being critiqued
- The chapter and page number in question
- The comment

Reader's Forum

If you wish to mail your comments to us, please use the form at the end of this manual and list the document page number with your questions and comments. Thank you.

Acknowledgments

I would like to thank the following people for their contributions to this manual:

- Technical contributors: Rob Carruthers, Mike Barron, and Ken King
- Document review team: Rob Carruthers, Mike Barron, Ken King, Chip Stroup, and Al Haddix
- Hardware documentation staff: Larry Bonura, Barbara Morris, and Josie Davis

Without the efforts of the aforementioned, this document would not have been possible.

Randall Stiles
CONVEX Hardware Documentation

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1
Description and Specifications

1.1 Overview

The CONVEX VMEbus Async/Printer controller is designed for maximum performance for communicating with single character devices such as a terminal or printer. The controller provides direct connection between an onboard microprocessor and various serial communication devices. This is in contrast to a block oriented device such as a disk controller which typically has only a DMA connection to a common data RAM.

The VMEbus Async/Printer controller assembly is divided into three major sections:

1. The VMEbus Async/Printer controller contains the processor and VMEbus interface, as well as instruction and data RAMs
2. The VMEbus Async/Printer panel assembly, mounted at the rear of a CONVEX expansion cabinet, contains various serial communication devices
3. The daughter cards, mounted to the communications panel, contain RS-232 interface drivers and receivers

This chapter describes the architecture and characteristics of the CONVEX VMEbus Async/Printer controller. The controller has the following features:

- An M68020 microprocessor with 256 Kbytes of 0-wait state Instruction RAM (IRAM)
- Programmable M68020 microprocessor interrupt levels and vectors
- 256 Kbytes of Dual Ported data RAM (DPR)
- 32 Kbytes of bootstrap Electronic Programmable Read Only Memory (EPROM)
- Two On board timers for kernel, network timing, and printer control
- Sixteen full-modem control async ports with ring-in capability
- Centronix and Dataproducts compatible printer interface capable of operating up to 50 feet (15 m) from the controller with a transfer rate of 300 Kbaud per second
- Very Large Scale Integration (VLSI) Application Specific Integration Circuit (ASIC) VMEbus interface that provides Direct Memory Access (DMA) capability for either instructions or data Random Access Memory (RAM) for the VMEbus

1.2 Logical Organization

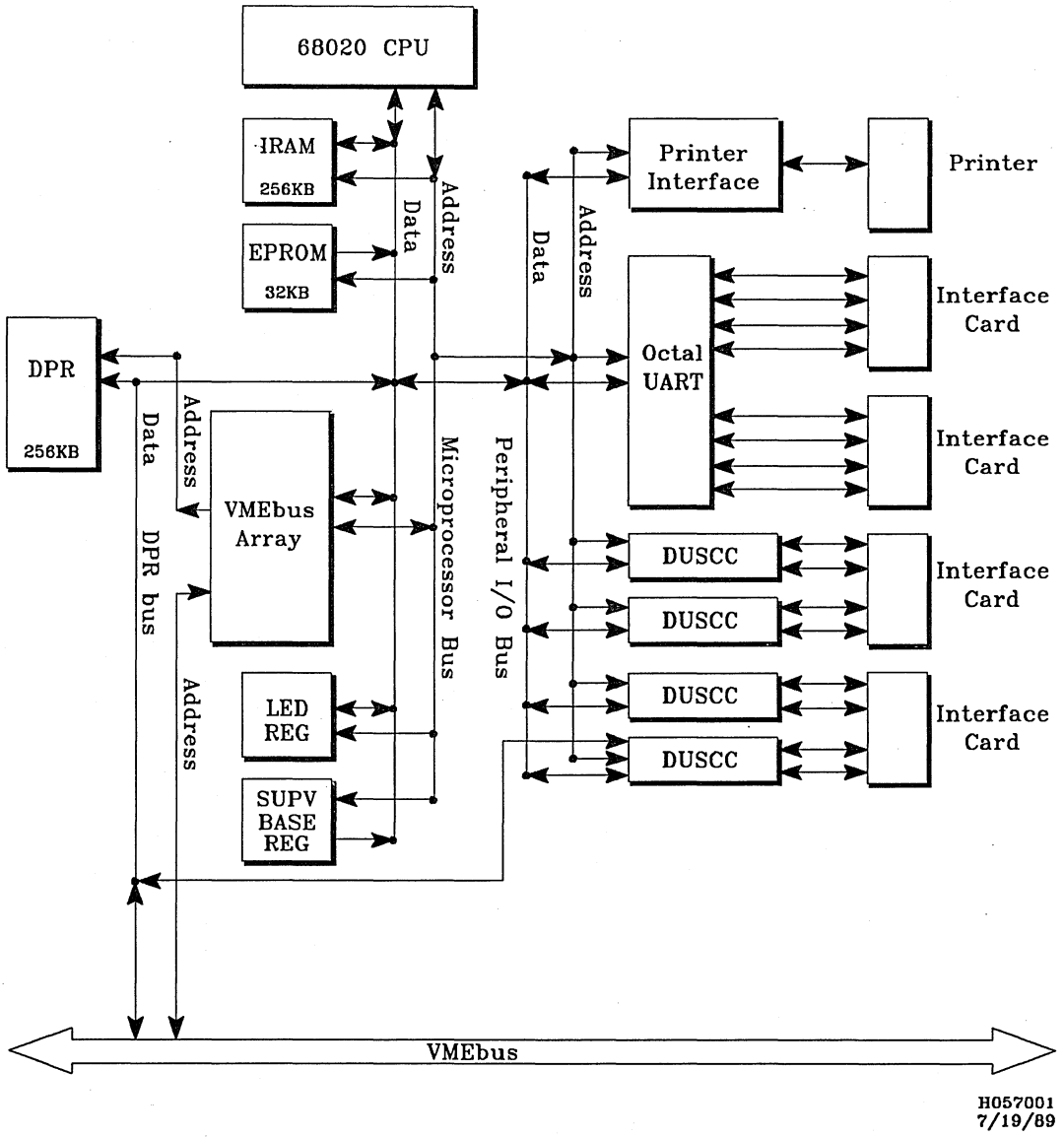
The VMEbus Async/Printer controller is divided into four major sections:

1. Local microprocessor and Instruction RAM (IRAM)
2. Local resource and communication device interfaces

3. Dual port RAM (DPR)
4. VMEbus interface

A block diagram of this controller configuration is shown in Figure 1–1, "VMEbus Async/Printer Controller Block Diagram". The physical functional areas are described in the sections following the figure.

Figure 1–1, VMEbus Async/Printer Controller Block Diagram



1.2.1 Local Microprocessor and Instruction RAM

An M68020 microprocessor is used in the VMEbus Async/Printer Controller because of its fast instruction execution capability. The M68020 microprocessor operates on a local data bus with 256 Kbytes of fast static Instruction RAM (IRAM) that enable zero wait-state operation. The zero wait-state RAM enables the firmware to execute with minimal latency.

Byte parity is maintained on the IRAM, and if a parity error is detected, an interrupt is sent to the M68020 microprocessor. The parity error interrupt level, and the parity error interrupt vector are programmed via the VMEbus array.

1.2.2 Bootstrap EPROM

The 32 Kbyte bootstrap EPROM is contained on the controller and is accessible on the upper byte of the M68020 microprocessor data bus. The EPROM is the only data resource that is not accessible to the VMEbus in the slave mode.

VMEbus slave mode accesses are mapped into the lower 32 Kbytes of instruction RAM when the EPROM is mapped in. This is a result of the dynamic bus sizing capability of the M68020 microprocessor and the byte implementation of the EPROM hardware. This also provides a straight-forward method of downloading and dumping the controller instruction RAM.

The EPROM is mapped to the lowest portion of address space, but can be mapped out via a control register in the VMEbus array. The EPROM contains the initial boot code as well as various diagnostic self-test routines that verify the download capability.

To verify the complete functionality of the controller, diagnostic software is downloaded via the VMEbus I/O Processor (VIOP). The run-time executive and communication device drivers are also loaded via the VIOP.

1.2.3 Local Resource and Communications Device Interface

The M68020 microprocessor has specific capabilities that enable it to execute its' kernel executive as well as manage the I/O device and VMEbus interfaces. These capabilities are described in the following sections.

1.2.3.1 Timers

Two internal timers are contained in the VMEbus array for schedule driven tasks. These timers are typically used in applications ranging from executive scheduling to network timing. The timers are configured as count down timers and generate interrupts when their counts go to zero. The interrupts are enabled or disabled via their respective interrupt bits in the Interrupt Enable Register (IER).

Each timer is capable of using one of four internal frequencies that are divided down from an external clock source. An external clock source and internal scaling, as well as a programmable preset value, provide the most common time intervals.

Four modes of timer operation are supported:

1. **Hold**—Causes the timer to remain at its' present count.
2. **Single shot**—The pre-programmed value counts to zero and causes an interrupt.
3. **Multi shot**—The counter generates an interrupt upon reaching a zero, then re-programs itself with the pre-programmed value and begins counting again.
4. **Initialization**—The base count is constantly loaded into the present count. The change from initialization mode to any other mode enables the timer function.

A timer is disabled by either disabling its' interrupt in the IER register or by setting the timer to mode 1 or 2.

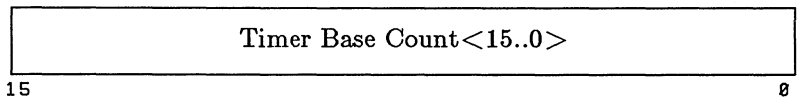
1.2.3.2 Register Definitions

Each timer has three registers associated with it:

- Base Count Register (BCR)
- Present Count Register (PCR)
- Timer Control Register (TCR)

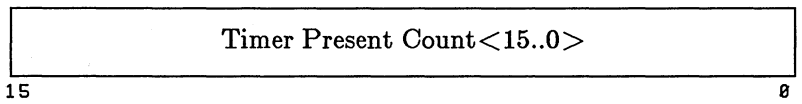
The first register is the sixteen bit, read/write Base Count Register. It contains a count that is decremented to zero to generate an interrupt. The format of the BCR is illustrated in Figure 1–2, "Timer Base Count Register":

Figure 1–2, Timer Base Count Register



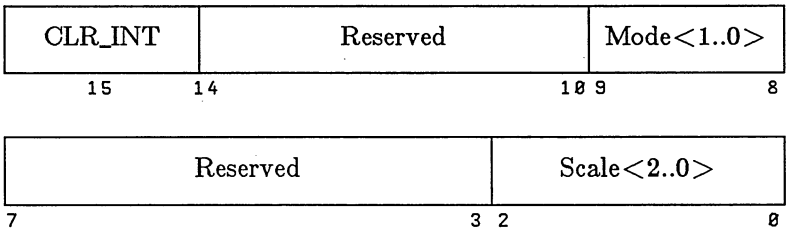
The second register is the read-only Present Count Register. It contains the present count value for the timer. The format of the PCR is illustrated in Figure 1–3, "Timer Present Count Register":

Figure 1–3, Timer Present Count Register



The third register is the Timer Control Register. The format of the TCR is illustrated in Figure 1–4, "Timer Control Register":

Figure 1–4, Timer Control Register



The bits in the TCR have the following definitions:

- **CLR_INT**—Clears the pending interrupt for the timer
- **Mode<1..0>**—These bits select the mode of operation as described in Table 1–1, "Timer Mode Selection":

Table 1–1, Timer Mode Selection

Mode<1>	Mode<0>	Operational Mode
0	0	Hold
0	1	Single shot
1	0	Multi shot
1	1	Initialization

- **Scale<2..0>**—These bits control the timer scaling factor as described in Table 1–2, "Timer Scaling Selection":

Table 1–2, Timer Scaling Selection

Scale<2>	Scale<1>	Scale<0>	Scaling Factor
0	0	0	External divided by 2
0	0	1	External divided by 4
0	1	0	External divided by 8
0	1	1	External divided by 16
1	x[1]	x	External divided by 1

[1] Not required

1.2.4 Bus Arbitration Timeout

When the VMEarray is attempting to acquire control of the VMEbus, it asserts the BUSREQUEST line to inform the VMEbus slot 1 master of its' requirement to use the VMEbus. The capability exists, under hardware failure mode, that the VMEbus does not respond to this

request. Alternatively, under a heavily loaded VMEbus system, the arbitration timing might be so long that software driver modification might be required.

To better inform both the test operator and the firmware designer of problems, a VMEbus local arbitration timer is implemented in the VMEarray. The timer begins counting when the VMEbus is requesting and stops counting when the VMEbus grant is obtained. In the event that the maximum count is achieved before the VMEbus is granted to the VMEarray, two possible operations can occur:

- If the VMEarray is requesting a VMEbus transfer and the arbitration timer expires, this fact is reported in the DMA error register and the operation is flagged as being terminated with error. This assertion interrupts the M68020 under the normal DMA termination interrupt.
- If the M68020 was requesting the VMEbus, the bus error is asserted, and the BAT error assertion in the Buserror Source Register (BSR) is set.

The VMEbus Arbitration Timeout values located in the TMOT register are listed in the following table:

Table 1–3, Timeout Register Values

BAT1	BAT0	Timeout Period
0	0	256 microseconds
0	1	1 millisecond
1	0	4 milliseconds
1	1	timer disabled

1.2.4.1 Asynchronous Communications

Asynchronous communications directly support 16 ports, each having full modem control, including Data Carrier Detect (DCD). The Signetics OCTAL UART (SCC2698) is used to implement the communication function for eight ports with discrete registers that provide the Data Set Ready (DSR) and Data Terminal Ready (DTR) functions. DCD interrupts are channeled through the OCTAL UARTs via their DCD pins.

The remaining eight ports are driven from Dual Universal Serial Communications Controller (DUSCC) chips that perform the asynchronous communication functions.

The RS-232 driver/receivers are located on a panel interface card that has 25-pin sub-miniature D connectors used for terminal and modem connections. This interface panel enables different physical interfaces to be used by swapping an interface card.

1.2.5 Printer Interface

The printer interface supports a transfer rate of 300 Kbaud per second. The interface is a Dataproducts printer interface, but it can be programmed to provide a Centronix interface. Software programmability provides maximum customer flexibility by enabling the Dataproducts and Centronixs printers to be switched via the */ioconfig* file.

The printer interface uses a 1 Kbyte FIFO between the microprocessor and the printer to ensure that M68020 microprocessor program execution is not degraded.

The physical connection from the controller to the printer is via a 34-pin cable. The VMEbus Async/Printer controller has open-collector drivers that are used to transfer the data to the printer.

1.3 Dual Port RAM

The VMEbus Async/Printer controller contains 256 Kbytes of common data RAM. This RAM can be accessed by the VMEbus interface as well as the M68020 microprocessor. The dual port nature of the RAM allows concurrent accesses without degrading the performance of the microprocessor. Byte parity is maintained on data in the Dual Ported RAM (DPR).

1.4 VMEbus Array

The VMEbus interface is implemented in a CMOS 20k gate array, as outlined in the following sections. These sections represent logical functions that are implemented in the VMEbus array.

1.4.1 VMEbus Interface

The primary purpose of the VMEbus array is to perform all of the necessary VMEbus interface functions under both slave and master mode. These characteristics do not include slot 1 functions, but do include bus requester, bus master, bus interrupt requester, with all modes supported excluding Unaligned Address Transfer (UAT) mode. The interrupt level and interrupt vector are programmable.

The VMEbus interface drivers are not on the VMEbus array because of the current driving capabilities required by the VMEbus specification.

VMEbus array DMA control allows data transfers in either standard or extended addressing modes, as well as the capability for the VIOP to access internal VMEbus array registers under Short Supervisory mode. Under slave mode, the VMEbus interface responds to address modifiers 3D, 39, 0D, or 09.

1.4.2 DMA Controller

Although the CONVEX VMEbus Bus Control Unit does not support block transfers, high data rates may be achieved by operating the VMEbus with a DMA controller directly linked to the VMEbus interface hardware. This DMA controller is capable of performing single cycle dual address block transfers in either standard or extended VMEbus address space between:

- VMEbus and Dual Ported RAM (DPR)
- VMEbus and Instruction RAM (IRAM)

Optionally, single cycle direct acknowledge DMA operations are supported between DPR and a block character device. The desired DMA VMEbus transfer is 30 Mbytes per second, although the VIOP/VBCU interface is only capable of 10 Mbytes per second. The programmers model for the DMA controller is consistent with other DMA type devices as described in the VMEbus array (VMEarray) specifications.

Under programmable control, the VMEbus array DMA controller generates address modifiers 3D, 39, 0D, or 09 when performing VMEbus master data transfers.

1.4.3 Dual Ported RAM Control

All control and arbitration for the Dual Ported RAM (DPR) is contained in the VMEbus array. This arbitration favors the M68020 microprocessor in conflicting accesses, allowing maximum instruction execution speed. Microprocessor DPR accesses are address multiplexed through the VMEbus array, where the array arbitrates the use of the DPR for the M68020 microprocessor. VMEbus accesses use external data latching for VMEbus accesses, with arbitration and data steering controlled via the VMEbus array.

1.4.4 Interrupt Control

All interrupts are channeled through the VMEbus array, allowing all interrupt vectors to be software programmable. This scheme allows for 20 external interrupts, as well as 12 internal VMEbus array interrupts, providing a total of 32 total microprocessor interrupts. The microprocessor interface includes an Interrupt Enable Register, an Interrupt Status Register, and diagnostic or scheduling Force Interrupt Register.

1.4.5 Local Control

Microprocessor accesses are decoded into address regions externally, with the decodes passed into the VMEbus array. This allows for fast instruction RAM accesses, as well as for fast internal bus arbitration. The VMEbus array provides the BUSERROR for accesses in which no devices are responding, essentially the DSACK timeouts.

1.5 M68020 Microprocessor Addressable Registers

The VMEbus Async registers are divided into the following three groups:

1. Internal registers for the VMEbus array
2. Internal registers associated with the discrete VLSI communication and control devices
3. Discrete registers implemented on the controller in MSI logic

Because the VMEbus array registers are detailed in the VMEarray specification, they are not redefined here. Also, the internal registers of the VLSI devices are detailed in the respective data sheets and are not redefined here.

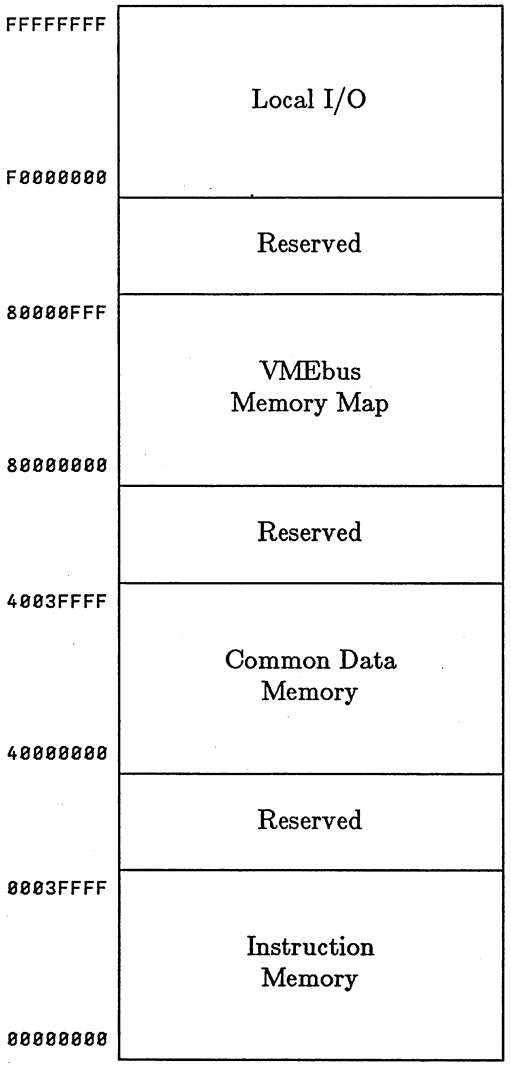
For the remaining discrete registers, a thorough discussion is provided. Memory maps, however, are detailed for all programmer visible registers accessible to the controller.

The VMEbus Async address space is divided into the following four distinct areas:

1. Local I/O registers
2. VMEbus access
3. Common Data RAM
4. Instruction RAM

The memory map of the address space is shown in Figure 1–5, "M68020 Microprocessor Memory Map":

Figure 1–5, M68020 Microprocessor Memory Map



This memory map architecturally defines 1 Gbyte of space for each distinct area. Future

expansion will be into the reserved areas, as the required chip technology develops. The remaining sections detail the M68020 microprocessor I/O area.

1.6 Internal Device Registers

The VMEbus Async internal device registers are divided into the following three groups:

1. VMEbus array registers
2. Octal UART registers
3. Dual universal serial communications controller registers

1.6.1 VMEbus Array Registers

The VMEbus array registers, detailed in the VMEbus array (VMEarray) specifications, are located at addresses F0000000 through F00000FE. Figure 1–6, "Interrupt Enable Register" indicates the external connections from the board to the VMEbus array. The mapping for the Interrupt Status Register and Interrupt Force Register are identical. All other registers are fully defined in the VMEarray specifications.

Figure 1–6, Interrupt Enable Register

NMI	INT A	INT B	VME DMA	IO3 DMA	IO2 DMA	IO1 DMA	IO0 DMA
31	30	29	28	27	26	25	24

res	VME IACK	TMR A	TMR B	NC	NC	NC	NC
23	22	21	20	19	18	17	16

NC	NC	NC	LCL PAR	IF PAR	DPR PAR	IRAM PAR	PNTR
15	14	13	12	11	10	9	8

OUA 4	OUA 3	OUA 2	OUA 1	DSC 4	DSC 3	DSC 2	DSC 1
7	6	5	4	3	2	1	0

The bits in the Interrupt Enable Register have the following definitions:

- **NMI, Non-maskable Interrupt**—Sent from the VIOP to the controller, always at level 7. The NMI cannot be disabled through the interrupt enable register.
- **INT A, Interrupt A**—Sent from the VIOP to the controller. User-definable function.
- **INT B, Interrupt B**—Sent from the VIOP to the controller. User-definable function.
- **VMEbus DMA Completion/Termination Interrupt**—From the VMEbus DMA controller in the VMEbus array indicating the requested DMA operation has completed. Status must be examined to determine error-free termination.

- **IO3 DMA, Completion/Termination Interrupt**—From the Channel 3 IO DMA controller, indicating a transfer from/to an IO external device to/from the DPR is complete. The status must be examined.
- **IO2 DMA, Completion/Termination Interrupt**—From the Channel 2 IO DMA controller, indicating a transfer from/to an IO external device to/from the DPR is complete. The status must be examined.
- **IO1 DMA, Completion/Termination Interrupt**—From the Channel 1 IO DMA controller, indicating a transfer from/to an IO external device to/from the DPR is complete. The status must be examined.
- **IO0 DMA, Completion/Termination Interrupt**—From the Channel 0 IO DMA controller, indicating a transfer from/to an IO external device to/from the DPR is complete. The status must be examined.
- **VMEbus IACK**—A requested VMEbus interrupt to the VIOP has been acknowledged.
- **TMR A, Timer A**—Counters have reached terminal count and issued a periodic interrupt to the processor. The timer mode indicates whether further interrupts can be issued.
- **TMR B, Timer B**—Counters have reached terminal count and issued a periodic interrupt to the processor. The timer mode indicates whether further interrupts can be issued.
- **NC, Not Connected**—Available for scheduling interrupts through Force Interrupt Register.
- **LCL PAR, Local Parity**—Error on interface from data panel to controller.
- **IF PAR, Interface Parity Error**—Data parity error detected from controller to panel.
- **DPR PAR, Data Parity**—Parity error in the Common Data RAM (CDR).
- **IRAM PAR, Parity Error**—Parity error detected in the M68020 microprocessor instruction RAM.
- **PNTR**—The printer FIFO is empty and can accept up to 2 Kbytes of data.
- **OUA4**—Attention request from channels G and H of the OCTAL UART.
- **OUA3**—Attention request from channels E and F of the OCTAL UART.
- **OUA2**—Attention request from channels C and D of the OCTAL UART.
- **OUA1**—Attention request from channels A and B of the OCTAL UART.
- **DSC4**—Attention request from DUSCC device number 4.
- **DSC3**—Attention request from DUSCC device number 3.
- **DSC2**—Attention request from DUSCC device number 2.
- **DSC1**—Attention request from DUSCC device number 1.

1.6.2 Octal Uart Registers

Table 1–4, "Octal Uart Register" lists the register locations for the Octal Uart controlling ports 0 through 7 that are accessible from the M68020 microprocessor. The Octal Uart registers are located from address 0xF0000500 to 0xF000053f.

Table 1–4, Octal Uart Register

Register-Read	Register-Write	Address	Bit Width
Mode register 1a,Mode register 2a	Mode register 1a,Mode register 2a	F0000500	8
Status register a	Clock select register a	F0000501	8
Reserved[1]	Command register a	F0000502	8
Rx holding register a	Tx holding register a	F0000503	8
Input port change register A	Auxiliary control register A	F0000504	8
Interrupt status register A	Interrupt mask register A	F0000505	8
Counter/Timer upper A	Counter/Timer upper register A	F0000506	8
Counter/Timer lower A	Counter/Timer lower register A	F0000507	8
Mode register 1b,Mode register 2b	Mode register 1b,Mode register 2b	F0000508	8
Status register b	Clock select register b	F0000509	8
Reserved[1]	Command register b	F000050A	8
Rx holding register b	Tx holding register b	F000050B	8
Reserved[1]	Reserved[1]	F000050C	8
Input Port A	Output port configuration register A	F000050D	8
Start C/T A	Reserved[1]	F000050E	8
Stop C/T A	Reserved[1]	F000050F	8

[1] Reserved registers should never be read during normal operation since they are reserved for internal diagnostics.

The listed registers are repeated at addresses F0000510-51F, F0000520-52F, F0000530-53F.

1.6.3 Dual Universal Serial Communications Controller Registers

Table 1–5, "DUSCC Register" lists the register locations for the first SCN68562 DUSCC that are accessible from the M68020 microprocessor. Mnemonics are also listed, as well as the size of the register.

Table 1–5, DUSCC Register

Register Name	Mnemonic	Address	Bit Width
Channel Mode Register 1	CMR1	F0000400	8
Channel Mode Register 2	CMR2	F0000401	8
SYN 1/Secondary Address 1 Register	S1R	F0000402	8
SYN 2/Secondary Address 2 Register	S2R	F0000403	8
Transmitter Parameter Register	TPR	F0000404	8
Transmitter Timing Register	TTR	F0000405	8
Receiver Parameter Register	RPR	F0000406	8
Receiver Timing Register	RTR	F0000407	8
Counter/Timer Preset Register, high	CTPRH	F0000408	8
Counter/Timer Preset Register, low	CTPRL	F0000409	8
Counter/Timer Control Register	CTCR	F000040A	8
Output and Miscellaneous Register	OMR	F000040B	8
Counter/Timer, high	CTH	F000040C	8
Counter/Timer, low	CTL	F000040D	8
Pin Configuration Register	PCR	F000040E	8
Channel Command Register	CCR	F000040F	8
Transmitter FIFO	TxFIFO	F0000410-13	8
Receiver FIFO	RxFIFO	F0000414-17	8
Receiver Status Register	RSR	F0000418	8
Transmitter and Receiver Status Register	TRSR	F0000419	8
Input and Counter/Timer Status Register	ICTSR	F000041A	8
General Status Register	GSR	F000041B	8
Interrupt Enable Register	IER	F000041C	8

The listed registers are repeated at addresses F0000420-43F, F0000440-45F, F0000460-47F,F0000480-49F, F00004A0-4BF, F00004C0-4DF, F00004E0-4FF .

1.7 Discrete Registers

Table 1–6, "Discrete Registers" lists the register locations for the device and VMEbus board control that are accessible from the M68020 microprocessor. Mnemonics are also listed, as well as the size of the register.

Table 1–6, Discrete Registers

Register Name	Mnemonic	Address	Bit Width
Data Terminal Ready Register	PDTR	F0000540	16
Data Set Ready Register	PDSR	F0000542	16
Printer Control Port Register	PCPR	F0000544	16
Printer Data Port Register	PDPR	F0000546	8
Interface Parity Clear Register	IPCR	F0000547	xx
Board Base Address Register	BBAR	F0000800	16
LED Display Register	LEDS	F0000802	16
Parity Error Clear Register	PERR	F0000804	xx
Cache Disable Status Register	CDSR	F0000806	1

1.7.1 Data Terminal Ready Register

Each async port receives a Data Terminal Ready (DTR) signal from the respective modem/terminal to which it is connected. These signals may be used to initiate control over the port. Upon detection of DTR, the port is assumed to be connected to a *ready* device. The Data Terminal Ready Register (PDTR) is a read-only register. Figure 1–7, "Data Terminal Ready Register" shows the format of the register:

Figure 1–7, Data Terminal Ready Register

DTR 15	DTR 14	DTR 13	DTR 12	DTR 11	DTR 10	DTR 09	DTR 08
15	14	13	12	11	10	9	8

DTR 07	DTR 06	DTR 05	DTR 04	DTR 03	DTR 02	DTR 01	DTR 00
7	6	5	4	3	2	1	0

1.7.2 Data Set Ready Register

The on-line capability for the controller is signaled to the modem/terminal by the Data Set Ready (DSR) line. This line is controlled by the Data Set Ready Register (PDSR), a write/read register. Figure 1–8, "Data Set Ready Register" shows the format of the register:

Figure 1–8, Data Set Ready Register

DSR 15	DSR 14	DSR 13	DSR 12	DSR 11	DSR 10	DSR 09	DSR 08
15	14	13	12	11	10	9	8

DSR 07	DSR 06	DSR 05	DSR 04	DSR 03	DSR 02	DSR 01	DSR 00
7	6	5	4	3	2	1	0

1.7.3 Printer Control Port Register

The Printer Control Port Register (PCPR) manipulates the printer interface controller state machine in such a manner as to efficiently transfer data to the cabled printer. Figure 1–9, "Printer Control Port Register" shows the format of this register:

Figure 1–9, Printer Control Port Register

Reserved	Full	Empty	Req Pend	PE	SLCT
15 13	12	11	10	9	8

Reserved	Pntr En	Daprl	Pntr Reset
7 3	2	1	0

The most significant byte of the register contains status information from the printer interface. The least significant byte contains command bits that control the interface.

> **NOTE**
>
> The most significant byte of the Printer Control Port Register is read only.

The bits in the Printer Control Port Register have the following definitions:

- **Reserved**—Reserved for future use.
- **Full**—When asserted, this bit indicates the FIFO is full.
- **Empty**—When asserted, this bit indicates the FIFO is empty. However, SLCT must be asserted and PE de-asserted before the printer can receive new data from the FIFO.
- **Req Pend**—This bit is the decoded request from the printer that the printer is ready to receive data.

- **PE**—When asserted, this bit indicates a potential error in the printer mechanics. This bit is set after the CHECK indicator on the printer front panel is illuminated.
- **SLCT**—When asserted, this bit indicates that the printer is online.
- **Reserved**—Reserved for future use.
- **Pntr En**—Enables printer interrupts when the FIFO is empty or an error occurs.
- **Daprl**—When asserted, the printer is placed in the Centronix interface compatible mode. When cleared, the printer is placed in the Dataproducts mode.
- **Pntr Reset**—To recover from a printer error condition, the FIFO may need to be cleared. When asserted, this bit clears the FIFO and resets the printer data port.

1.7.4 Printer Data Port

Printer data is deposited in a 74ACT7202 FIFO in order to release the M68020 microprocessor of data transfer as soon as possible. This 1K x 9 FIFO is emptied by the Printer Interface Controller, a hardware state machine which empties all data from the FIFO as the printer requests the respective data. The Printer Data Port Register (PDPR) is a byte wide register which receives data from the M68020 microprocessor data bus lines 31 through 24, where bit <31> is the most significant bit in the transmitted character.

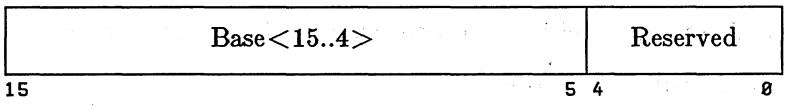
1.7.5 Interface Parity Clear Register

The Interface Parity Clear Register (IPCR) is used to clear parity error assertions captured at the end of the cable interface, occurring on data parity problems where the data is sent to an I/O device. A data independent write to this address clears the interrupt assertions, which can be read in the Interrupt Status Register.

1.7.6 Board Base Address Register

The Board Base Address Register (BBAR) contains the base address of the VMEbus controller as seen by the VIOP running Short Supervisory VMEbus cycles. This value is configured using jumpers and read by the M68020 microprocessor which then programs the internal VMEbus array Supervisory Base Address Register. Figure 1–10, "Board Base Address Register" shows the format of the register.

Figure 1–10, Board Base Address Register

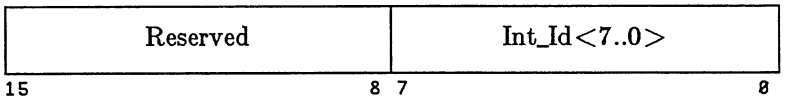


1.7.7 LED Display Register

The LED Display Register (LEDS) controls eight discrete LEDs located on the front edge of the controller. The LEDs indicate status as well as port activity. These eight LEDs are controlled by

the software in such a manner as to relay useful information about board activity. Figure 1–11, "LED Display Register" shows the format of the register:

Figure 1–11, LED Display Register



1.7.8 Parity Error Clear Register

The Parity Error Clear Register (PERR) captures parity error assertions received by the local controller, either via instruction RAM, common data RAM, or interface read operations. A data independent write to the PERR register clears these interrupt assertions.

1.7.9 Cache Disable Status Register

The Cache Disable Status Register (CDSR) provides the status of the 68020 cache disable input pin. If bit <0> is a 1, then the cache is disabled, if bit <0> is 0, the cache is not disabled. To disable the 68020 on-board cache, pin 12 of the U086A4 must be grounded.

1.8 VMEbus Accessible Registers

Table 1–7, "VMEbus Accessible Registers" lists the register locations for the VMEbus array accessible from the VMEbus. Mnemonics as well as the size of the register are also listed.

Table 1–7, VMEbus Accessible Registers

Register Name	Mnemonic	Address	Bit Width
Reset Register, MS byte	RESET_CTL_MS	xx01	8
Reset Register, LS byte	RESET_CTL_LS	xx03	8
VME LOCAL Map Register, MS byte	VME_LOCAL_MS	xx05	8
VME LOCAL Map Register, middle byte	VME_LOCAL_MID	xx07	8
VME LOCAL Map Register, LS byte	VME_LOCAL_LS	xx09	8
VME Address Match Register, MS byte	VME_AD_MATCH_MS	xx0B	8
VME Address Match Register, middle byte	VME_AD_MATCH_MID	xx0C	8
VME Address Match Register, LS byte	VME_AD_MATCH_LS	xx0E	8
VME Address Modifier Match Register	VME_AM_MATCH	xx11	8

These registers are defined in the VMEarray specifications, and therefore are not redefined here.

1.9 VMEbus Async/Printer Specifications

The specifications for the CONVEX VMEbus Async/Printer Controller are listed in the following table:

Table 1–8, VMEbus Async/Printer Controller Specifications

Parameter	Value/Comment
Width	14.43 in. (36.65 cm)
Height	6.29 in. (15.98 cm)
Thickness	0.65 in. (*approx*) (1.65 cm)
Weight	3 lb. (*approx*) (1.36 Kg)
Power Dissipation, Maximum	50 W
Temperature Range,[1] Maximum	32 °F to 104 °F (0 °C to 40 °C)
Temperature Range,[1] Recommended	70 °F to 80 °F (21 °C to 26.6 °C)
Temperature Change, Maximum Rate	14.4 °F/hr (8 °C/hr)
Humidity Range, Maximum	10% to 95% with no condensation
Humidity Range, Recommended	20% to 80% with no condensation

[1] At altitudes above 3,280 ft (1,000 m), lower air densities affect air conditioning. If the unit is located above this altitude, lower the recommended temperature value by 1 °F/1,000 ft (2 °C/1,000 m).

The VMEbus Async/Printer controller is physically connected to the panel by a 100-wire cable. The cable is used to pass address, data, control, and interrupt signals between the controller and a communication device. The cable is also used to provide power to the panel and daughter cards.

The daughter cards are configured to handle four asynchronous RS-232 ports. The four daughter cards physically bolt to the panel and provide connections for sixteen asynchronous communication devices.

1.10 LED and Configuration Descriptions

The VMEbus Async/Printer controller and panel assembly LEDs and jumper configurations are described in the following sections.

1.10.1 Controller LED and Configuration Descriptions

The VMEbus Async/Printer controller LEDs are listed in the following table:

Table 1–9, Async/Printer Controller LED Descriptions

LED	Position	Description
HEARTBEAT	0x01	1 HZ heartbeat
IDLE	0x02	Executing scheduler idle loop
VMEDMA	0x04	VMEbus DMA request pending
TXQFULL	0x08	Controller send queue full
DUSCCSVC	0x10	Servicing DUSCC interrrupt
OCTARTSVC	0x20	Servicing octal UART interrupt
SPURINT	0x40	Toggle on spurious interrupt
SYSFAIL	0x80	VMEbus SYSFAIL asserted

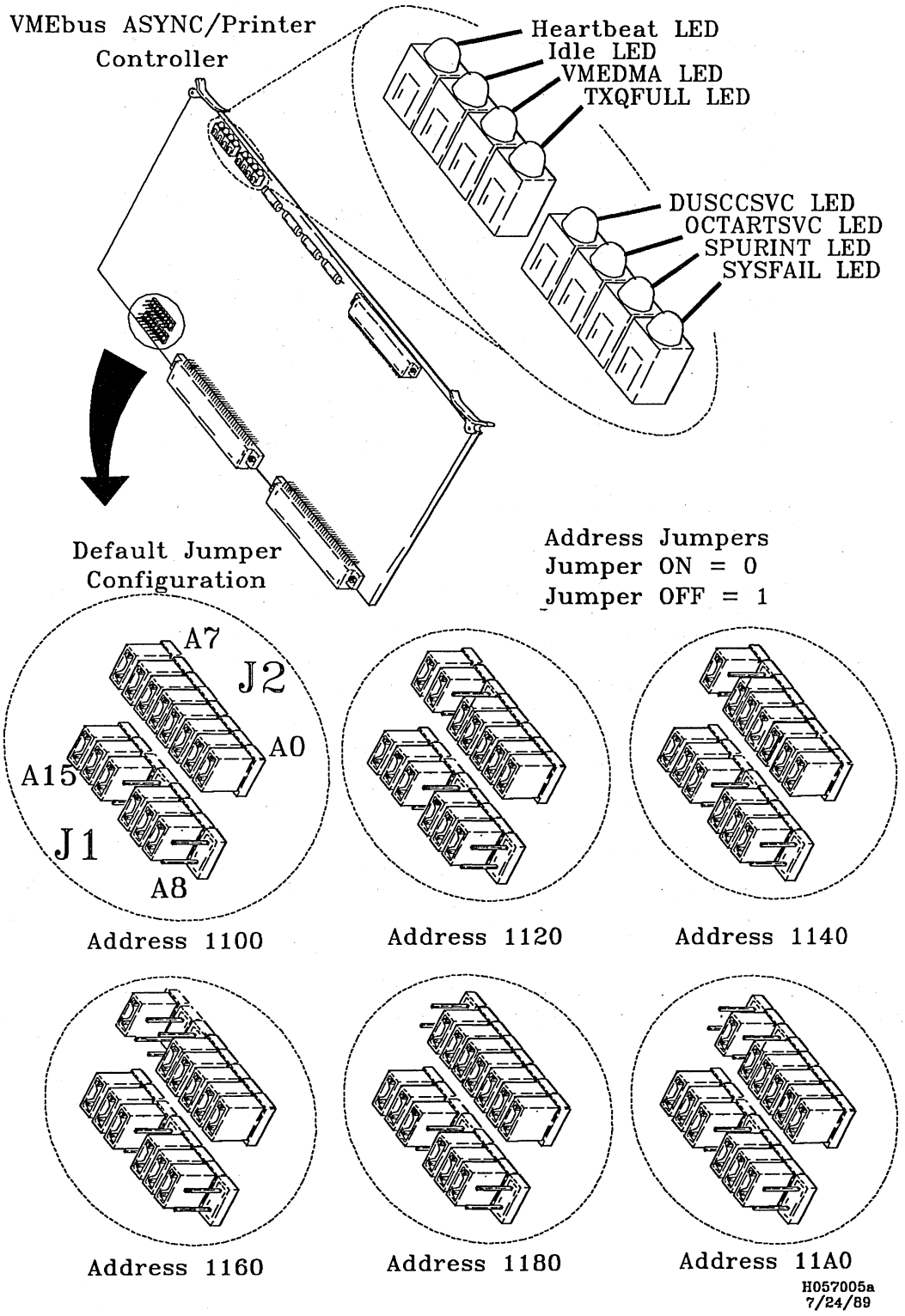
Table 1–10, "Controller Address and Interrupt Levels" lists the controller addresses and interrupt levels for the VMEbus Async/Printer controller.

Table 1–10, Controller Address and Interrupt Levels

Controller Number	Address	Interrupt Level	Bus Request Level
0	0x1100	7	3
1	0x1120	Not assigned	3
2	0x1140	Not assigned	3
3	0x1160	Not assigned	3
4	0x1180	Not assigned	3
5	0x11A0	Not assigned	3

The VMEbus Async/Printer controller LEDs and jumper configurations are shown in Figure 1–12, "Async/Printer Controller LEDs and Address Jumpers":

Figure 1–12, Async/Printer Controller LEDs and Address Jumpers



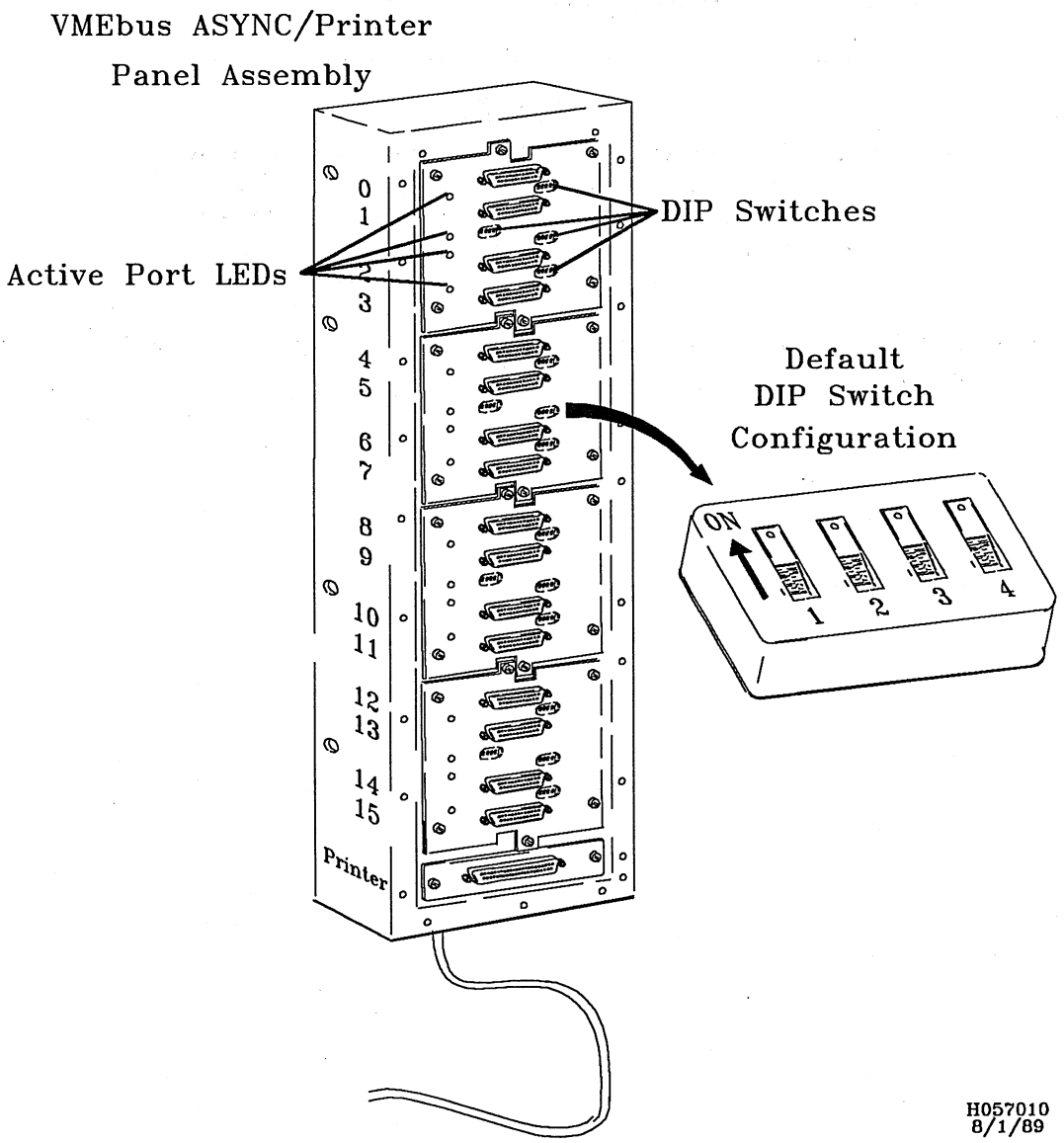
1.10.2 Panel Assembly LED and Configuration Descriptions

A VMEbus Async/Printer panel assembly LED indicates an active port. The LED for a port is illuminated when the operating system opens the port.

> **NOTE**
>
> The dip switches on all daughter cards should be set to the off position for asynchronous operation. The dip switchs are provided for use in synchronous communications in future applications.

The LEDs and dip switches on the VMEbus Async/Printer panel assembly are shown in Figure 1–13, "Async/Printer Panel Assembly LEDs and Switches":

Figure 1–13, Async/Printer Panel Assembly LEDs and Switches



Chapter 2
Unpacking and Installation

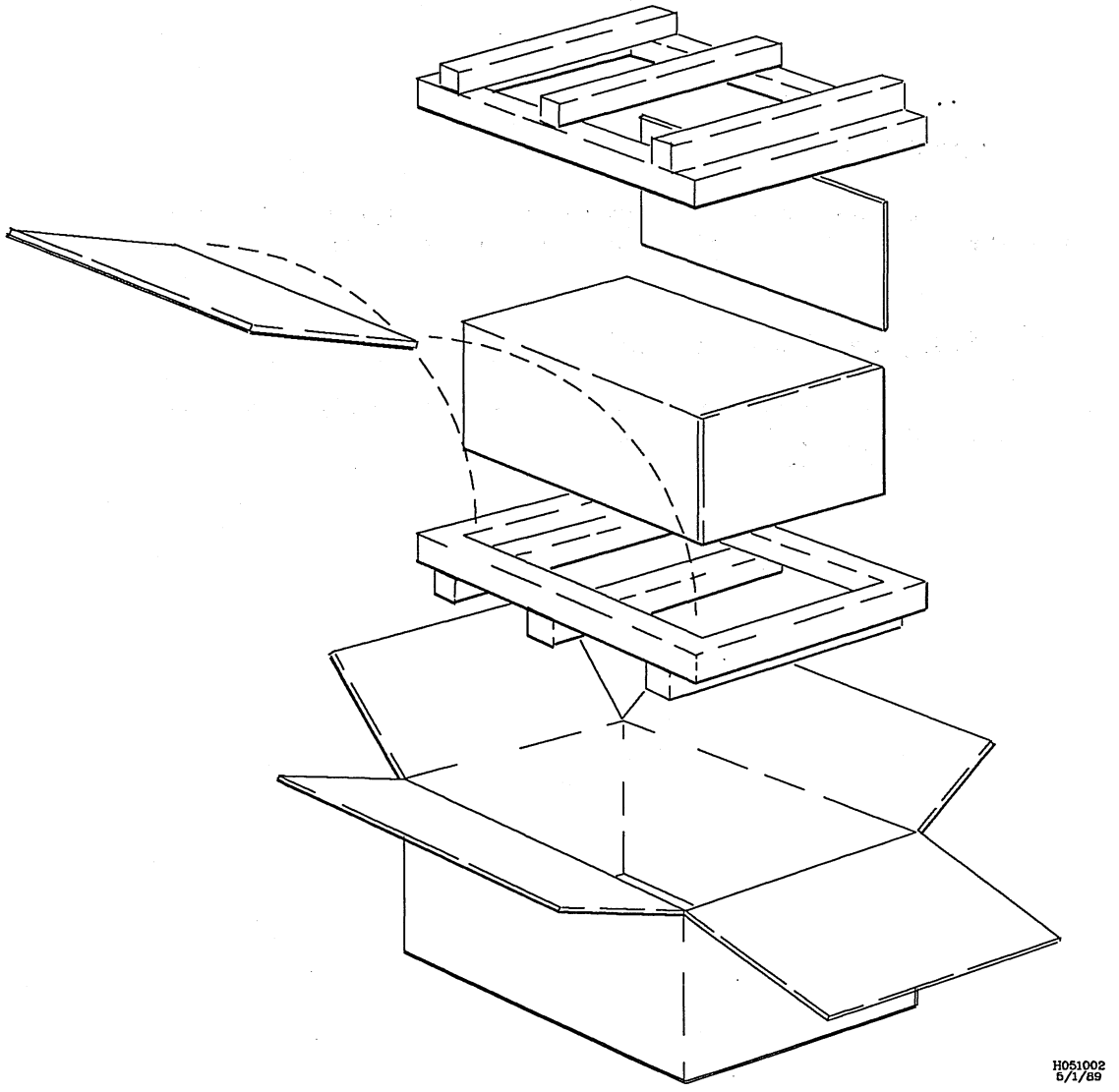
2.1 Overview

Unpacking and inspection are discussed, major components of the CONVEX VMEbus Async/Printer controller assembly are identified, and installation procedures are provided.

2.1.1 Shipping Configuration

A typical VMEbus Async/Printer controller shipping configuration consists of the items illustrated in Figure 2–1, "Shipping Carton":

Figure 2–1, VMEbus Async/Printer Controller Shipping Carton



2.1.2 Inspection for Damage

All shipping containers have been specially designed to protect their contents under normal shipping conditions. Carefully inspect each carton for signs of shipping damage as it is unpacked. If damage is found after visual inspection, document the damage with photographs and contact the transport carrier immediately. Unpack the equipment as described in the next section.

2.1.3 Electrostatic Discharge (ESD) Damage

Typically ESD damage occurs to electronic circuit boards during handling. Static charges take place when various objects are separated or rubbed together, often creating a high voltage level charge. If a high voltage level charge is discharged into electronic computer circuits it will damage the electronic components. The main factors that determine a voltage level charge are:

- Types of materials
- Relative humidity
- Rate of change or separation

Table 2–1, "Static Charge Levels and Relative Humidity" provides an approximation of electrostatic charge levels based on various personnel activities and humidity levels:

Table 2–1, Static Charge Levels and Relative Humidity

Personnel Activity[1]	Humidity[2] & Charge Levels (Volts[3])			
	26%	32%	40%	50%
Person walking across linoleum floor	6,150V	5,750V	4,625V	3,700V
Person walking across carpet	18,450V	17,250V	13,875V	11,100V
Person getting up from a plastic chair	24,600V	23,000V	18,500V	14,800V

1 Source: B. A. Unger, *Electrostatic Discharge Failures of Semiconductor Devices* (Bell Laboratories, 1981).
2 A high rate of air flow produces higher static charges than a low air flow rate, for the same relative humidity level.
3 Some data in this table has been extrapolated.

Table 2–2, "Components Susceptibility to ESD Damage" lists various components and their susceptibility to static damage:

Table 2–2, Components Susceptibility to ESD Damage

Susceptibility Ranges of Various Devices Exposed to Electrostatic Discharge (Human Body Model[1])	
Device Type	**Level of ESD Susceptibility (Volts)**
MOSFET	>10
JFET	> 140
CMOS	> 250
Schottky Diodes, TTL	> 300
Bipolar Transistors	> 380
ECL (For Hybrid use, PCB level)	> 500
SCR	> 680

[1] Source: B. A. Unger, *Electrostatic Discharge Failures of Semiconductor Devices* (Bell Laboratories, 1981).

2.1.4 Unpacking

The customer's bill of material lists all VMEbus Async/Printer equipment shipped from CONVEX. It should be used as a checklist to ensure that all equipment has arrived. Refer to Table 2–3, "Typical VMEbus Async/Printer Controller Bill of Material" for a bill of material for the VMEbus Async/Printer controller assembly:

Table 2–3, Typical VMEbus Async/Printer Controller Bill of Material

Product Number	Description	Quantity
550-000117-210	VMEbus Async/Printer assembly[1]	1
410-001193-200	VMEbus Async/Printer controller	1
500-000276-200	VMEbus Async/Printer panel assembly	1

[1] An assembly contains a complete set of CONVEX Async/Printer hardware.

CAUTION

The VMEbus Async/Printer controller can be damaged by Electrostatic Discharges (ESD) during unpacking procedures. A grounded wrist strap (or other grounding method) must be used when the VMEbus Async/Printer controller is removed from its protective packing to prevent ESD damage.

1. Unpack each item of equipment from its shipping container.
2. Inspect each item of equipment for any sign of shipping damage as it is unpacked.
3. If equipment damage is found, document and proceed to the next section.

NOTE

Save all packing material until after operational checkout of the equipment. This enables equipment to be returned safely to CONVEX, if required.

2.1.5 Damage Claims

If the VMEbus Async/Printer controller or related hardware is damaged, a damage claim must be completed. Damage claims should be completed by the customer and given to the shipping representative. Claim forms are normally obtained from the shipping representative.

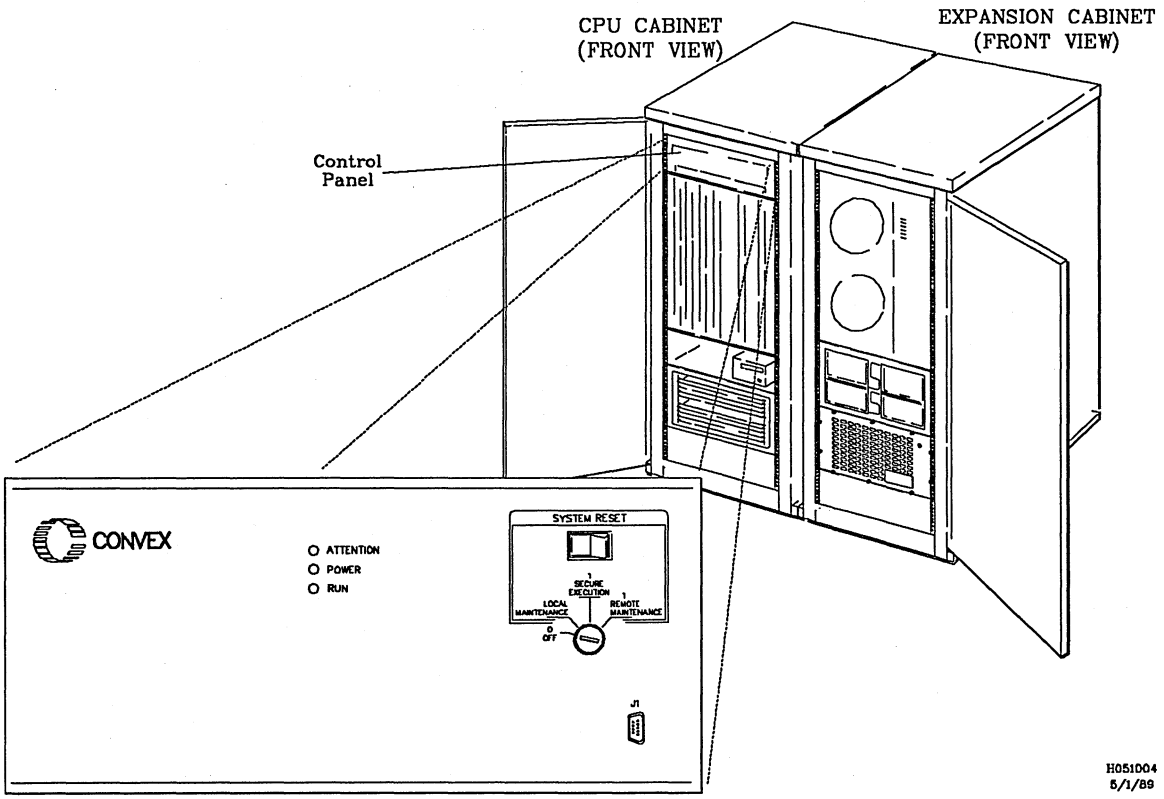
2.2 Installation Procedures

CAUTION

Failure to shut the system down before removing power to the VMEbus chassis will cause a system crash. Refer to the *CONVEX Processor Operation Guide (C100 Series, C200 Series)* for power down procedures on a CONVEX computer.

1. Turn the processor's front control panel key switch to the **OFF** position as shown in Figure 2–2, "Typical Front Panel Power Control Switch":

Figure 2–2, Typical Front Panel Power Control Switch

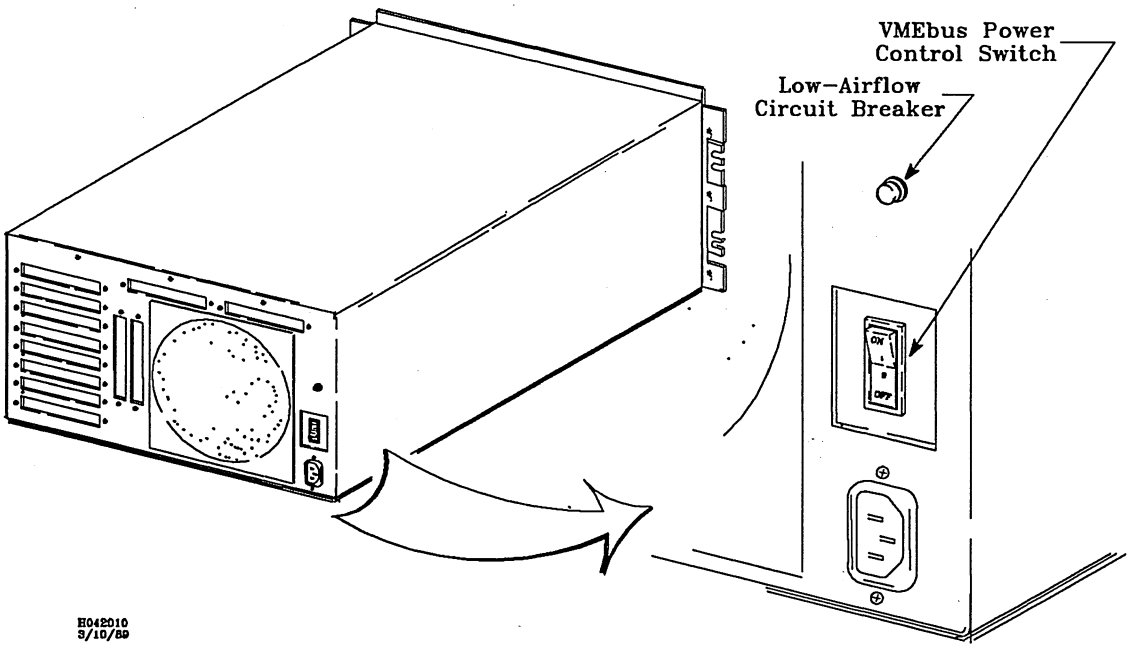


CAUTION

Failure to remove power to the VMEbus chassis before installing or removing equipment will damage electronic components.

2. Set the VMEbus chassis power control switch to the **OFF** position as shown in Figure 2–3, "VMEbus Chassis Power Control Switch":

Figure 2–3, VMEbus Chassis Power Control Switch

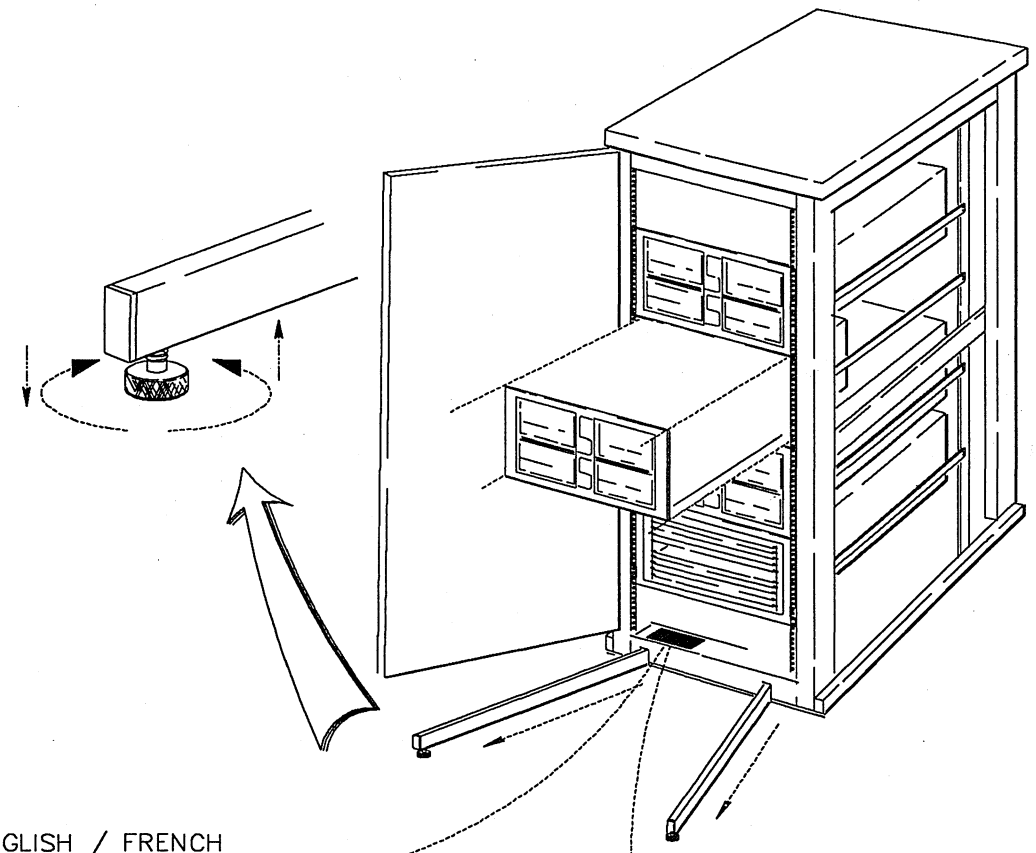


WARNING

Expansion cabinet stabilizer bars must be extended prior to installing a VMEbus chassis, or before extending the VMEbus chassis assembly from its expansion cabinet for service. Failure to do so will make the expansion cabinet unstable, increasing the possibility of it falling forward. This can cause injury to personnel and will cause damage to equipment.

3. Extend the expansion cabinet stabilizer bars and adjust feet until they are in firm contact with the floor as shown in Figure 2–4, "Expansion Cabinet Stabilizer Bars":

Figure 2–4, Expansion Cabinet Stabilizer Bars



ENGLISH / FRENCH

CAUTION	ATTENTION
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	POUR REDUIRE LE RISQUE D'ACCIDENT ATTRIBUABLE A L'INSTABILITE DE L'UNITE, DEPLOYER LES STABILISATEURS AVANT DE SORTIR LES PERIPHERIQUES. 1. POUR DEPLOYER LES STABILISATEURS, TIRER COMPLETEMENT LES BRAS ANTI-BASCULEMENT ET ABAISER LES PATTES DE FACON QU'ELLES REPOSENT SOLIDEMENT SUR LE SOL. 2. S'ASSURER QUE TOUS LES PERIPHERIQUES SON MUNIS DE VIS DE BLOCAGE. 3. NE JAMAIS SORTIR PLUS D'UN PERIPHERIQUE A UN MOMMENT DONNE.

ENGLISH / GERMAN

CAUTION	ACHTUNG
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	ZUR VERMEIDUNG VON GEFAHRDUNG DURCH EIN INSTABILES GERAT SIND VOR DER HERAUSNAHME VON PERIPHERALS DER STABILISIERUNGSMECHANISMUS BETATIGT WERDEN. 1. UM DIE STABILISIERUNGSEINRICHTUNG ZU BETATIGEN, SIND DER "ANTITILT KANAL" GANZ HERAUS ZU ZIEHEN UND DER UNTERE STUTZFUSS AUF DEN BODEN ZU FUHREN. 2. OBERPRUFEN SIE, OB IN ALLEN ANDEREN VERSCHIEBBAREN GERATEN DER SICHERUNGSMECHANISMUS BETATIGT IST. 3. ZIEHEN SIE NIE MEHR ALS EIN GERAT HERAUS.

H015020
5/1/89

4. Unlock the 2 VMEbus chassis lock screws and extend the chassis on its slides.
5. Unlock the 12 top panel lock screws on the VMEbus chassis top panel and remove the top panel.

> **NOTE**
>
> The VMEbus backplane slot positions are labeled on the front of each chassis. Cable opening numbers are stamped on the outside rear panel on all CONVEX VMEbus chassis. VMEbus controller cables exit the chassis at the rear through cable openings. Cables from a given controller should always exit the VMEbus chassis through the same cable opening.

Table 2–4, "Cable Opening Numbers for VMEbus Chassis" defines cable opening numbers and device types for the three types of CONVEX VMEbus chassis:

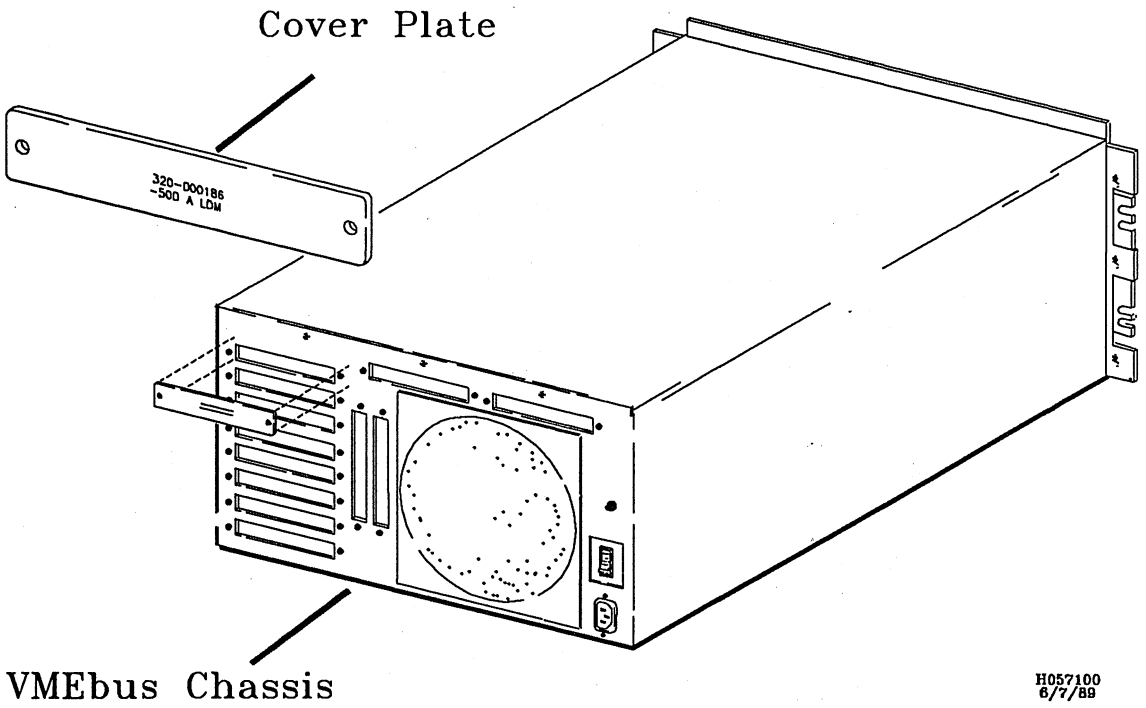
Table 2–4, Cable Opening Numbers for VMEbus Chassis

Cable Opening Number	Dual (10-slot) VMEbus	Single (9-slot) VMEbus	Combo VME/Mbus
J1	VBCU0	VBCU	VBCU
J2	VME0 Ctlr 1	VMEbus Ctlr 1	VMEbus Ctlr 1
J3	VME0 Ctlr 2	VMEbus Ctlr 2	VMEbus Ctlr 2
J4	VME0 Ctlr 3	VMEbus Ctlr 3	VMEbus Ctlr 3
J5	VME0 Ctlr 4	VMEbus Ctlr 4	VMEbus Ctlr 4
J6	VME1 Ctlr 4	VMEbus Ctlr 5	VMEbus Ctlr 5
J7	VME1 Ctlr 3	VMEbus Ctlr 6	Mbus Ctlr 3
J8	VME1 Ctlr 2	VMEbus Ctlr 7	Mbus Ctlr 2
J9	VME1 Ctlr 1	VMEbus Ctlr 7[1]	Mbus Ctlr 1
J10	unassigned	unassigned	Mbus Ctlr 0
J11	unassigned	unassigned	unassigned
J12	VBCU1	unassigned	MBCU

[1] This controller slot is reserved for the *second* board of a two-board controller; the first board, of the two-board set, must be located in the *previous* VMEbus slot.

6. Refer to Table 2–4 "Cable Opening Numbers for VMEbus Chassis" and remove the appropriate cable cover plate on the rear of the VMEbus chassis as shown in Figure 2–5, "Cable Cover Plate":

Figure 2–5, Cable Cover Plate



CAUTION

Failure to have the VMEbus chassis power control switch set to **OFF** before installing the VMEbus Async/Printer controller will damage electronic components on the controller or VMEbus chassis.

CAUTION

The VMEbus Async/Printer controller can be damaged by Electrostatic Discharges (ESD) during installation. A grounded wrist strap (or other grounding method) must be used when handling the controller to prevent ESD damage.

7. Refer to Table 2–5, "Controller Address and Interrupt Levels" and set the VMEbus address jumpers on the VMEbus Async/Printer controller to the appropriate positions as shown in Figure 2–6, "VMEbus Async/Printer Controller Address Jumpers":

Table 2–5, Controller Address and Interrupt Levels

Controller Number	Address	Interrupt Level	Bus Request Level
0	0x1100	7	3
1	0x1120	Not assigned	3
2	0x1140	Not assigned	3
3	0x1160	Not assigned	3
4	0x1180	Not assigned	3
5	0x11A0	Not assigned	3

8. Install the VMEbus Async/Printer controller into the appropriate slot in the VMEbus chassis.
9. Route the cable through the appropriate cable opening on the rear of the VMEbus chassis, then connect the end of cable to the controller as shown in Figure 2–7, "VMEbus Chassis Cabling Openings":

Figure 2–6, VMEbus Async/Printer Controller Address Jumpers

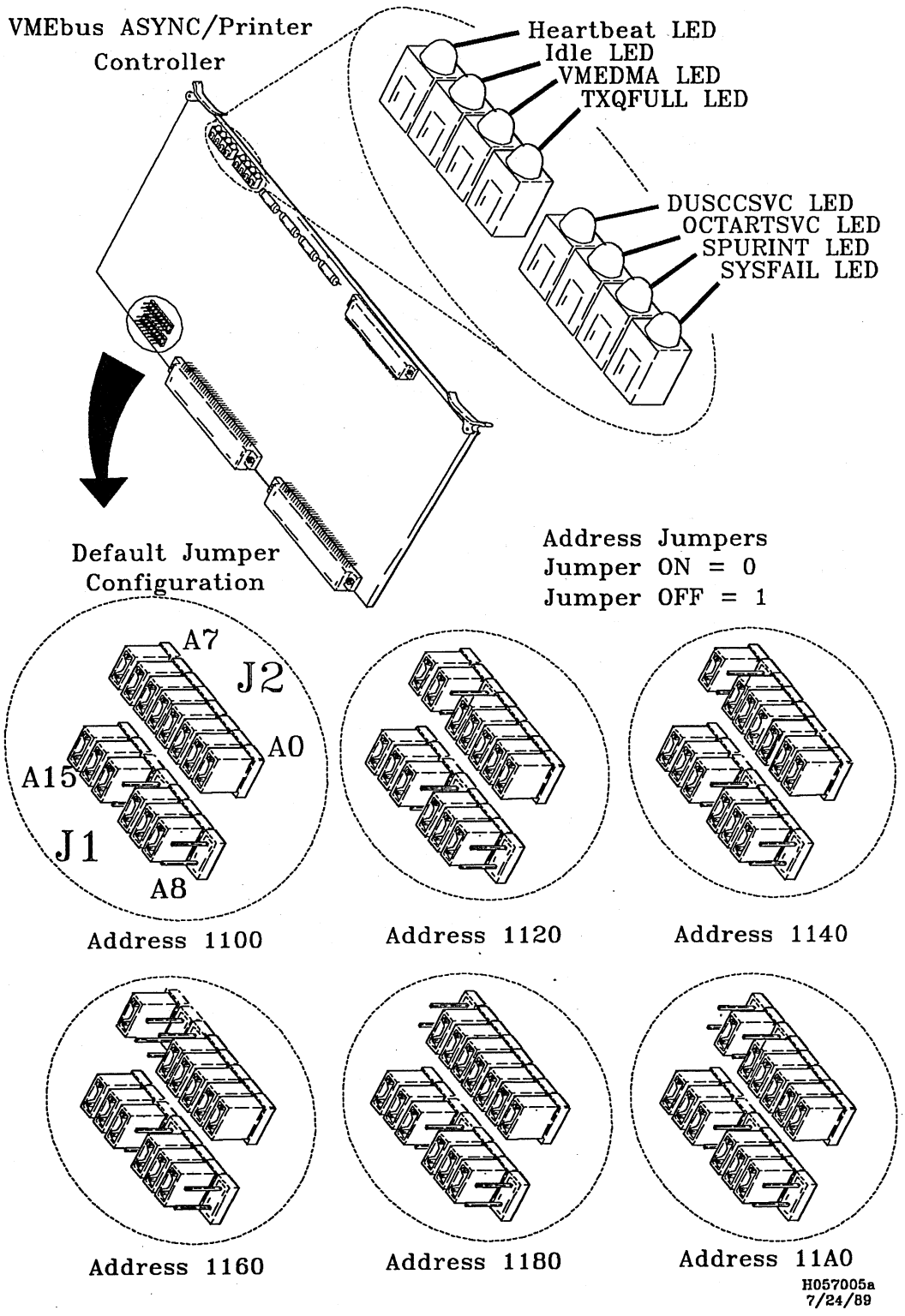
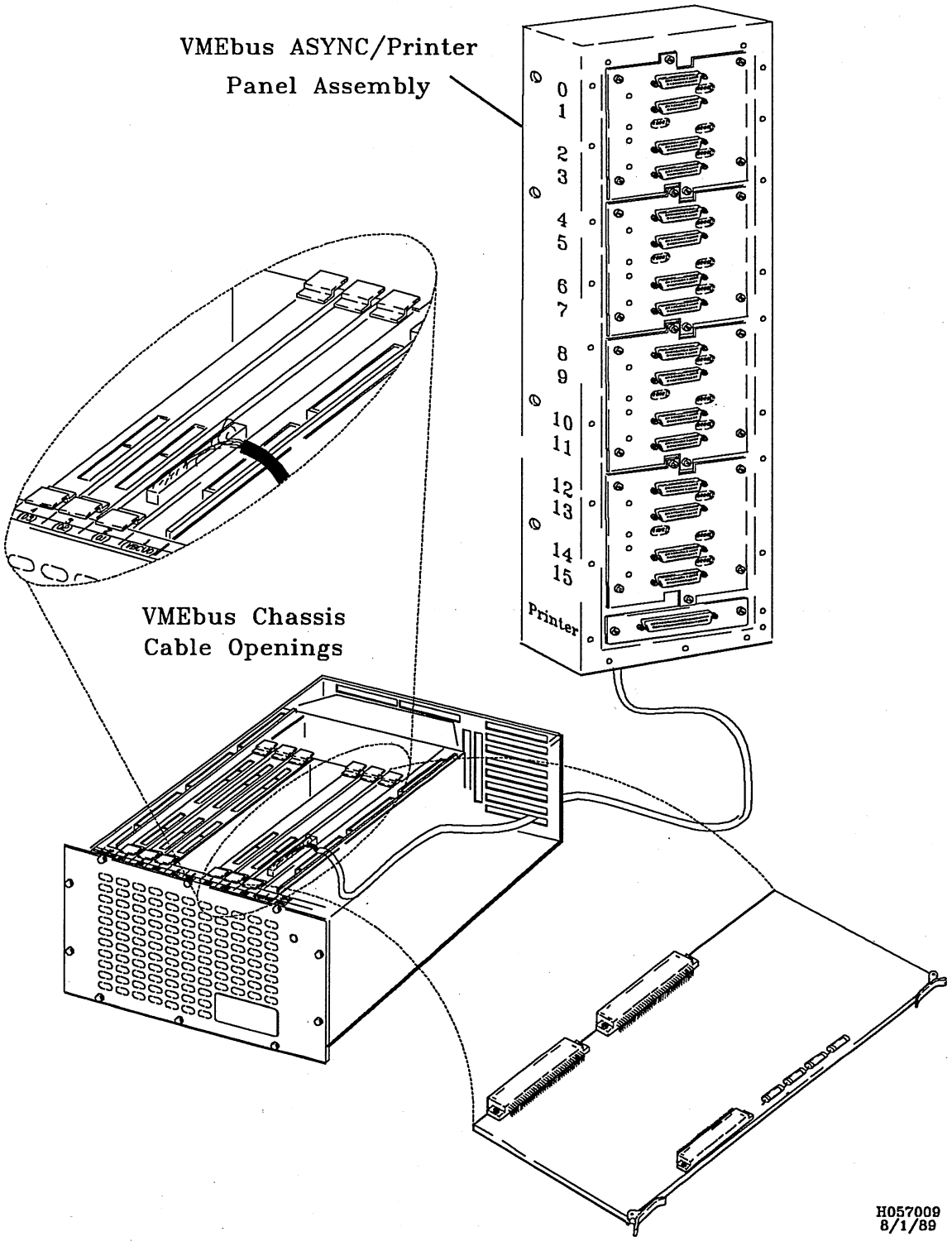


Figure 2–7, VMEbus Chassis Cabling Openings

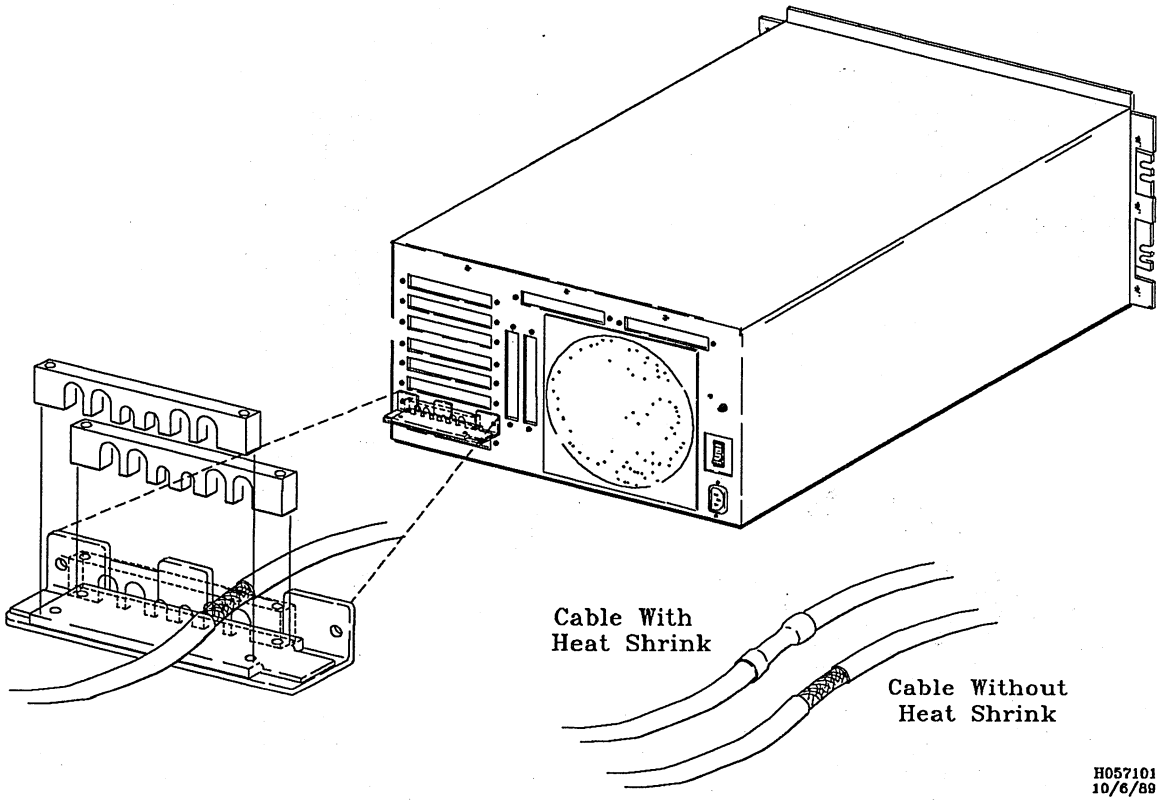


CAUTION

Failure to contact the exposed cable shield with the cable clamp will result in the loss of the EMI shielding.

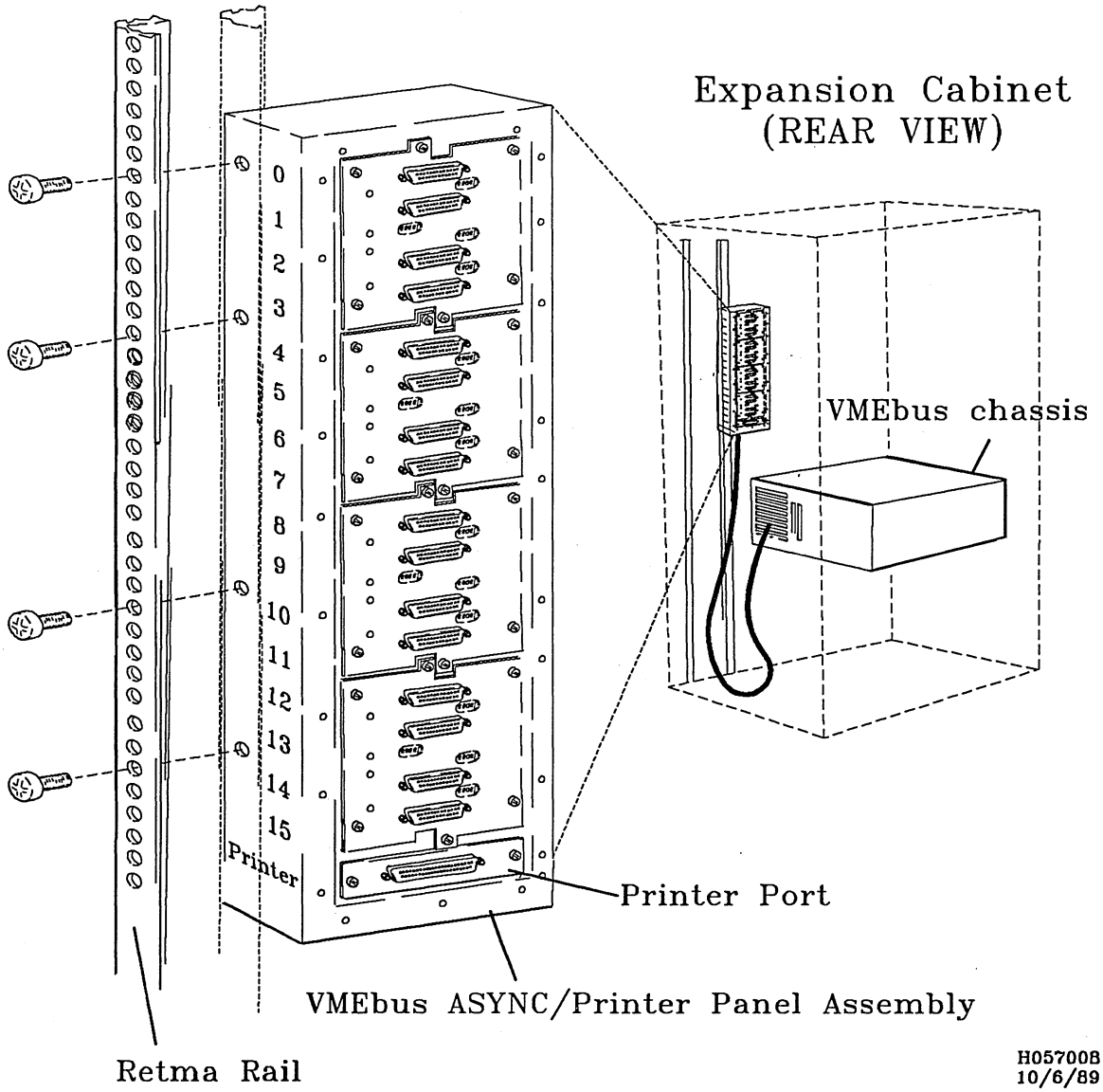
10. Install cable(s) (with exposed shields) in the inner cable clamp, then install the clamp on the mounting bracket. Install the outer clamp over the insulated cable area, then mount the complete assembly to the rear of the VMEbus chassis with the screws provided as shown in Figure 2–8, "Cable Clamps and Cable Shields":

Figure 2–8, Cable Clamps and Cable Shields



11. Mount the Async/Printer panel assembly to the Retma rails at the rear of the CONVEX expansion cabinet with the screws provided as shown in Figure 2–9, "Async/Printer Controller Panel Mounting":

Figure 2–9, Async/Printer Controller Panel Mounting



CAUTION

Do not operate the VMEbus chassis with its top panel removed. The panel must be installed to obtain proper airflow inside the VMEbus chassis.

12. Install the VMEbus chassis top panel and secure it with the 12 locking screws.
13. Return the VMEbus chassis to its retracted position and secure it with the 2 locking screws.
14. Return the expansion cabinet stabilizer bars to their retracted positions.
15. Set the processor's front control panel key switch to the **ON** position.
16. Set the VMEbus chassis power control switch to the **ON** position.

> **NOTE**
>
> Refer to Chapter 3 for diagnostic test information and procedures on the VMEbus Async/Printer controller.

Chapter 3
Integration and Test

3.1 Overview

The VMEbus Async/Printer controller must be integrated into the CONVEX Operating System (ConvexOS) before it can be used. Guidelines for integrating a VMEbus Async/Printer controller into ConvexOS are contained in this chapter as well as information on the VMEbus Async/Printer controller diagnostic test.

3.1.1 Software Integration

ConvexOS contains all of the software drivers for the hardware supported by CONVEX. This means that a system generation is not required when the VMEbus subsystem is installed on a CONVEX computer.

System-level hardware is identified to ConvexOS via a configuration file (*/ioconfig*) located on the Service Processor Unit (SPU) disk. The */ioconfig* file describes, in hierarchical fashion, the connections between VIOPs, controller(s), and peripheral devices. ConvexOS uses this information to assign a physical device number to a device of a given type.

Each type of VMEbus device is identified to ConvexOS by a mnemonic device code. These codes, and other information, are entered into the */ioconfig* file on the SPU disk. This file contains entries, such as VIOP number, VMEbus chassis number, controller type, control and status register (csr) address, interrupt number, and peripheral device type. A typical */ioconfig* file is shown in Figure 3–1, "Example */ioconfig* File":

Figure 3–1, Example */ioconfig* File

```
viop 5
 vme 0
  ctlr MTC-201 csr 0x3fc0 int 7
   unit 0 type MTD-202
  ctlr MTC-201 csr 0x1040 int 6
   unit 0 type MTD-202
  ctlr DKC-204 csr 0x800 int 3
   unit 0 type DKD-208
  ctlr LAN-007 csr 0xfd80 int 5
   unit 0 type ve
 vme 1
  ctlr DKC-204 csr 0x800 int 3
   unit 0 type DKD-206
  ctlr DKC-203 csr 0xa00 int 4
   unit 0 type DKD-214
  ctlr ACM-201 csr 0x1100 int 6
   unit 0 type TTY
   unit 1 type TTY
   unit 2 type TTY
   unit 3 type TTY
   unit 4 type TTY
   unit 5 type TTY
   unit 6 type TTY
   unit 7 type TTY
   unit 8 type TTY
   unit 9 type TTY
   unit 10 type TTY
   unit 11 type TTY
   unit 12 type TTY
   unit 13 type TTY
   unit 14 type TTY
   unit 15 type TTY
   unit 16 type PRT-CEN
                PRT-DAT
```

Whenever a VMEbus controller is added or removed, the information in the hardware section of the configuration file (*/ioconfig*) must be changed, otherwise system operation problems will occur. The *CONVEX System Manager's Guide* should be consulted when making these changes.

3.2 Testing the VMEbus Async/Printer Controller

The VMEbus Async/Printer controller is tested by the *dev5300* diagnostic program. This program verifies the operation of the VMEbus Async/Printer controller and cable adapter. The *dev5300* diagnostic program verifies that:

- VIOP can boot the VMEbus Async/Printer controller program from memory
- Communications between the VIOP, VBCU, and VMEbus Async/Printer controller are established
- VBCU responds to forced interrupts
- The VMEbus Async/Printer controller can correctly execute instructions
- VMEbus cable adapter operates in the loopback mode

The *dev5300* diagnostic program is an offline program that must be executed on the SPU while the CPU is halted. The procedures for executing this test are beyond the scope of this manual. However, this information is contained in the *CONVEX PBUS I/O System Diagnostics Manual.* This manual should be consulted before running this test.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 4
Maintenance Procedures and IPB

4.1 Overview

Guidelines for obtaining technical assistance, and maintenance procedures for the VMEbus Async/Printer controller are contained in this chapter. Also, an Illustrated Parts Breakdown (IPB) for all Field Replaceable Units (FRUs) on the VMEbus Async/Printer controller is included.

4.2 CONVEX Technical Assistance

CONVEX offers two sources of help if problems arise:

- CONVEX Technical Assistance Center (TAC)
- CONVEX *contact* utility

4.2.1 CONVEX Technical Assistance Center

Contact the CONVEX Technical Assistance Center (TAC) for real time support on urgent hardware and software problems. The TAC can be reached from all locations in the continental United States by calling 1(800)952-0379, or by calling 1(214)952-4379 from other locations in Alaska, Hawaii, or Canada. From all other locations, contact the nearest CONVEX office.

4.2.2 CONVEX *contact* Utility

Use the CONVEX *contact* utility for reporting minor hardware and software problems. Refer to Appendix B for an example of the CONVEX *contact* utility.

4.3 Maintenance Procedures

The following sections define the maintenance procedures for the VMEbus Async/Printer controller and panel assembly.

4.3.1 VMEbus Async/Printer Controller

The following sections define the procedures to remove and replace the VMEbus Async/Printer controller.

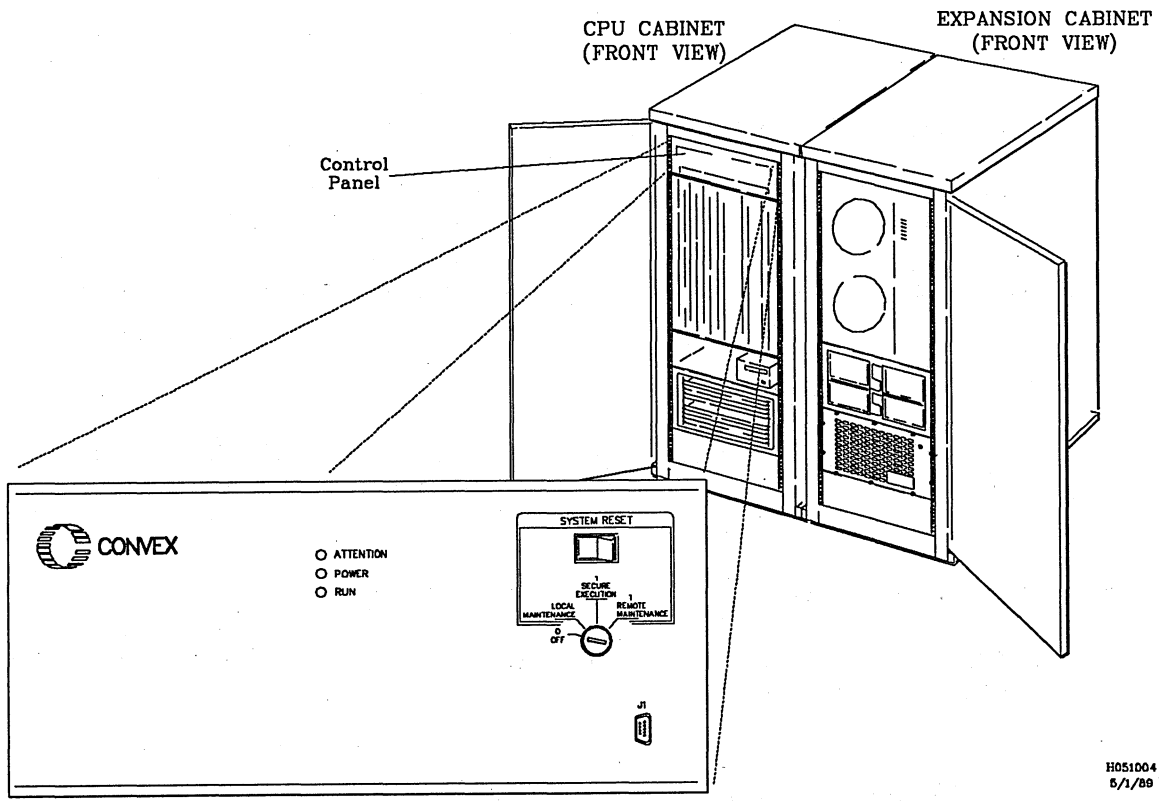
4.3.1.1 Removal

Follow these procedures to remove the VMEbus Async/Printer controller.

> **CAUTION**
>
> Failure to shut the system down before removing power to the VMEbus chassis will cause a system crash. Refer to the *CONVEX Processor Operation Guide (C100 Series, C200 Series)* for power down procedures on a CONVEX computer.

1. Turn the processor's front control panel key switch to the **OFF** position as shown in Figure 4–1, "Typical Front Panel Power Control Switch":

Figure 4–1, Typical Front Panel Power Control Switch

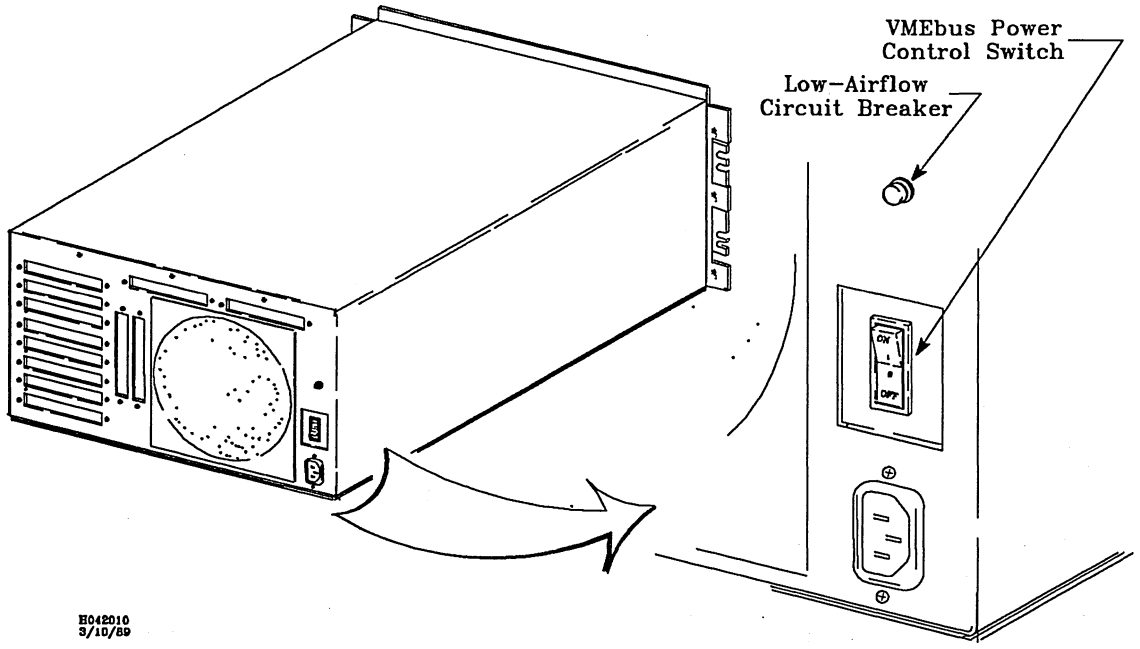


CAUTION

Failure to remove power to the VMEbus chassis before installing or removing equipment will damage electronic components.

2. Set the VMEbus chassis power control switch to the **OFF** position as shown in Figure 4–2, "VMEbus Chassis Power Control Switch":

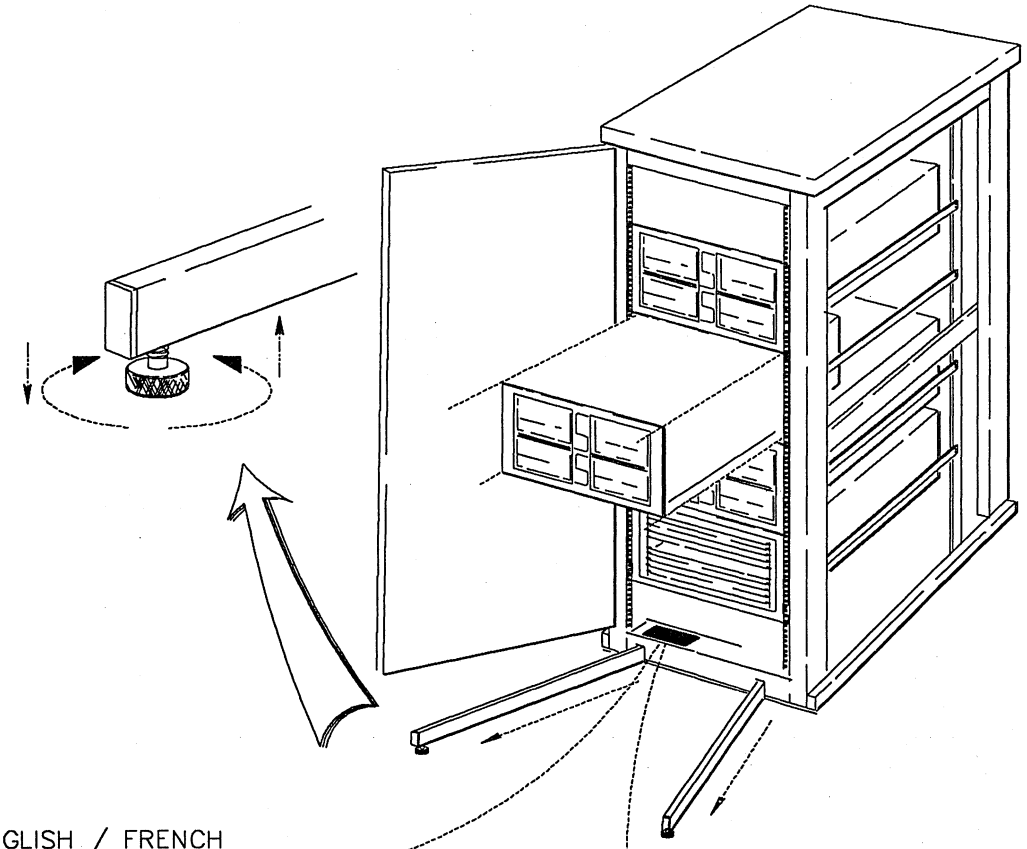
Figure 4–2, VMEbus Chassis Power Control Switch



WARNING

Expansion cabinet stabilizer bars must be extended prior to installing a VMEbus chassis, or before extending the VMEbus chassis assembly from its expansion cabinet for service. Failure to do so will make the expansion cabinet unstable, increasing the possibility of it falling forward. This can cause injury to personnel and will cause damage to equipment.

Figure 4–3, Expansion Cabinet Stabilizer Bars



ENGLISH / FRENCH

CAUTION	ATTENTION
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	POUR REDUIRE LE RISQUE D'ACCIDENT ATTRIBUABLE A L'INSTABILITE DE L'UNITE, DEPLOYER LES STABILISATEURS AVANT DE SORTIR LES PERIPHERIQUES. 1. POUR DEPLOYER LES STABILISATEURS, TIRER COMPLETEMENT LES BRAS ANTI-BASCULEMENT ET ABAISER LES PATTES DE FACON QU'ELLES REPOSENT SOLIDEMENT SUR LE SOL. 2. S'ASSURER QUE TOUS LES PERIPHERIQUES SON MUNIS DE VIS DE BLOCAGE. 3. NE JAMAIS SORTIR PLUS D'UN PERIPHERIQUE A UN MOMMENT DONNE.

ENGLISH / GERMAN

CAUTION	ACHTUNG
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	ZUR VERMEIDUNG VON GEFAHRDUNG DURCH EIN INSTABILES GERAT SIND VOR DER HERAUSNAHME VON PERIPHERALS DER STABILISIERUNGSMECHANISMUS BETATIGT WERDEN. 1. UM DIE STABILISIERUNGSEINRICHTUNG ZU BETATIGEN, SIND DER "ANTITILT KANAL" GANZ HERAUS ZU ZIEHEN UND DER UNTERE STUTZFUSS AUF DEN BODEN ZU FUHREN. 2. OBERPRUFEN SIE, OB IN ALLEN ANDEREN VERSCHIEBBAREN GERATEN DER SICHERUNGSMECHANISMUS BETATIGT IST. 3. ZIEHEN SIE NIE MEHR ALS EIN GERAT HERAUS.

H015020
5/1/89

3. Extend the expansion cabinet stabilizer bars, and adjust feet until they are in firm contact with the floor as shown in Figure 4–3, "Expansion Cabinet Stabilizer Bars":
4. Unlock the 2 VMEbus chassis lock screws and extend the chassis on its slides.
5. Unlock the 12 top panel lock screws on the VMEbus chassis top panel and remove the top panel.
6. Disconnect the cable attached to the VMEbus Async/Printer controller.

> **CAUTION**
>
> The VMEbus Async/Printer controller can be damaged by Electrostatic Discharge (ESD). A grounded wrist strap (or other grounding method) must be used when handling the VMEbus Async/Printer controller.

7. Remove the VMEbus Async/Printer controller from its VMEbus chassis slot.

4.3.1.2 Replacement

> **Note**
>
> If a VMEbus Async/Printer Controller is to be initially installed, refer to the section "Installation Procedures" in Chapter 2 "Unpacking and Installation" for installation information.

Follow these procedures to replace the VMEbus Async/Printer controller.

> **CAUTION**
>
> The VMEbus Async/Printer controller can be damaged by Electrostatic Discharge (ESD). A grounded wrist strap (or other grounding method) must be used when handling the VMEbus Async/Printer controller.

1. Install the VMEbus Async/Printer controller into its VMEbus chassis slot.
2. Install the second VMEbus Async/Printer controller or MBCU into its proper position if applicable.
3. Connect the cable to the VMEbus Async/Printer controller.

> **CAUTION**
>
> Do not operate the VMEbus chassis with its top panel removed. The panel must be installed to obtain proper airflow inside the VMEbus chassis.

4. Install the VMEbus chassis top panel and secure it with the 12 locking screws.
5. Return the VMEbus chassis to its retracted position and secure it with the 2 locking screws.
6. Return the expansion cabinet stabilizer bars to their retracted positions.
7. Set the processor's front control panel key switch to the **ON** position (see Figure 4–2, Typical Front Panel Power Control Switch).
8. Set the VMEbus chassis power control switch to the **ON** position (see Figure 4–3, VMEbus Chassis Power Control Switch).

4.3.2 VMEbus Async/Printer Panel Assembly

The following sections define the procedures to remove and replace the VMEbus Async/Printer panel assembly.

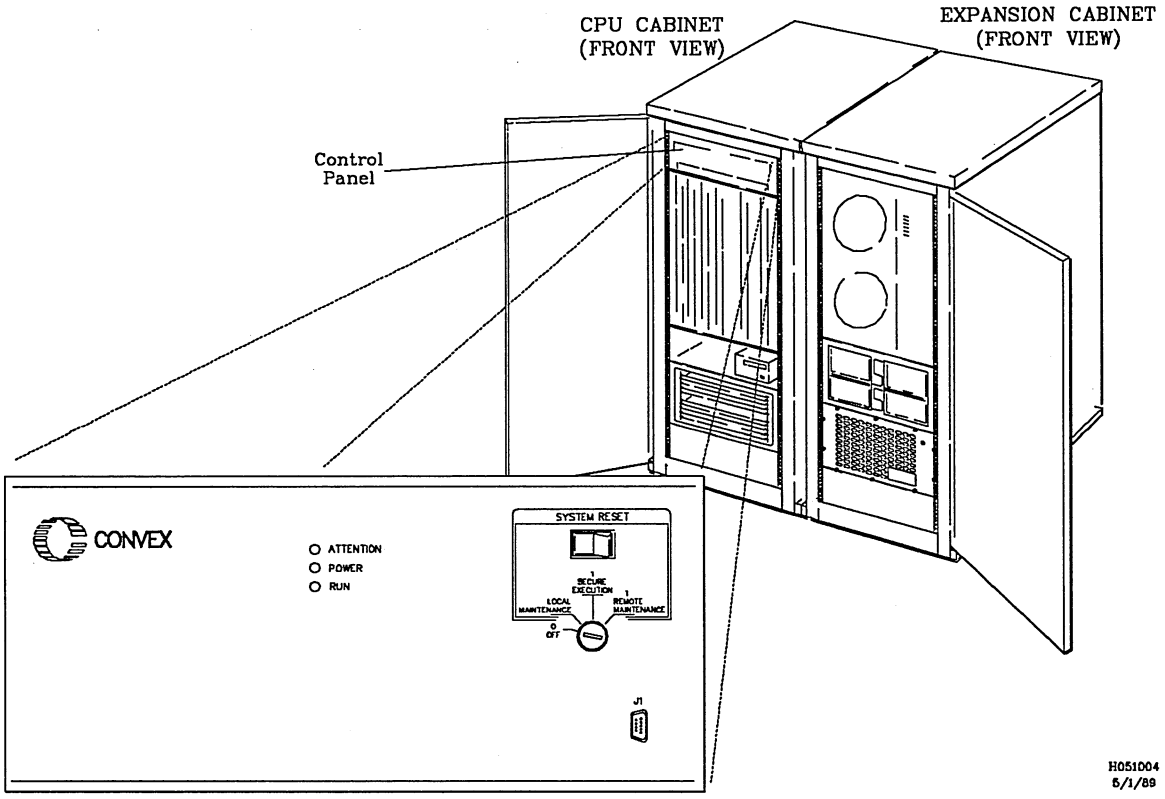
4.3.2.1 Removal

Follow these procedures to remove the VMEbus Async/Printer panel assembly.

> **CAUTION**
>
> Failure to shut the system down before removing power to the VMEbus chassis will cause a system crash. Refer to the *CONVEX Processor Operation Guide (C100 Series, C200 Series)* for power down procedures on a CONVEX computer.

1. Turn the processor's front control panel key switch to the **OFF** position as shown in Figure 4–4, "Typical Front Panel Power Control Switch":

Figure 4–4, Typical Front Panel Power Control Switch

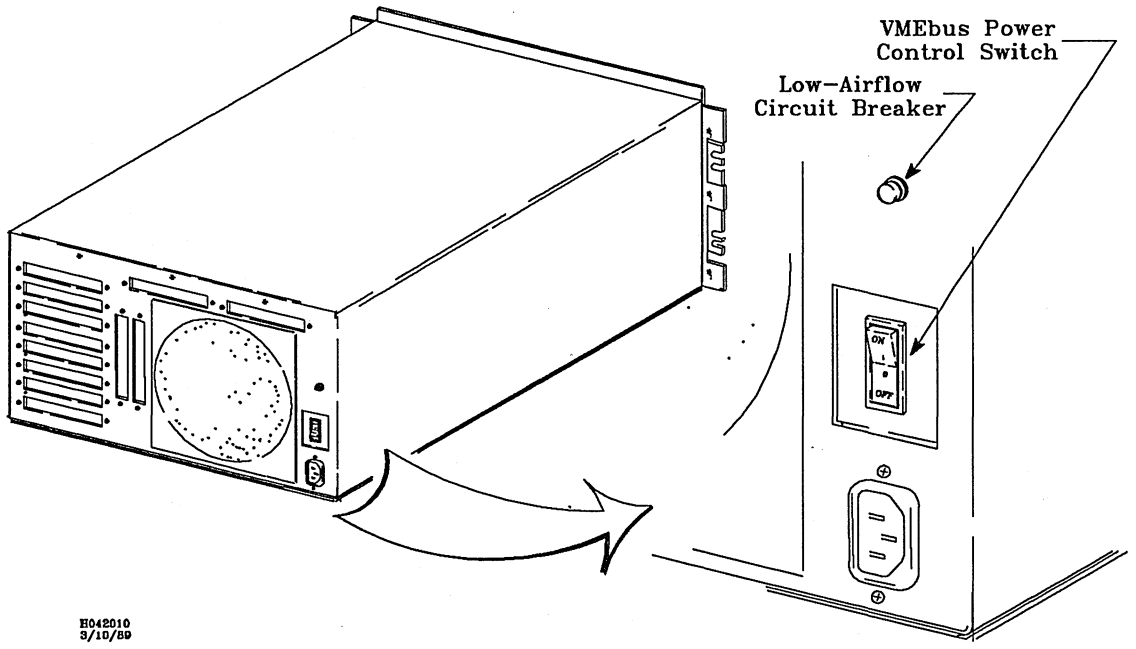


CAUTION

Failure to remove power to the VMEbus chassis before installing or removing equipment will damage electronic components.

2. Set the VMEbus chassis power control switch to the **OFF** position as shown in Figure 4–5, "VMEbus Chassis Power Control Switch":

Figure 4–5, VMEbus Chassis Power Control Switch

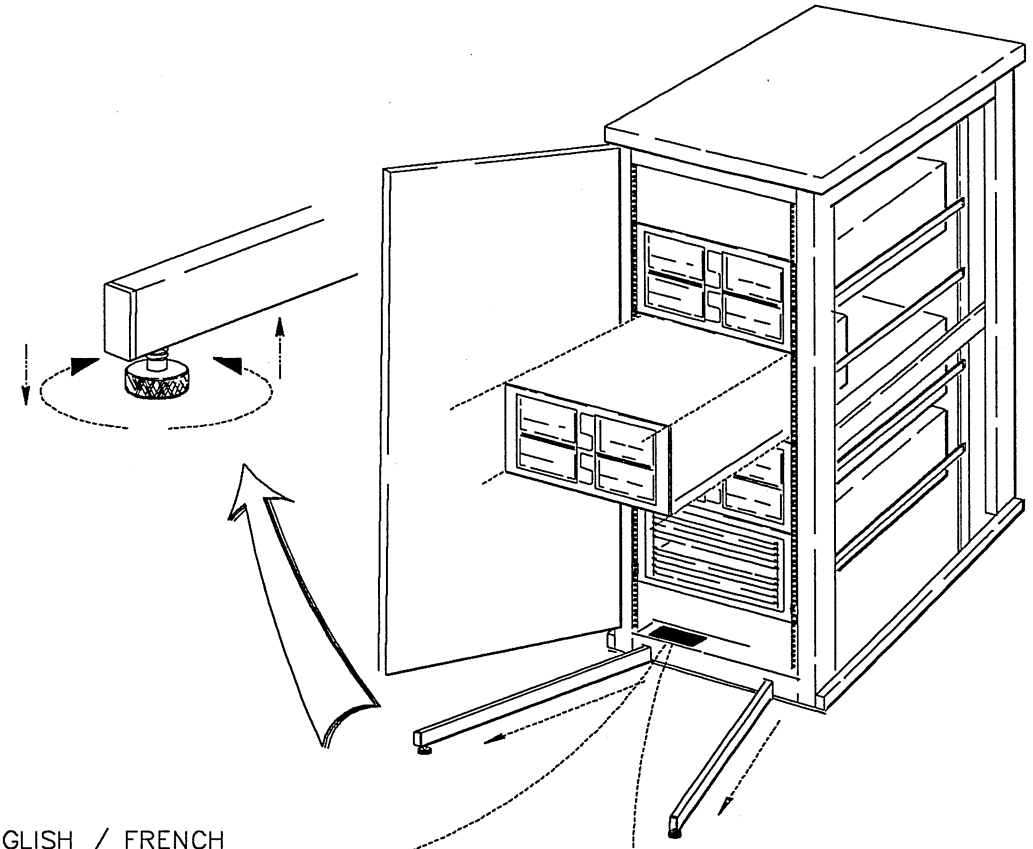


WARNING

Expansion cabinet stabilizer bars must be extended prior to installing a VMEbus chassis, or before extending the VMEbus chassis assembly from its expansion cabinet for service. Failure to do so will make the expansion cabinet unstable, increasing the possibility of it falling forward. This can cause injury to personnel and will cause damage to equipment.

3. Extend the expansion cabinet stabilizer bars, and adjust feet until they are in firm contact with the floor as shown in figure 4–6, "Expansion Cabinet Stabilizer Bars":

Figure 4–6, Expansion Cabinet Stabilizer Bars



ENGLISH / FRENCH

CAUTION	ATTENTION
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	POUR REDUIRE LE RISQUE D'ACCIDENT ATTRIBUABLE A L'INSTABILITE DE L'UNITE, DEPLOYER LES STABILISATEURS AVANT DE SORTIR LES PERIPHERIQUES. 1. POUR DEPLOYER LES STABILISATEURS, TIRER COMPLETEMENT LES BRAS ANTI-BASCULEMENT ET ABAISER LES PATTES DE FACON QU'ELLES REPOSENT SOLIDEMENT SUR LE SOL. 2. S'ASSURER QUE TOUS LES PERIPHERIQUES SON MUNIS DE VIS DE BLOCAGE. 3. NE JAMAIS SORTIR PLUS D'UN PERIPHERIQUE A UN MOMMENT DONNE.

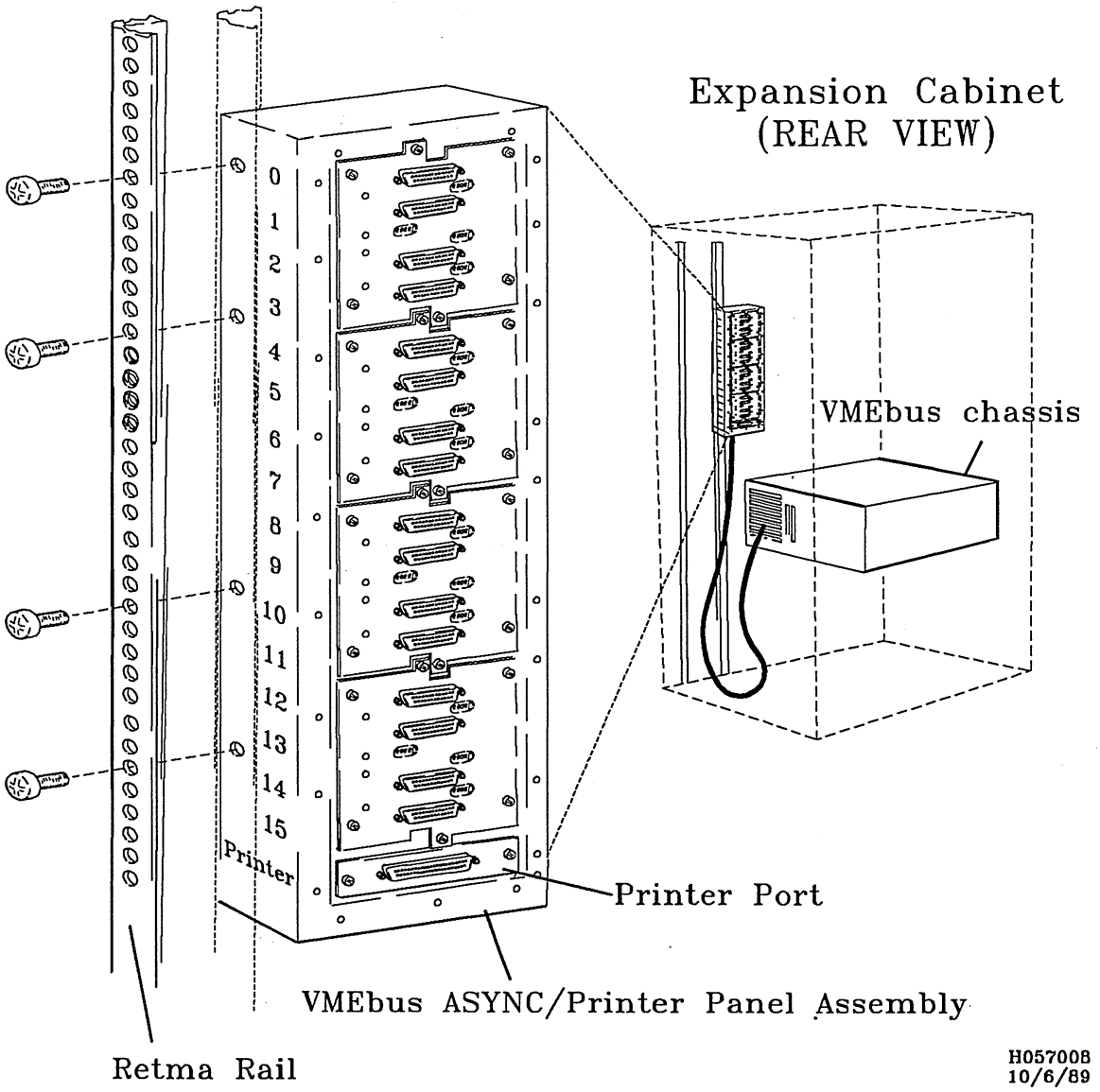
ENGLISH / GERMAN

CAUTION	ACHTUNG
TO REDUCE RISK OF POSSIBLE INJURY DUE TO UNSTABLE UNIT, ACTUATE STABILIZER BEFORE ANY PERIPHERAL IS EXTENDED. 1. TO ACTUATE STABILIZER, FULLY EXTEND ANTITILT CHANNELS AND LOWER CHANNEL SUPPORT FEET FIRMLY TO THE FLOOR. 2. INSURE THAT LOCKING MECHANISMS ARE INSTALLED IN ALL OTHER EXTENDABLE UNITS. 3. NEVER EXTEND MORE THAN ONE UNIT AT A TIME	ZUR VERMEIDUNG VON GEFAHRDUNG DURCH EIN INSTABILES GERAT SIND VOR DER HERAUSNAHME VON PERIPHERALS DER STABILISIERUNGSMECHANISMUS BETATIGT WERDEN. 1. UM DIE STABILISIERUNGSEINRICHTUNG ZU BETATIGEN, SIND DER "ANTITILT KANAL" GANZ HERAUS ZU ZIEHEN UND DER UNTERE STUTZFUSS AUF DEN BODEN ZU FUHREN. 2. OBERPRUFEN SIE, OB IN ALLEN ANDEREN VERSCHIEBBAREN GERATEN DER SICHERUNGSMECHANISMUS BETATIGT IST. 3. ZIEHEN SE NE MEHR ALS EIN GERAT HERAUS.

H015020
5/1/89

4. Unlock the 2 VMEbus chassis lock screws and extend the chassis on its slides.
5. Unlock the 12 top panel lock screws on the VMEbus chassis top panel and remove the top panel.
6. Disconnect the cable attached to the VMEbus Async/Printer controller.
7. Unlock the locking screws on the mounting bracket and remove the complete assembly from the rear of the VMEbus chassis. Then unlock the locking screws on the inner cable clamp and remove the inner cable clamp from the cable (with exposed shields). Then unlock the locking screws on the outer cable clamp and remove the outer clamp from the insulated cable area.
8. Remove the panel assembly cable from the VMEbus chassis.

Figure 4–7, Async/Printer Panel Assembly Mounting



9. Unlock the locking screws for the panel assembly and remove the panel assembly from the Retma rails at the rear of the CONVEX expansion cabinet as shown in Figure 4–7, "Async/Printer Controller Panel Mounting":

4.3.2.2 Replacement

> **Note**
>
> If a VMEbus Async/Printer panel assembly is to be initially installed, refer to the section "Installation Procedures" in Chapter 2 "Unpacking and Installation" for installation information.

Follow these procedures to replace the VMEbus Async/Printer panel assembly.

1. Route the panel assembly cable through the appropriate cable opening on the rear of the VMEbus chassis, then connect the end of cable to the controller as shown in Figure 4–8, "VMEbus Chassis Cabling Openings":

> **CAUTION**
>
> Failure to contact the exposed cable shield with the cable clamp will result in the loss of the EMI shielding.

2. Install cable (with exposed shields) in the inner cable clamp, then install the clamp on the mounting bracket. Install the outer clamp over the insulated cable area, then mount the complete assembly to the rear of the VMEbus chassis with the screws provided as shown in Figure 4–9, "Cable Clamps and Cable Shields":

Figure 4–8, VMEbus Chassis Cabling Openings

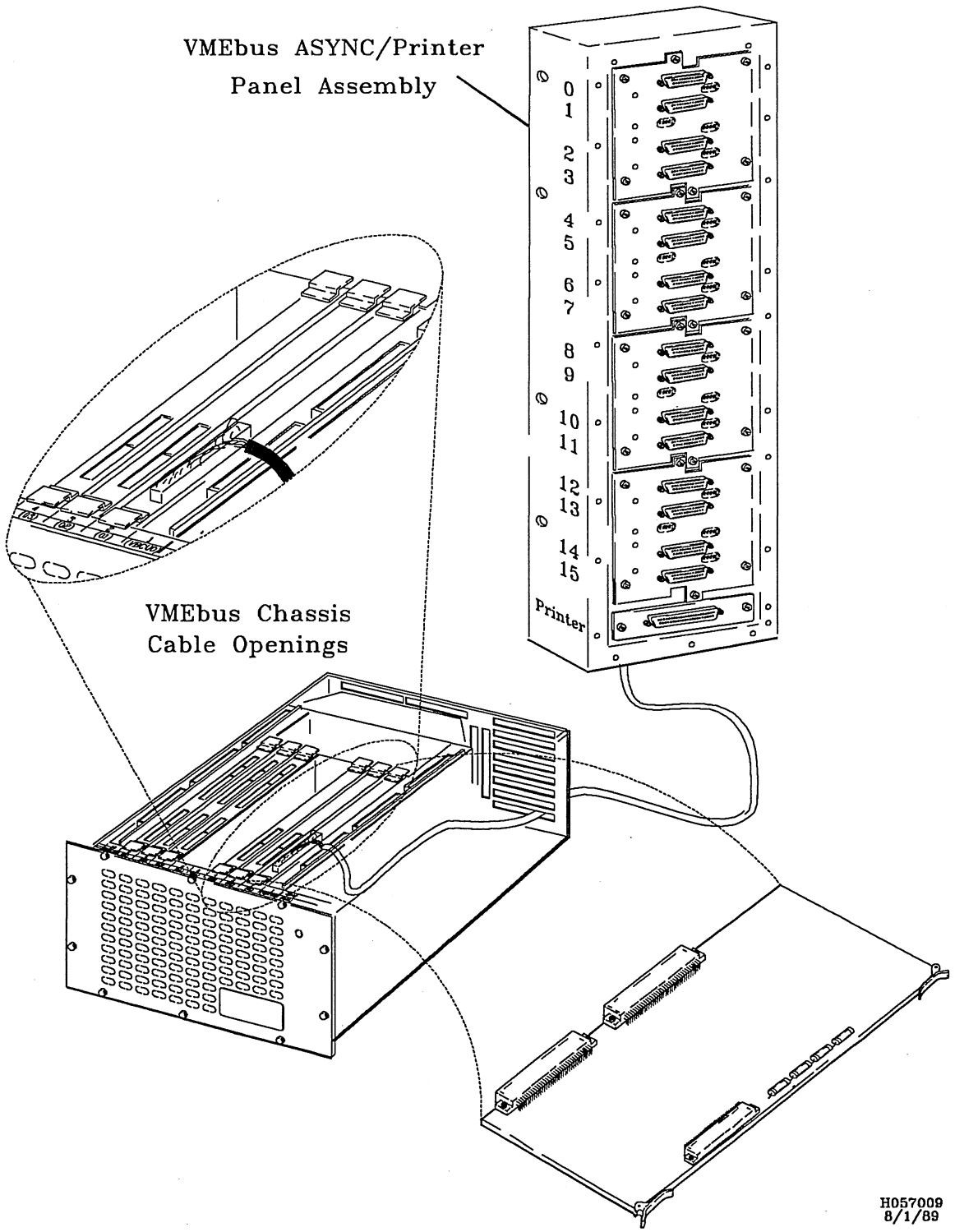
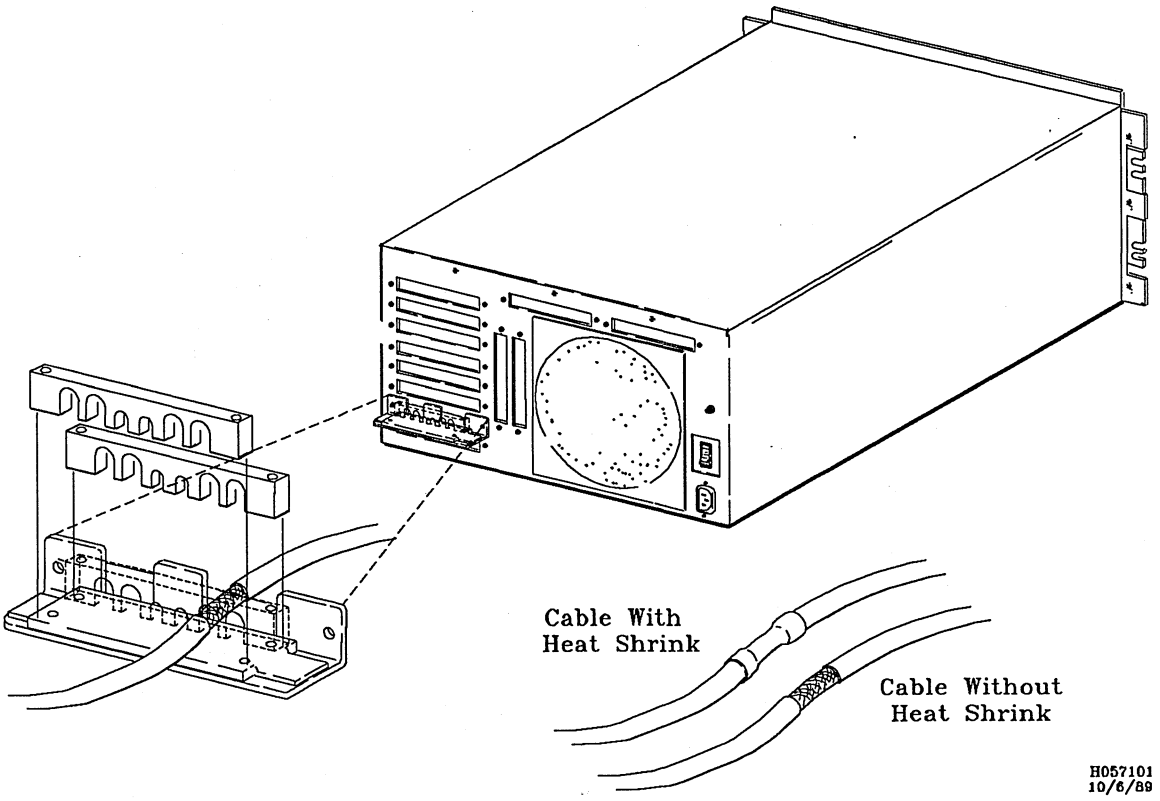
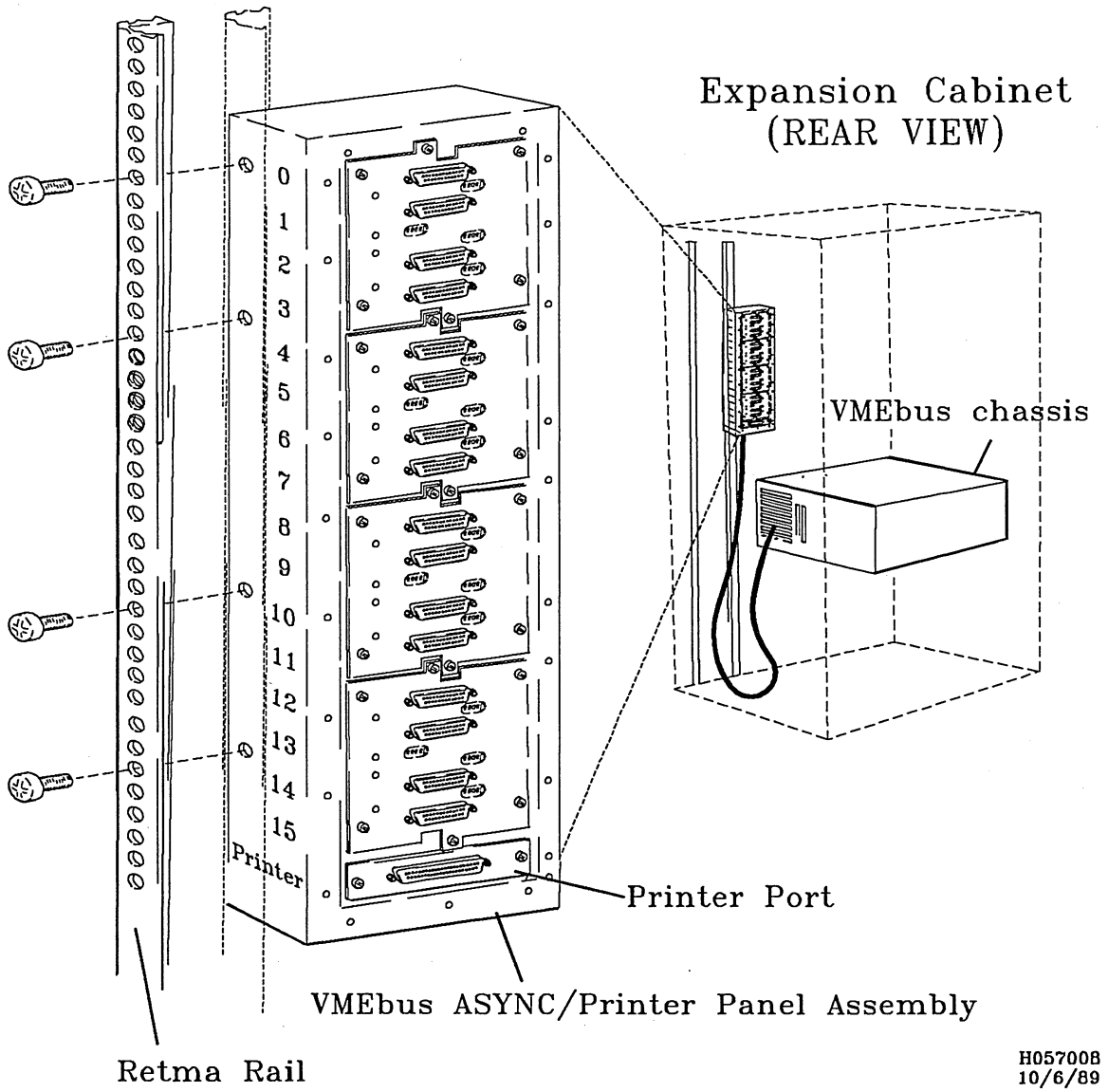


Figure 4–9, Cable Clamps and Cable Shields



3. Mount the Async/Printer panel assembly to the Retma rails at the rear of the CONVEX expansion cabinet with the screws provided as shown in Figure 4–10, "Async/Printer Controller Panel Mounting":

Figure 4–10, Async/Printer Panel Assembly Mounting



CAUTION

Do not operate the VMEbus chassis with its top panel removed. The panel must be installed to obtain proper airflow inside the VMEbus chassis.

4. Install the VMEbus chassis top panel and secure it with the 12 locking screws.
5. Return the VMEbus chassis to its retracted position and secure it with the 2 locking screws.
6. Return the expansion cabinet stabilizer bars to their retracted positions.
7. Set the processor's front control panel key switch to the **ON** position.
8. Set the VMEbus chassis power control switch to the **ON** position.

> **NOTE**
>
> Refer to Chapter 3 for diagnostic test information and procedures on the VMEbus Async/Printer controller.

4.4 Illustrated Parts List

This section is the Illustrated Parts Breakdown (IPB) for the VMEbus Async/Printer controller. Table 4–1, "VMEbus Async/Printer controller Parts List," lists the CONVEX part numbers for all Field Replaceable Units (FRUs). The table also contains a figure number reference for each FRU:

Table 4–1, VMEbus Async/Printer Controller Parts List

Description	Part Number	Quantity	Figure No.
Async/Printer controller assembly[1]	550-000117-210	1	4-8
Async/Printer controller	410-001193-200	1	4-8
Async/Printer panel assembly	500-000276-200	1	4-8

[1] An assembly contains a complete set of CONVEX Async/Printer hardware.

Appendix A
Async/Printer Configurator Document

This appendix contains a copy of the VMEbus Async/Printer Controller Configurator document.

> **NOTE**
>
> The *VMEbus Async/Printer Controller Configurator* document contains basic configuration information for the VMEbus Async/Printer controller and related hardware. In the event of changes regarding VMEbus Async/Printer controller configuration, an updated version of the document will be made available. Configurator document updates should be inserted into this appendix.

VME Async/Printer Controller Configurator

1.1 Scope

The purpose of this document is to provide configuration information for the VASYNC Controller (Convex P/N 410-001193-200).

1.2 Controller Configuration

The VASYNC Controller has two jumper blocks which are used to set the board's base address. The jumper blocks are defined below.

Jumper block J2 - Lower left side of board								
Address	A7	A6	A5	A4	A3	A2	A1	A0
Jumper	1	2	3	4	5	6	7	8

Jumper block J1 - Lower left side of board								
Address	A15	A14	A13	A12	A11	A10	A9	A8
Jumper	1	2	3	4	5	6	7	8

Assigned address and interrupt levels for the VASYNC Controller are as follows. See the figure on page 3 of this document. Interrupt levels for the second through sixth controllers are not assigned. They should be set to a unique level upon installation in the target VMEbus.

Standard Address and Interrupt levels			
Controller #	**Address**	**Interrupt Level**	**Bus Request Level**
1	0x1100	7	3
2	0x1120	None Assigned	3
3	0x1140	None Assigned	3
4	0x1160	None Assigned	3
5	0x1180	None Assigned	3
6	0x11A0	None Assigned	3

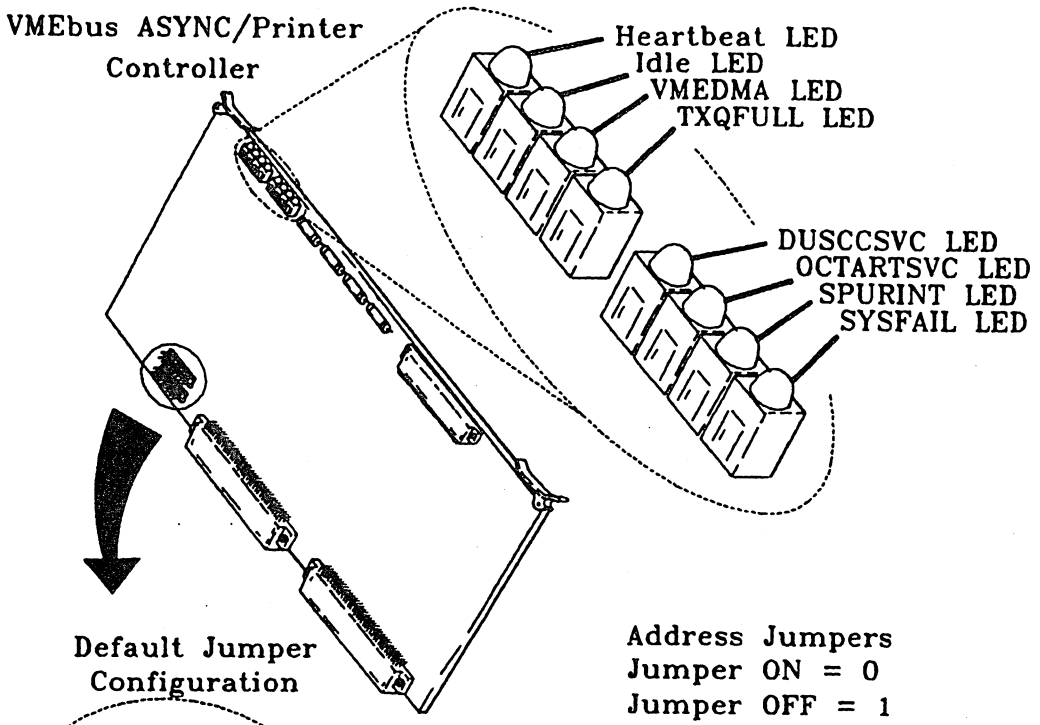
NOTE: Configuration jumpers are Convex P/N 312-000143-001.

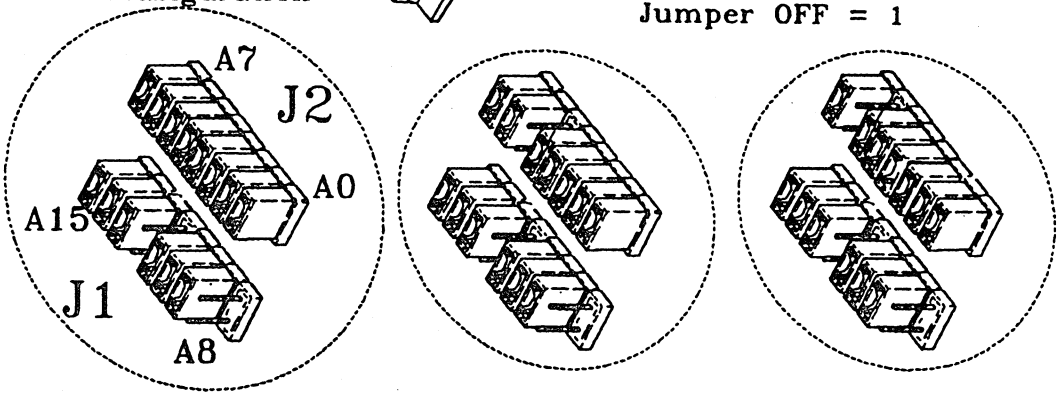
1.3 Panel Configuration

Each port on the VASYNC Panel has a block of four dip switches which control sourcing of the transmit and receive clocks. See the figure on page 4 of this document. The settings are defined below.

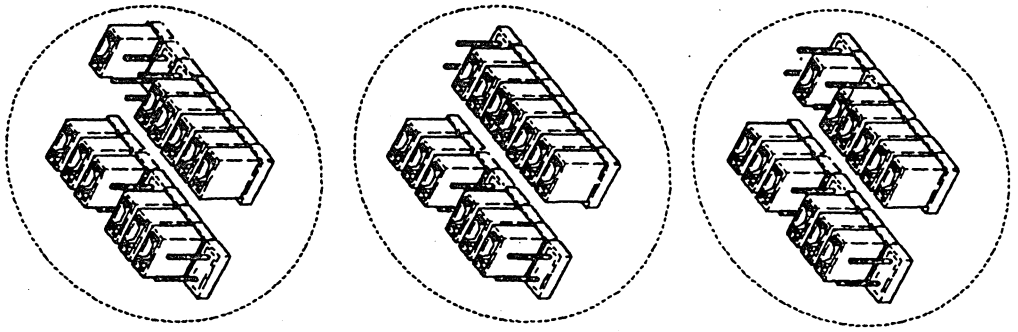
Panel dip switch block				
Mode of Operation	Switch number			
	1	2	3	4
Asynchronous	Off	Off	Off	Off

NOTE: The VME chassis power supply must be turned off before any of these switch settings are changed. Otherwise, damage to the Octal UART and DUSCCs will occur.





Address 1100 | Address 1120 | Address 1140

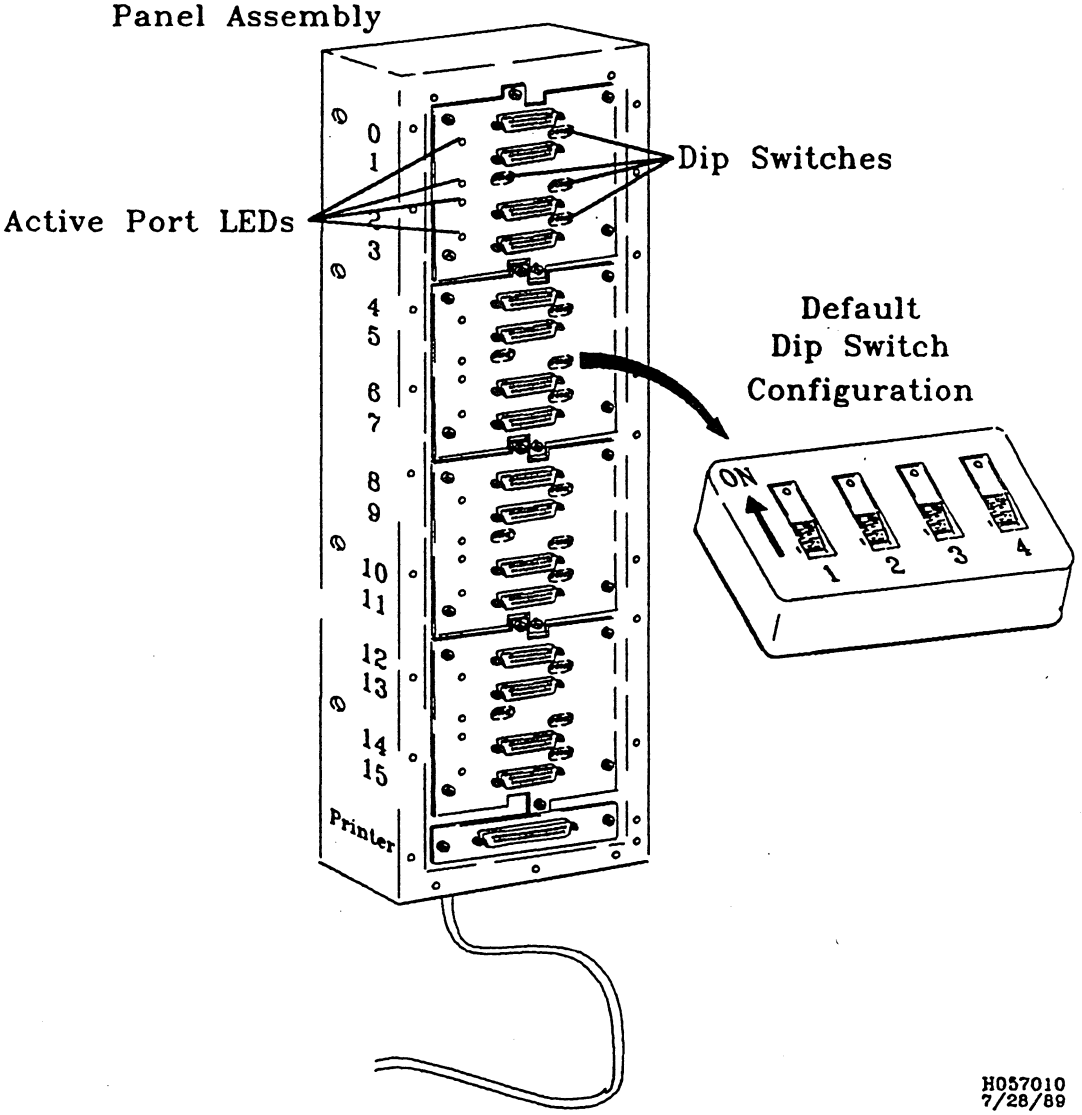


Address 1160 | Address 1180 | Address 11A0

H057005a
7/24/89

CONVEX	TITLE: VASYNC Configurator	REV: B.0 7/28/89
		ENGR: Ken King
	Part Number: 410-001193-600A	PAGE: 3

VMEbus ASYNC/Printer
Panel Assembly



H057010
7/28/89

CONVEX

TITLE: VASYNC Configurator

Part Number: 410-001193-600A

REV: B.0 7/28/89

ENGR: Ken King

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix B
Reporting Problems

B.1 Overview

This appendix introduces the CONVEX Technical Assistance Center (TAC) and the *contact* utility. The *contact* utility is an online system for reporting problems to the TAC. To learn *contact* by using it, enter **contact** at the system prompt and then answer the questions as they appear on the screen. To find out more about using *contact*, read through this appendix. It describes prerequisites and tips for using *contact* and the step-by-step process *contact* takes you through.

B.2 Technical Assistance Center

The CONVEX Technical Assistance Center (TAC) is staffed by technical specialists who can address the diverse questions and problems that arise in a supercomputing environment. If you have a hardware, software, or documentation problem, contact the TAC. This group stands ready to solve such problems.

B.3 The *contact* Utility

The TAC recommends using the *contact* utility to report a hardware, software, or documentation problem. The *contact* utility is an interactive utility that helps the TAC track reports and route them to the the CONVEX personnel most qualified to fix them.

After invoking *contact*, it prompts for information about the problem. When you finish your report, *contact* electronically mails it to the TAC. You are notified within 48 hours that the TAC has received your report.

B.4 Prerequisites

To use *contact* requires

- a UNIX-to-UNIX Communication Protocol (UUCP) connection to the TAC
- the full path name of the program or utility in question
- the version number of the program or utility in question

B.4.1 UUCP Connection

Before using *contact*, check with your system administrator to be sure there is a UUCP connection to the TAC. A UUCP connection allows files to be copied from one UNIX system to another. The *uucp* (UNIX-to-UNIX copy) command relies on either a dial-up or hard-wired UUCP communication line.

B.4.2 Finding the Program Path Name

To determine the full path name of the program or utility in question, use the *which* command. The following screen illustrates using the *which* command to find the full path name of the loader (*ld*) utility:

```
>which ld
/bin/ld
>
```

In this example, the full path name of the loader is */bin/ld*.

For more information on the *which* command, refer to the *which*(1) man page. You can also use the *info* online information system. Enter **info which** at the system prompt. If you use the C shell (*csh*), you can also use the *whence* command to find the program path name. The *whence* command works like *which*, only faster.

B.4.3 Finding the Program Version Number

To determine the version number of the program or utility in question, use the *vers* command. The following screen illustrates using the *vers* command (enter **vers**, then the path name of the program or utility) to find the version number of the loader (*ld*) utility.

```
>vers /bin/ld
/bin/ld: 7.0
>
```

In this example, the loader utility version number is 7.0.

For more information on the *vers* command, refer to the *vers*(1) man page. You can also use the *info* online information system. To do so, enter **info vers** at the system prompt.

B.5 Tips on Using the *contact* Utility

The *contact* utility is interactive and easy to use. This section lists tips to help use it efficiently. In particular, this section tells how to

- use a *.contact* file
- abort a contact session
- resubmit an aborted report
- suspend a contact session
- move from one prompt to another
- use tilde-escape sequences in the *contact* utility

B.5.1 Using a *.contact* File

When invoked, *contact* prompts for information regarding the problem. The first prompt is for your name, title, phone number, and company name. You can, however, create a *.contact* file to skip this first prompt. Follow these steps:

1. Create a *.contact* file in your home directory.
2. Enter your name, job title, phone number, and company name, each on a new line.

When you invoke *contact*, it automatically includes the *.contact* file as input for the first prompt and proceeds to the next prompt.

B.5.2 Aborting the Report

To abort a contact report, either enter the interrupt key (usually CTRL-C) or choose the abort option when prompted by the *contact* utility. Using CTRL-C to abort does not save the contents of the report. Using the abort option saves the contents of the report in a file named *dead.report* in your home directory.

B.5.3 Submitting the *dead.report* File

When aborting a contact session, the *contact* utility saves the report in a file named *dead.report* in your home directory. Using the *contact* command with the *-r* option automatically merges the contents of the *dead.report* file into the new contact session. Enter

contact -r

and *contact* finds the *dead.report* file in your home directory and merges it into the contact report. You can then edit the report. When you end the editing session, *contact* returns to the final prompt, which asks you to review, edit, submit, or abort the report.

B.5.4 Suspending a Report

Sometimes it is necessary to stop in the middle of a contact report and return to the shell (for instance, to suspend the contact session to find the program path name or version number). To suspend the contact session, press CTRL-Z. To return to the contact session, enter **fg**. Using CTRL-Z and the *fg* (foreground) command lets you switch back and forth between the *contact* utility and the shell. You cannot, however, use CTRL-Z and *fg* to switch back and forth if you are using a Bourne shell (*sh*).

B.5.5 Ending a Response

The *contact* utility prompts for information pertinent to your hardware, software, or documentation question. Some prompts require one-line responses; to move to the next prompt, press RETURN. Other prompts require more than a one-line response; to move to the next prompt, press CTRL-D.

B.5.6 Tilde-Escape Sequences

The *contact* utility treats input beginning with a tilde (~) as a special sequence. The character following the tilde is considered a request for a special function. The following tilde sequences are recognized by *contact*:

~e	Start the text editor (defined in your EDITOR environment variable).
~h	Display a list of available tilde-escape sequences.
~p	Print the contact report to the terminal screen.
~r *filename*	Read the contents of *filename* as a response to the current prompt. Some prompts require only a one-line response. This tilde-escape sequence only works for prompts that allow more a than one-line response.
~~	Insert a single tilde as the first character in the line.

B.6 Using the *contact* Utility

The *contact* utility prompts for the following information:

- your name, title, phone number, and corporate name
- the name and version of the product involved
- a one-line summary of the problem
- a detailed description of the problem
- the priority of the problem
- instructions on how to reproduce the problem
- comments about the problem
- comments about the documentation supporting the problem
- files to include in the contact report

The following is a step-by-step discussion of these prompts:

1a. To invoke the *contact* utility, enter **contact** at the system prompt. The system responds with a welcome message and a series of questions regarding your hardware, software, or documentation question. The following screen illustrates the *contact* command and the system response:

```
>contact
Welcome to contact version 0.11 ()

Enter your name, title, phone number, and corporate name (^D to terminate)
>
```

1b. If there is a *.contact* file in your home directory, *contact* skips the first prompt. The following screen illustrates the *contact* command and the system response when a *.contact* file is in your home directory:

```
>contact
Welcome to contact version 0.11 ()

Enter the name of the product involved
>
```

2. The *contact* utility prompts for the version number of the product. If you do not know the version number, use CTRL-Z to suspend the session. Use the *which* (or *whence* if using *csh*) and *vers* commands to find the version number of the product. Use the *fg* command to return to the session and enter the version number in the form X.X or X.X.X.X.
3. The *contact* utility prompts for a one-line summary of the problem. This summary is the subject header in any further correspondence regarding the problem. Make this summary as descriptive as possible in one line.
4. The *contact* utility prompts for a detailed description of the problem. Make this description as complete as possible. Include source code and a stack backtrace whenever possible. (Refer to the *adb*(1) or *csd*(1) man page for information on obtaining a stack backtrace.) The more information provided, the quicker the TAC can isolate and solve the problem.
5. The *contact* utility prompts for the priority of the problem. The following screen illustrates this prompt and the priority levels from which to choose; you must enter a priority number.

```
Enter a problem priority, based on the following:
1) Critical    - work cannot proceed until the problem is resolved.
2) Serious     - work can proceed around the problem, with difficulty.
3) Necessary   - problem has to be fixed.
4) Annoying    - problem is bothersome.
5) Enhancement - requested enhancement.
6) Informative - for informational purposes only.
>
```

6. The *contact* utility prompts for an explanation of how to reproduce the problem. Include the command syntax and options you used and anything else you did to make your program run.
7. The *contact* utility prompts for any other pertinent comments. Include any relevant information.
8. The *contact* utility prompts for suggestions regarding the documentation supporting the product. Indicate if the documentation could be revised to address the question.
9. The *contact* utility asks for the names of files necessary to reproduce the problem. The following screen illustrates the *contact* prompt and sample user response:

```
Are there any files that should be included in this report (yes | no)?
>yes
Please enter the names of the files, one to a line (^D to terminate)
>test.f
>~/subroutines/sub.f
>
```

> **NOTE**
>
> Tilde-escape sequences are not recognized in responses to this prompt. Instead, *contact* treats a tilde in this section to mean your home directory. This convention is based on use of the tilde for expanding file names in *csh*.

If the files specified are small text files, they are automatically included in the contact report. If the files are too big to be included in this report, *contact* gives further instructions on how to submit these files.

To specify a directory, combine the directory files into a single file using the *tar* command (refer to the *tar*(1) man page for further information) or enter each file name in the directory on a single line in the contact report.

10. The *contact* utility prompts you to review, edit, submit, or abort the contact report. The following screen illustrates this prompt:

```
Please select one of the following options:
1) Review the problem report.
2) Edit the problem report.
3) Submit the problem report.
4) Abort the problem report.
>
```

Choose the number of the option you want to select. These options let you do the following:

Review	Review the text of your contact report. You are then prompted again to select an option.
Edit	Edit the text of the contact report. If you choose to edit the report, *contact* puts you in your default text editor.
Submit	Send the report to the CONVEX TAC. You are notified within 48 hours that the TAC has received the report. This option exits the *contact* utility and returns you to the shell environment.
Abort	Save the text of your report in a file named *dead.report* in your home directory. This option exits the *contact* utility and returns you to the shell environment.

Index

S

T

U

V

W

Electronic Mail

Using Electronic Mail

The Hardware Documentation Group has an email address for documentation comments. Use this service to give us a quick response mechanism if you have special documentation questions that you would like addressed immediately. If you have a technical question, you should still contact the Technical Assistance Center, as described previously. To use email response service, just send mail addressed to:

```
cnvxhwdoc@convex.COM
```

We will read your comments and give you a personal reply.

What to Include in an Email Message

When you use the electronic mail service, please provide the following information:

- The reader's name and company name
- A return email address in INTERNET notation or UUCP (bang) notation
- The manual that is being critiqued
- The chapter and page number in question
- The comment

Reader's Forum

If you wish to mail your comments to us, please use the form on the next page and list the document page number with your questions and comments. Thank you.

THIS PAGE INTENTIONALLY LEFT BLANK

CONVEX VMEbus Async/Printer Controller Service Guide
Document No. 081-001430-000, First Edition

Reader's Forum

Please use this form to submit comments or questions concerning the clarity and service of this manual. Constructive critical comments are most welcome and help us continue in our efforts to generate quality customer documentation. Please list the page number for questions or comments.

_____

_____

_____

_____

_____

_____

_____

_____

From:

Name_____ Title_____

Company_____ Date_____

Address and Phone No._____

FOR ADDITIONAL INFORMATION OR DOCUMENTATION:

Location	Phone Number
From all locations in continental U.S.	1(800)952-0379
From locations in Alaska, Hawaii, & Canada	1(214)497-4379
From all other locations	Contact nearest CONVEX office

Direct mail orders to:

CONVEX Computer Corporation
Customer Service
PO Box 833851
Richardson TX 75083-3851 USA

(Fold Here First)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 1046 RICHARDSON, TEXAS

POSTAGE WILL BE PAID BY ADDRESSEE

CONVEX Computer Corporation
Customer Service
PO Box 833851
Richardson TX 75083-3851

(Fold Here Second)

(Tape or Staple)