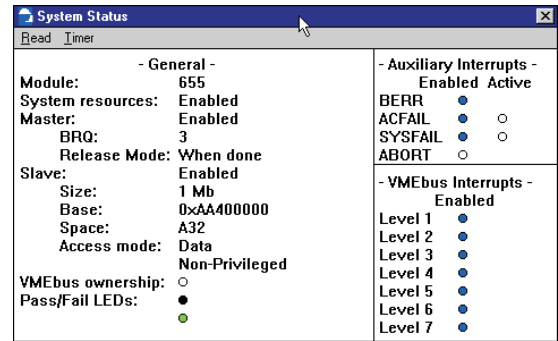




XVME-984/2 Windows® 3.1 Board Support Package

Xycom Automation's Windows® 3.1 Board Support Package is a comprehensive software package that simplifies the process of designing VMEbus application programs for Xycom Automation VME PC hardware in a Windows 3.1 environment. The package consists of *VMEman* (a collection of Windows 3.1 applications to access the VMEbus), a Library Demonstration Program, a 16-bit Windows 3.1 kernel mode driver for the Tundra Universe PCI-to-VME bridge chip, and a 16-bit Application Program Interface (API).



FEATURES

- The VMEbus manager (*VMEman*) Windows 3.1 application, featuring *Monitor*, *Probe*, and *Interrupts* menus
- Library Demonstration Program
- 16-bit Windows 3.1 driver for Tundra Universe PCI-to-VME bridge chip
- Interactive driver configuration application
- 16-bit API
 - VMEbus read and write routines with DMA support
 - Dual-access memory management functions
 - Support for read-modify-write operations
 - Virtual memory mapping operation for VMEbus address space
 - VMEbus interrupt generation routines
 - Routines for handling VMEbus interrupts
- Windows 3.1 memory protection when sharing memory between processors

BENEFITS

Get a Comprehensive Package

With the VMEbus manager, Library Demonstration Program, driver for Tundra Universe PCI-to-VME bridge chip, and API, Xycom Automation's Windows 3.1 Board Support Package is a comprehensive solution for your VME board support needs.

Simple Configuration and Testing of VME Systems

The *VMEman* Windows 3.1 application provides simple tools for configuring and testing VME systems.

Windows Interface

The VMEbus manager and Library Demonstration Program includes easy-to-use, fill-in-the-blank dialogs and drop-down menus.

Save Development Time

The 16-bit API provides a library of routines, which allows users to get their applications up and running quickly by simplifying the design and implementation of Windows 3.1 applications for Xycom Automation VMEbus PC processors. The development time and effort required to handle VMEbus data and interrupts in Windows 3.1 applications is significantly reduced.

A Range of Modules Supported

The Windows 3.1 Board Support Package supports:

XVME processor modules

- XVME-653
- XVME-654
- XVME-656
- Legacy Boards
- Digital I/O**
- XVME-200/290
- XVME-201
- XVME-202
- XVME-212
- XVME-220
- XVME-240

XVME-244

- XVME-260

Analog I/O

- XVME-500/590
- XVME-505/595
- XVME-530
- XVME-531
- XVME-540
- XVME-542
- XVME-545
- XVME-560
- XVME-564
- XVME-566

Counter modules

- XVME-203/293
- XVME-230

VMEBUS MANAGER

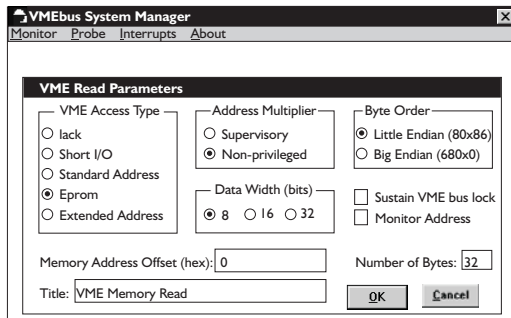
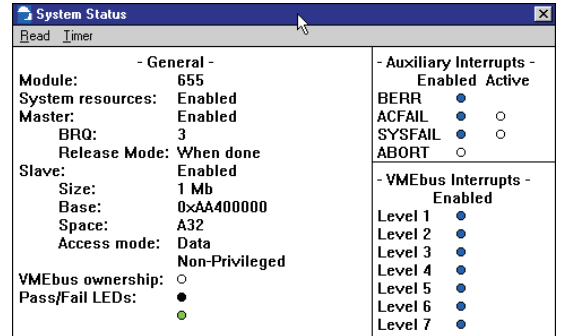
The VMEbus Manager, *VMEman*, is a Windows application program for accessing and monitoring the VMEbus. When using a Xycom Automation PC processor, the program allows you to quickly test your VMEbus system configuration with a convenient mechanism for reading and writing to VMEbus memory and monitoring the status of VMEbus interrupts and the PC status registers.

The VMEbus Manager functions are grouped into three menus.

Monitor Menu

The System Status option opens a window that displays the VMEbus address and size of the dual access memory, along with the onboard battery condition. Simulated LEDs indicate the state of the seven VMEbus interrupt signals, as well as ACFAIL, SYSFAIL, VMEbus ownership, condition of the auxiliary nonmaskable interrupts, BERR, ABORT push button, watchdog timer, and the front panel pass/fail LEDs.

VME IDs will search the VMEbus Short I/O address space for any modules that conform to the Xycom Automation standard I/O architecture. If any conforming modules are found, the address and a brief description of each module is displayed.



Probe Menu

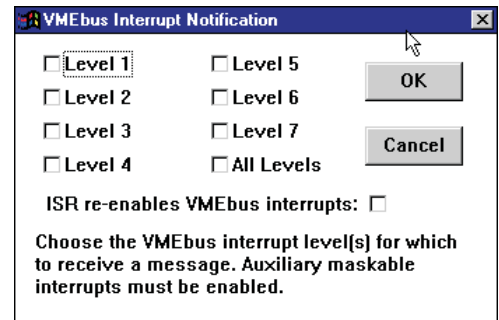
The Probe Menu provides the option of performing reads and/or writes to any of the VMEbus address spaces or to a dual access memory window. Reads and writes can be performed continuously at timed intervals you select.

VMEbus reads can search through memory for a pattern. VMEbus writes can perform a blockfill with a pattern and block size you specify.

You can also manipulate the state of the front panel pass and fail LEDs and VMEbus ownership.

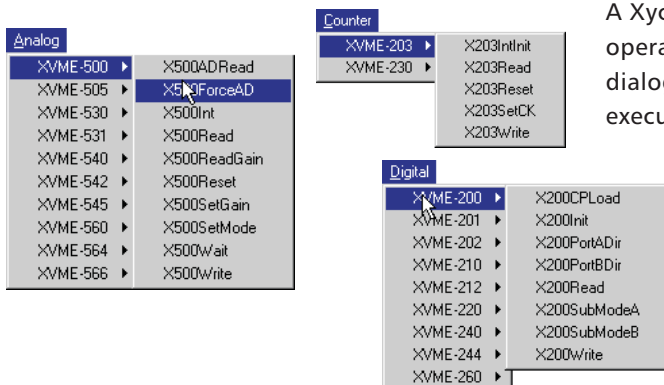
Interrupts Menu

Any of the following interrupts can be enabled or disabled: auxiliary nonmaskable and maskable, BERR/IOCHCK, and watchdog timer. A VMEbus Interrupt Notify Level can also be set, which when enabled, displays a window containing the VMEbus interrupt level and vector returned during an interrupt acknowledge cycle. In addition, you can generate or acknowledge VMEbus interrupts on any of the seven interrupt levels.



LIBRARY DEMONSTRATION PROGRAM

The Library Demonstration Program provides examples of how to use the routines contained in the Dynamic Link Library (DLL). The Demonstration Program can be used to read from or write to Xycom Automation's family of VMEbus I/O modules. The program source code is included.



A Xycom Automation I/O module can be configured and its operation verified through the use of drop-down menus and dialog boxes. Each of the DLL routines can be configured and executed by selecting an I/O module from an Analog, Counter, or Digital drop-down menu.

The Interrupts menu (not shown) allows you to select one or all of the VMEbus interrupt levels to display an interrupts notification window when an interrupt is generated. This feature is especially useful for debugging the interrupt system.

When used with the VMEbus Manager, the Library Demonstration Program is a useful tool during system implementation.

DEVICE DRIVER

A Windows 3.1 16-bit kernel mode driver takes full advantage of the Tundra Universe PCI-to-VMEbus bridge chip. This driver performs all of the manipulations needed to access the VMEbus, allocate and access dual-access memory, handle and generate VMEbus interrupts on all levels, and access control registers. Driver configurations can be adjusted using the VMEbus Manager.

APPLICATION PROGRAM INTERFACE (API)

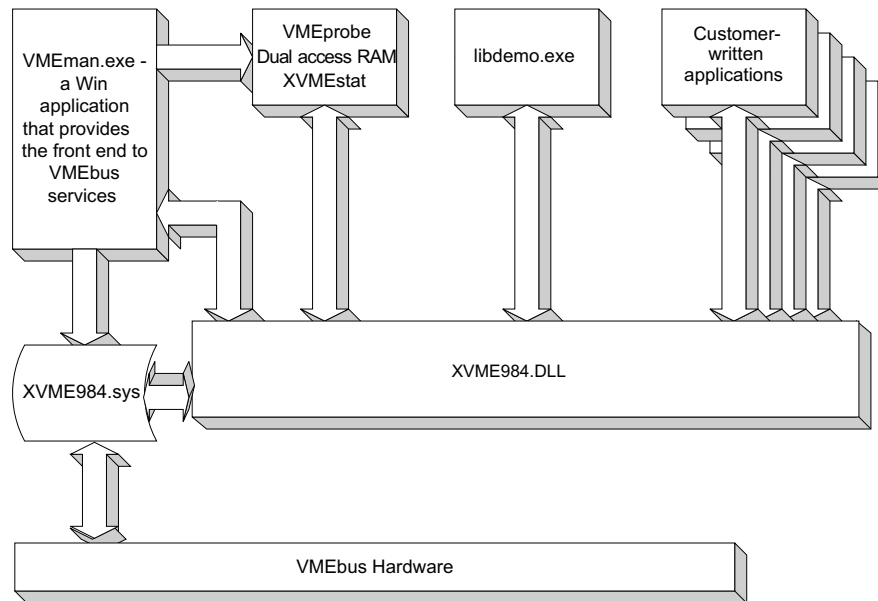
The 984/2 Dynamic Link Libraries provide a 16-bit, high-level API between an application and the Windows 3.1 driver. This interface is a comprehensive set of functions that simplifies the development of VMEbus application programs.

General Purpose

Routine	Description
BOOL AllocDualPortMem	Allocates a contiguous block of memory to use for dual access
void DisableNMIInt	Disables the auxiliary NMI error conditions
void DisableVMEInterrupts	Disables the auxiliary maskable interrupts used for VMEbus interrupt levels 1-7
void DisableWDTimer	Disables the Watchdog timer
void EnableNMIInt	Enables the auxiliary NMI error conditions
void EnableVMEInterrupts	Enables the auxiliary maskable interrupts used for VMEbus interrupt levels 1-7
void EnableWDTimer	Enables the Watchdog timer
BOOL FreeDualPortMem	Frees the memory that was previously allocated using AllocDualPortMem
WORD GenVMEBusInt	Generates an interrupt on the VMEbus
DWORD GetIntAddress	Returns the address of the interrupt handler for the interrupt vector specified by the vector parameter
void GetRMWdata8	Transfers a number of bytes from the real mode window to a local buffer
void GetRMWdata16	Transfers a number of words from the real mode window to a local buffer
void GetRMWdata32	Transfers a number of double words from the real mode window to a local buffer
WORD IsXVMEDosDeviceDriver Installed	Determines if the supporting MS-DOS device driver XVME984.SYS is installed
BYTE LockVMEBus	Requests the VMEbus and then waits for access
void Mask8259	Sends the appropriate mask to the slave 8259 interrupt controller to either set or reset the IRQ level
BYTE NotifyOnVMEInterrupt	Adds the window handle (hWnd) to the list of window handles to be notified when a VMEbus interrupt occurs
DWORD FAR PASCAL PutIntAddress	Sets an interrupt vector to point to an interrupt handling routine
void PutRMWdata8	Transfers a number of bytes from a local buffer to an offset in the real mode window
void PutRMWdata16	Transfers a number of words from a local buffer to an offset in the real mode window
void PutRMWdata32	Transfers a number of double words from a local buffer to an offset in the real mode window
void QuitNotifyOnVMEInterrupt	Removes the window handle (hWnd) from the list of window handles to be notified when a VMEbus interrupt occurs
DWORD Readtimecptr	Returns the system-timer time counter
WORD ReadVMEBusMemoryRM	Transfers a block of memory from the specified VMEbus memory to the processor module through the real mode window
WORD Read_Flag_Reg	Returns the contents of the CPU flag register
void ReleaseVMEBus	Releases the VMEbus from XVME-CPU control
void ResetWDTimer	Resets the Watchdog timer after it has timed out
void Restore_Flag_Reg	Changes the contents of the CPU flag register
WORD SendA32ToXVMEDosDevice Driver	Sends a 32-bit address to the XVME948.SYS MS-DOS Device Driver
Set_RM_Window	Sets the real mode window to the desired VMEbus address space
void StrobeWDTimer	Retrigger the Watchdog timer
WORD WriteVMEBusMemoryRM	Transfers a block of memory from the processor module to the specified VMEbus memory through the real mode window
WORD XvmeCPUType	Returns the CPU type (as defined in XVME984.H) of the XVME-CPU board as determined upon initialization

Digital I/O, Analog I/O, Counter I/O, and Interrupt routines also are included.

BOARD SUPPORT PACKAGE COMPONENTS



ORDERING INFORMATION

Standard Configurations

Order Number	Description
XVME-984/2	Windows 3.1 Board Support Package

Xycom Automation, Inc.
 734-429-4971 • Fax: 734-429-1010
<http://www.xycomautomation.com>

Canada Sales 905-607-3400
Northern Europe Sales +44-1604-790-767
Southern Europe Sales +39-011-770-53-11



© 2000 Xycom Automation, Inc. Specifications may change without notice. Xycom Automation is a trademark of Xycom Automation. Other brand or product names are the property of their respective owners.