# VIRTUAL MBX DRIVER

## Virtual MBX® Driver for Modbus Plus Host Interface Adapters, Modbus, Modbus Plus and TCP/IP Networks

## Version 6.0 for Windows® XP/2000/NT/Server 2003



Document last revision date March 6, 2006

# TABLE OF CONTENTS

**Introduction** ..... **3**

**MBX Architecture and Companion Products** ..... **4**
MBX Driver ..... 5
Ethernet MBX Driver ..... 5
Serial MBX Driver ..... 5
MBX Gateway Driver ..... 6
Virtual MBX Driver ..... 6
MBX Bridge ..... 6
MBX OPC Server ..... 7
MBX SDK ..... 7
Blending MBX Supported Networks ..... 8

**Theory of Operation** ..... **9**

**Configuration** ..... **11**
Typical Driver Configuration for DOS Applications ..... 11
Creating Shortcuts with the Virtual MBX Configuration Editor ..... 12
Creating Shortcuts with Windows NT Explorer ..... 13
Typical Driver Configuration for Win16 Applications ..... 18
Virtual MBX Driver Configuration Editor ..... 20
16-bit Windows Applications Tab ..... 20
16-bit DOS Applications Tab ..... 21
Product Version Tab ..... 21

**Validation & Troubleshooting** ..... **22**

# INTRODUCTION

The Virtual MBX Driver allows all existing 16-bit DOS or Windows 3.x NETLIB/NetBIOS-compatible applications to run under Windows XP/2000/NT in their original binary form. This includes programs such as Modsoft, ModLink, MBPSTAT and hundreds of other custom applications written by software developers and system integrators. All of these applications can now run concurrently with other 32-bit applications, sharing the same 32-bit device driver.

The Virtual MBX Driver fully emulates the operation of the DOS based SA85.SYS and MBPHOST.SYS drivers from Modicon. Therefore, the SA85.SYS and MBPHOST.SYS drivers are no longer needed and should not be used in Windows XP/2000/NT.

The Virtual MBX Driver fully supports all NETLIB/NetBIOS features. This includes support for Data Master/Slave, Program Master/Slave, Global Data and Peer Cop. The Virtual MBX Driver uses the MBX Driver, Ethernet MBX Driver and Serial MBX Driver for all of its local I/O functions. It can also use the MBX Gateway Driver for network-based installations. The MBX Driver, Serial MBX Driver, Ethernet MBX Driver or MBX Gateway Driver is required for the proper operation of the Virtual MBX Driver.

The Virtual MBX Driver provides an excellent bridge for users migrating to Windows XP/2000/NT who are not ready to abandon their favorite 16-bit applications. Once the drivers are installed, you can execute 16-bit programs exactly the same way as in their original environment. In addition, you can concurrently execute multiple instances of the same applications. For example, you can run multiple instances of MBPSTAT.EXE and monitor the operation of multiple Modbus Plus networks.

## What Should I Do Next?

The Cyberlogic MBX family for Windows XP/2000/NT consists of several well-integrated products that provide connectivity for Modicon Modbus®, Modbus Plus® and Ethernet networks in distributed environments. For more information about these products, refer to the MBX Architecture and Companion Products section.

For architectural and implementation details of the Virtual MBX Driver product, read the Theory of Operation section.

After installation, some files may need to be properly set up for the Virtual MBX Driver to operate. You will find information on this topic in the Configuration section.

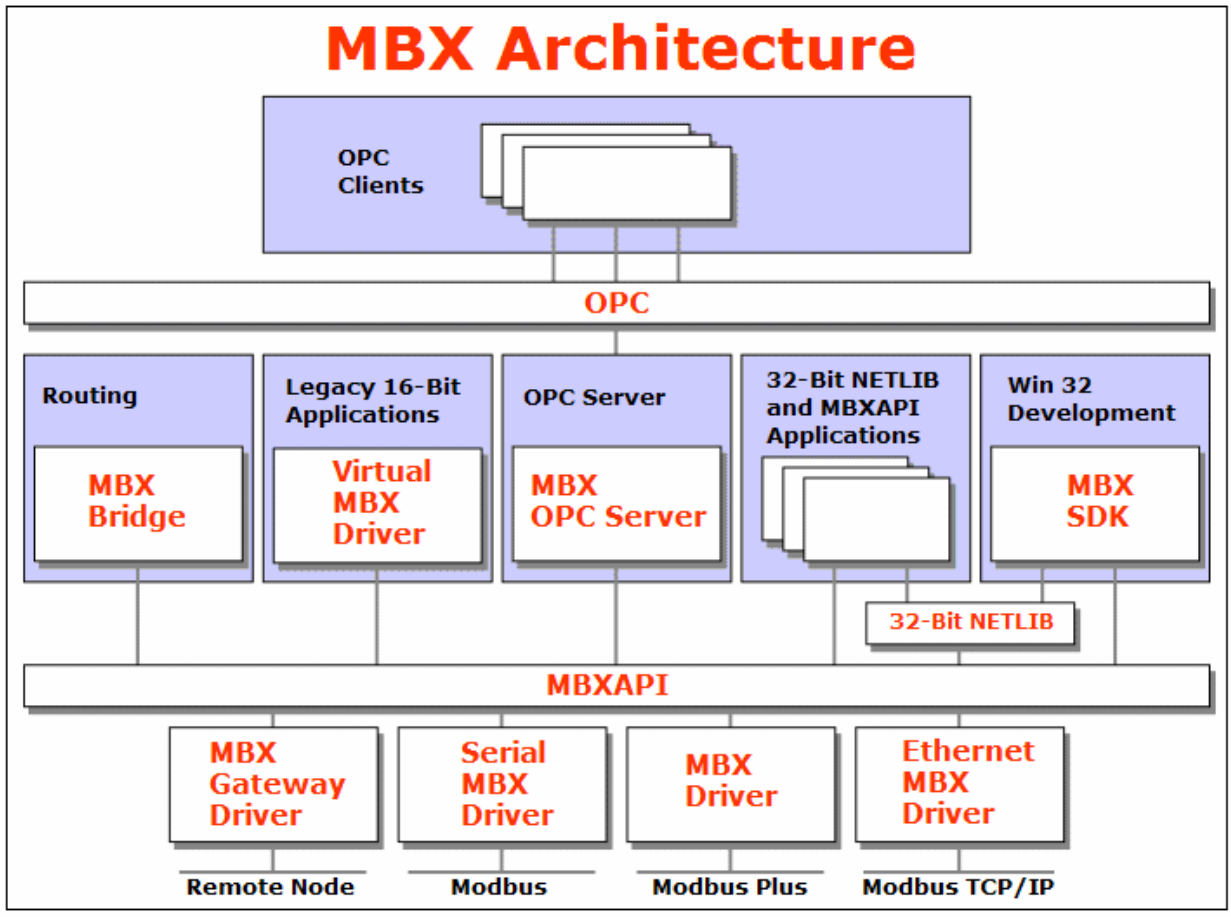
If you have already configured the driver, refer to the Validation & Troubleshooting section to verify that it operates as expected. This section also include a list of frequently asked questions to help with solving communications problems.

The content of this document is also provided in the PDF file format. PDF files can be viewed using the Adobe® Reader program. The printer-friendly PDF files can be used to print the complete document with good quality output.

# MBX ARCHITECTURE AND COMPANION PRODUCTS

This section illustrates the layout of the MBX architecture. It includes a description of each MBX product along with suggested methods for employing these products to support Modicon networks.

The Cyberlogic MBX family for Windows XP/2000/NT consists of several well-integrated products that provide connectivity for Modicon's Modbus, Modbus Plus and Modbus TCP/IP (Ethernet) networks in distributed environments.




*The MBX architecture presents a consistent framework to address different connectivity needs.*

Software products available in the MBX family are:

MBX Driver: This is Cyberlogic's device driver for Modbus Plus host interface adapters. The MBX Gateway Server is included for remote connectivity.

Ethernet MBX Driver: This driver provides Modbus Plus emulation over TCP/IP. The MBX Gateway Server is included for remote connectivity.

Serial MBX Driver: This driver provides Modbus Plus emulation over serial Modbus. The MBX Gateway Server is included for remote connectivity.

MBX Gateway Driver: This product provides access to Modicon's Modbus, Modbus Plus and Modbus TCP/IP networks from remote locations.

Virtual MBX Driver: This driver works with the other MBX drivers to permit 16-bit legacy software to run in 32-bit Windows operating systems.

MBX Bridge: This product allows you to bridge any combination of Modicon networks by routing messages between MBX devices.

MBX OPC Server: Cyberlogic's premium OPC Server connects OPC compliant client software applications to data sources over all Modicon networks.

MBX SDK: This is a software development kit for MBXAPI and NETLIB compliant development.

# MBX Driver

The 32-bit MBX Driver provides connectivity between Modicon ModConnect host interface adapters and 32-bit applications running under Windows XP/2000/NT.

The kernel mode device driver of the MBX Driver is the highest performance Modbus Plus driver in the industry. The driver operates in either interrupt or polled mode and supports all current Modicon ModConnect host interface adapters for ISA, EISA, MCA, PCI and PC Card (PCMCIA) buses. Multiple interface cards can be installed at the same time, limited only by the number of available slots. Full implementation of all Modbus Plus features provides support for Data Master/Slave, Program Master/Slave, Global Data and Peer Cop. The high-performance native API (MBXAPI) of the MBX Driver takes advantage of the event-driven, multitasking, multithreaded features of 32-bit operating systems.

The driver includes the MBX Gateway Server for remote access by the MBX Gateway Driver and is fully compatible with all other MBX family products.

# Ethernet MBX Driver

The 32-bit Ethernet MBX Driver provides connectivity between Modbus TCP/IP compatible processors and Windows XP/2000/NT based 32-bit applications using either Modicon NETLIB or Cyberlogic's high-performance MBXAPI interface specification. It provides Data Master/Slave and Program Master/Slave features of Modbus Plus on Ethernet networks.

The driver includes the MBX Gateway Server for remote access by the MBX Gateway Driver and is fully compatible with all other MBX family products. The Ethernet MBX Driver does not require a special Ethernet adapter. It is compatible with all Ethernet cards supported by Windows.

# Serial MBX Driver

The Serial MBX Driver provides connectivity to Modbus-compatible devices through the standard serial COM ports. It supports both master and slave node communications.

The driver includes the MBX Gateway Server for remote access by the MBX Gateway Driver and is fully compatible with all other MBX family products.

# MBX Gateway Driver

The MBX Gateway Driver lets you access Modbus, Modbus Plus and Modbus TCP/IP networks from a remote location. Through a standard LAN, your local applications can use MBX devices on Gateway nodes as though they were on your local system.

The remote client running the MBX Gateway Driver must be a Windows XP/2000/NT node. By accessing the Modbus, Modbus Plus and Ethernet networks connected to server nodes on a network, the MBX Gateway Driver provides complete MBX Driver functionality to the client node, including support for Data Master/Slave, Program Master/Slave, Global Data and Peer Cop. A host interface adapter, such as a Modicon SA85 card, is not required on the client node. MBX Gateway Driver nodes can communicate with multiple Gateway servers and all Windows XP/2000/NT-compatible computer networks are supported.

The MBX Gateway Driver is compatible with all other MBX family products.

# Virtual MBX Driver

The Virtual MBX Driver enables 16-bit NETLIB/NetBIOS-compatible applications, such as Modsoft and Concept, to run concurrently with 32-bit applications on the same computer. It allows multiple 16-bit applications and multiple instances of a single 16-bit application to run under the 32-bit Windows operating systems.

The Virtual MBX Driver is fully compatible with all MBX components and requires at least one of these drivers to operate:

- MBX Driver
- Ethernet MBX Driver
- Serial MBX Driver
- MBX Gateway Driver

# MBX Bridge

The MBX Bridge seamlessly routes messages between MBX-compatible devices. For example, the MBX Bridge can route messages between Ethernet and Modbus Plus networks, between Modbus and Modbus Plus networks or any other combination of the supported networks. Depending on the user's needs, it requires one or more of the following products to operate:

- MBX Driver
- Ethernet MBX Driver
- Serial MBX Driver
- MBX Gateway Driver

# MBX OPC Server

The Cyberlogic MBX OPC Server connects OPC-compliant clients to Modicon Modbus, Modbus Plus and Ethernet networks. It supports the latest OPC Data Access and OPC Alarms and Events specifications and uses the MBX drivers for connectivity to Modicon networks.

The MBX OPC Server supports multiple, priority-based access paths for reliable, redundant communications. It also supports both solicited and unsolicited communications and uses an advanced transaction optimizer to guarantee minimum load on your networks. With only a couple of mouse clicks, the MBX OPC Server will automatically detect and configure the attached networks and node devices in seconds. Other noteworthy features include DirectAccess, Data Write Protection and Health Watchdog.

# MBX SDK

Software developers can use the MBX SDK to provide connectivity to Modbus, Modbus Plus and Ethernet networks from their 32-bit C/C++ applications.

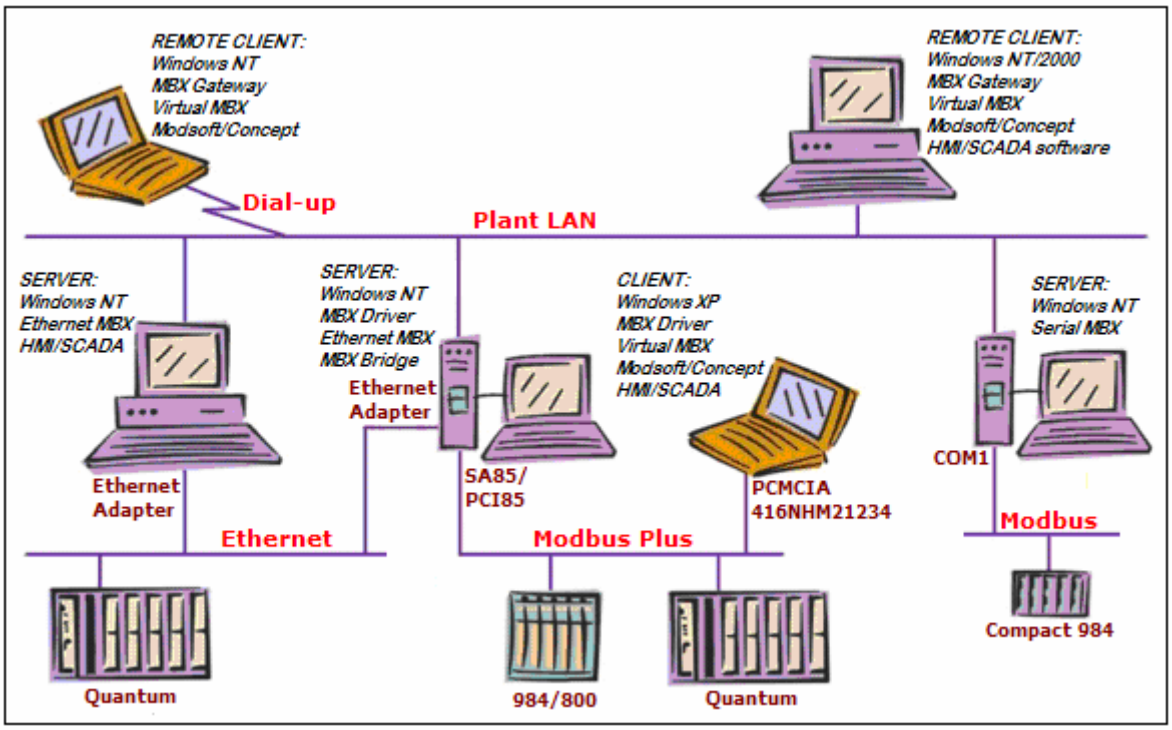
The SDK supports two styles of interfaces, the industry-standard NETLIB and Cyberlogic's high-performance MBXAPI. The NETLIB interface is an excellent bridge for developers who would like to port their 16-bit applications to the 32-bit Windows environments. Developers of new applications can use either the NETLIB or the MBXAPI interface.

Since all MBX driver products are built on the same MBX architecture, applications developed with the MBX SDK can be used with all MBX drivers and can execute under all 32-bit Windows operating systems.

# Blending MBX Supported Networks

The MBX driver products provide support for all Modicon networks through a common architecture, with identical programming interfaces: the MBXAPI and the industry-standard NETLIB. This ensures that virtually all of the existing Modbus Plus compatible software programs can operate over all Modicon supported networks with no code modifications. A product operating with one of the MBX driver products, such as the MBX Driver, will operate with the rest of the MBX driver products as well.

Migration of existing installations to new hardware products does not require the user to discard working, proven software solutions. As depicted in the following diagram, a user can now mix Modbus, Modbus Plus and Ethernet based hardware products in existing installations without losing software, network or integration investment.




*MBX enabled system deployment:*
*New hardware solutions will blend into existing installations without software or network modifications*

# THEORY OF OPERATION

This section is intended to familiarize you with the main features of the Virtual MBX Driver.

Prior to 32-bit Windows operating systems, DOS and 16-bit Windows applications used Modbus Plus communications through the use of a DOS device driver called MBPHOST.SYS (previously called SA85.SYS). This driver implements the NetBIOS-style interface used in other standard networks. All NetBIOS commands were sent to the driver through the pre-selected software vector, typically vector 5C hex.

Some applications communicated to the driver through the direct NetBIOS calls. To simplify the driver's interface, Modicon introduced a simpler application programming interface (API) called NETLIB. This new interface became the standard API for Modbus Plus communications.

## Virtual MBX Driver Architecture

The Virtual MBX Driver completely emulates the operation of the MBPHOST.SYS and SA85.SYS drivers. It fully supports all NETLIB/NetBIOS features. This includes support for Data Master/Slave, Program Master/Slave, Global Data and Peer Cop. Once the driver is installed, any existing DOS or Win16 application can run under Windows XP/2000/NT without any code modifications.

The Virtual MBX Driver uses the MBX Driver, Ethernet MBX Driver or Serial MBX Driver for all of its local I/O functions. It can also use the MBX Gateway Driver for network-based remote installations.

Windows XP/2000/NT provides a Virtual DOS Machine (VDM) within the Win32 subsystem to support real-mode and protected-mode MS-DOS applications. Windows XP/2000/NT launches a VDM instance, which is necessary to support MS-DOS or Windows 3.x applications. Several copies of the VDM may run simultaneously without interfering with each other. The applications run independently, as they would in a native MS-DOS/Windows 3.x environment.

When the VDM is launched, it:

1. Allocates memory to create a virtual DOS machine

2. Reads the files …\config.nt and …\autoexec.nt

3. Loads drivers and executes batch files as specified

4. Loads the application and executes it

The config.nt and autoexec.nt files located in the system32 directory are global for the entire Windows XP/2000/NT system. If you use a shortcut to launch your DOS application, you can specify private versions of these files, which may be located in any directory. However, 16-bit Windows applications must use the global config.nt and autoexec.nt files located in the system32 directory.

To run the Virtual MBX Driver with your application, the config.nt file must contain the following line:

*device=%SystemRoot%\System\mbp16.sys*

The Virtual MBX Driver also must be configured by adding the following line to the autoexec.nt or by typing from the command prompt:

*%SystemRoot%\System\mbp16vec {NetBIOS_vector}*

The NetBIOS_vector identifies the software interrupt vector used by the NetBIOS interface. If not specified, the default value (5C hex) is used. Once the above lines have executed, the Virtual MBX Driver is fully installed.

Although 16-bit DOS applications do not require a shortcut to run, we recommend that all 16-bit DOS applications use shortcuts. This allows the user to configure each VDM for a specific application using different autoexec.nt and config.nt files. A 16-bit Windows (Win16) application, such as Concept, will need the Virtual MBX Driver loaded globally. In this case, insert the lines mentioned above into the global config.nt and autoexec.nt files located in the system32 directory and restart the system. For more information on how to add these lines for Win16 applications, refer to the Configuration section.

The Virtual MBX Driver Setup program installs example shortcuts and sample versions of the autoexec.nt and config.nt files containing the above commands in the installation directory (the default directory is *\Program Files\Cyberlogic\Virtual MBX Driver*). For information on how to configure shortcuts to other 16-bit DOS applications, refer to the Configuration section.

# CONFIGURATION

The Virtual MBX Driver Setup program installs example shortcuts to MBPSTAT and Modsoft. If these are the only applications you plan to use, no additional configuration is required. However, the Virtual MBX Driver provides an editor that can configure the driver for additional applications.

The procedures you will use are broken into several sections:

- Typical Driver Configuration for DOS Applications is a good place to start if you have a DOS application and you are a first-time user. It is a tutorial that walks you through a complete driver configuration session.
- Typical Driver Configuration for Win16 Applications is the place to start for first-time users with 16-bit Windows applications. This, too, is a tutorial that walks you through a complete driver configuration session.
- Virtual MBX Driver Configuration Editor describes in detail each feature of the editor.

# Typical Driver Configuration for DOS Applications

This section shows a typical configuration session. Use it only as a guideline. Only the most common features are shown here. For detailed descriptions, refer to the Virtual MBX Driver Configuration Editor section.

Under Windows NT, you can start a DOS application from either a DOS command prompt or a shortcut. If you prefer a command prompt, use the *Command Prompt with vMBX* icon located in the *Virtual MBX Driver* group created by the installation program. If you use this icon, you need not do any additional configuration.

However, we recommend that all 16-bit DOS applications use shortcuts. This allows you to configure your application's VDM by using private versions of the autoexec.nt and config.nt files.

During the Virtual MBX Driver installation, pre-configured versions of autoexec.nt and config.nt files are placed in the product directory. In most cases, these files can be used without any modifications. In that case, follow the Creating Shortcuts with the Virtual MBX Configuration Editor section.

Shortcuts can also be created directly from the Windows NT Explorer. Using Explorer is more complicated, but it allows you to select custom versions of the autoexec.nt and config.nt files. If you prefer to use Explorer, follow the Creating Shortcuts with Windows NT Explorer section.

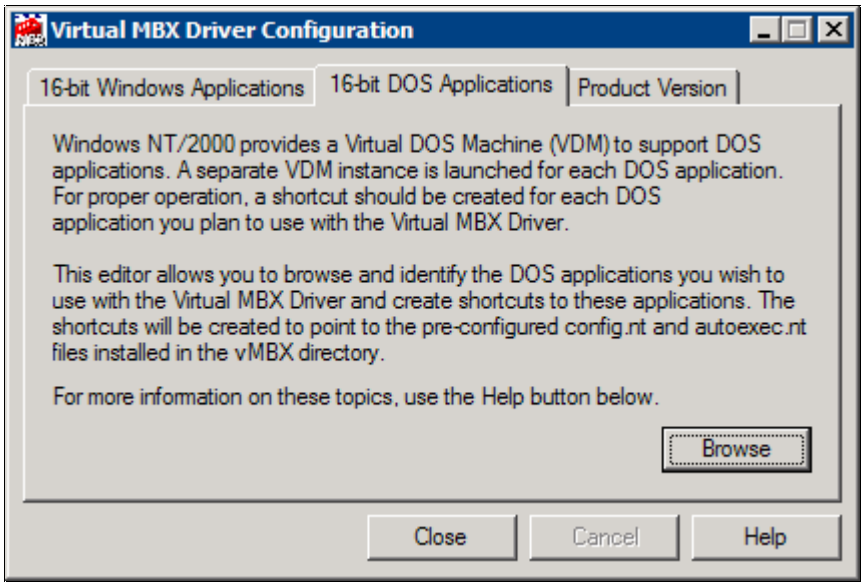
# *Creating Shortcuts with the Virtual MBX Configuration Editor*

This section describes a procedure to configure your application's shortcut by using the Virtual MBX Driver Configuration Editor.

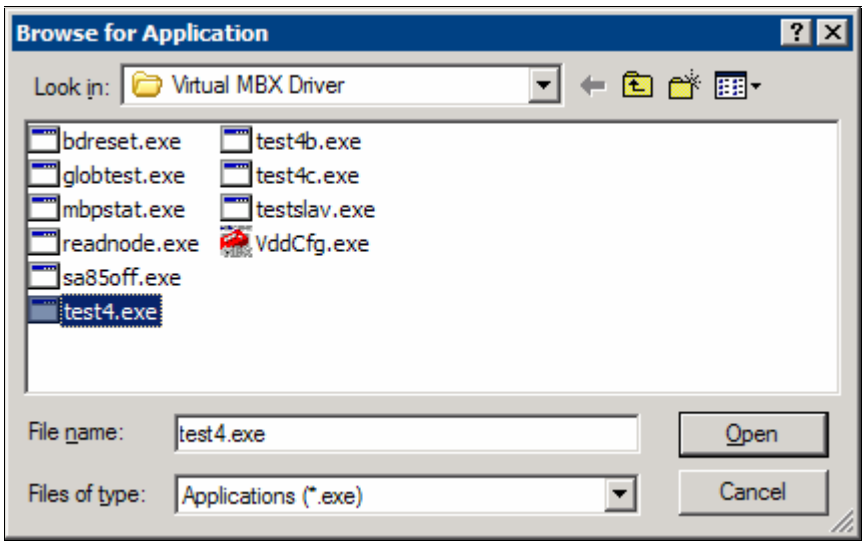
Shortcuts created with the Virtual MBX Configuration Editor will point to the pre-configured versions of the autoexec.nt and config.nt files placed in the product installation directory. As an example, we will create a shortcut to the test4.exe program located in the same directory.

1. From the Windows *Start* menu, locate the *MBX Virtual Driver* submenu and select the *Virtual MBX Configuration* menu item.

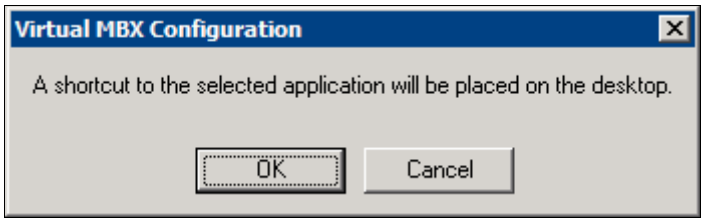
2. Select the *16-bit DOS Applications* tab, then click the *Browse* button.




3. Select the *test4.exe* program and then click the *Open* button.



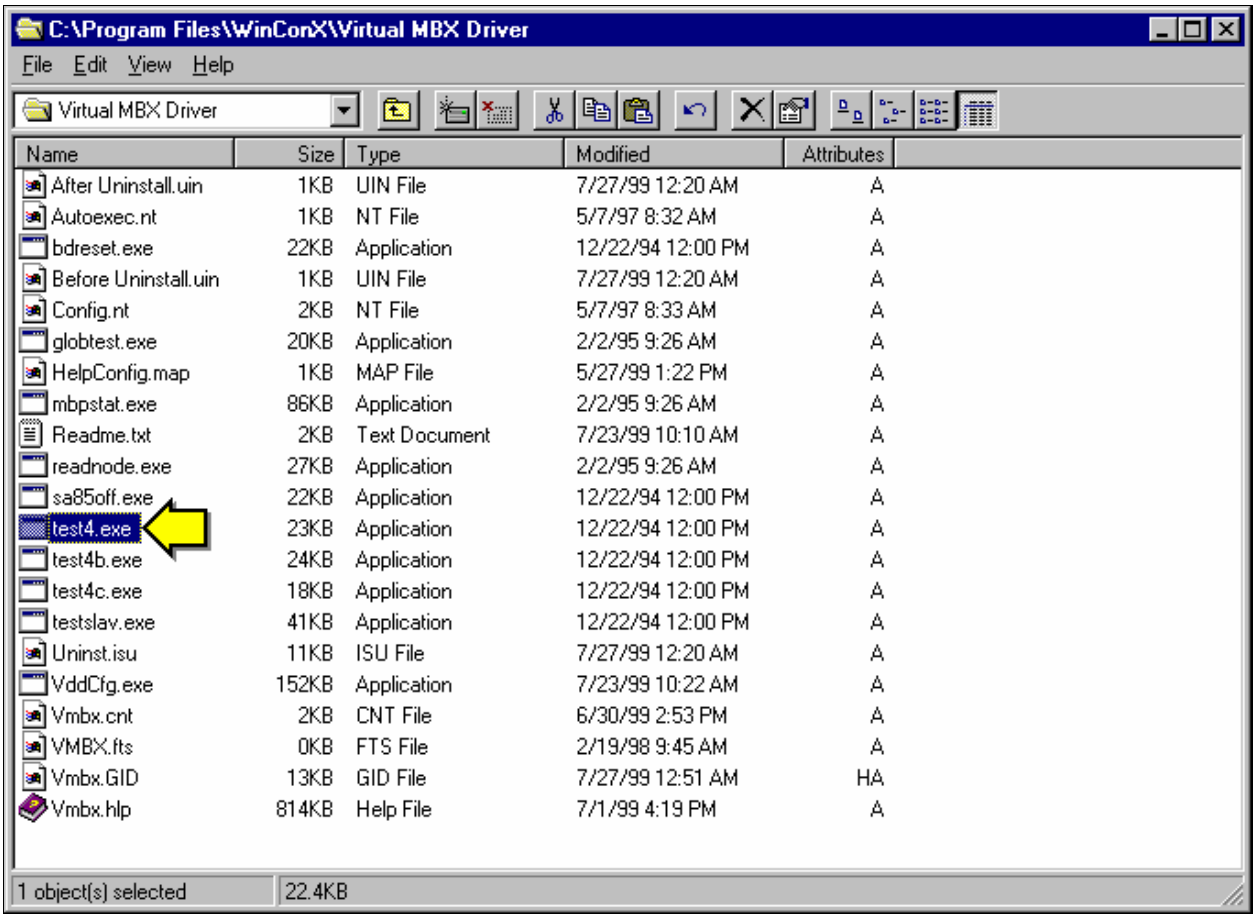

4. Click the *OK* button. A shortcut will be placed on the desktop. Use Windows NT Explorer to edit this shortcut or to move it to a different location.



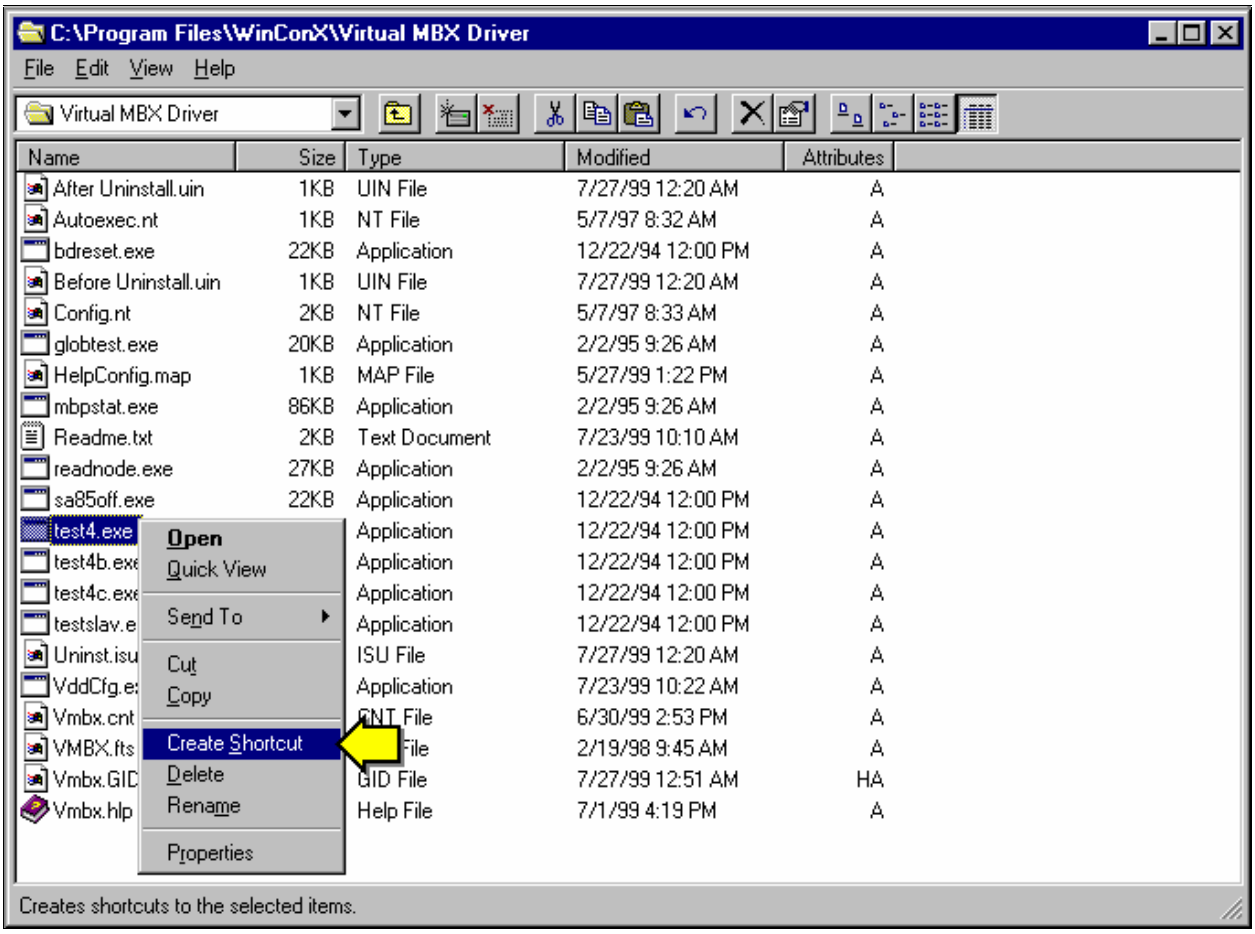

# *Creating Shortcuts with Windows NT Explorer*

This section describes the procedure to configure your application's shortcut by using Windows NT Explorer. As an example, we will use the 16-bit test4.exe program located in the product installation directory.

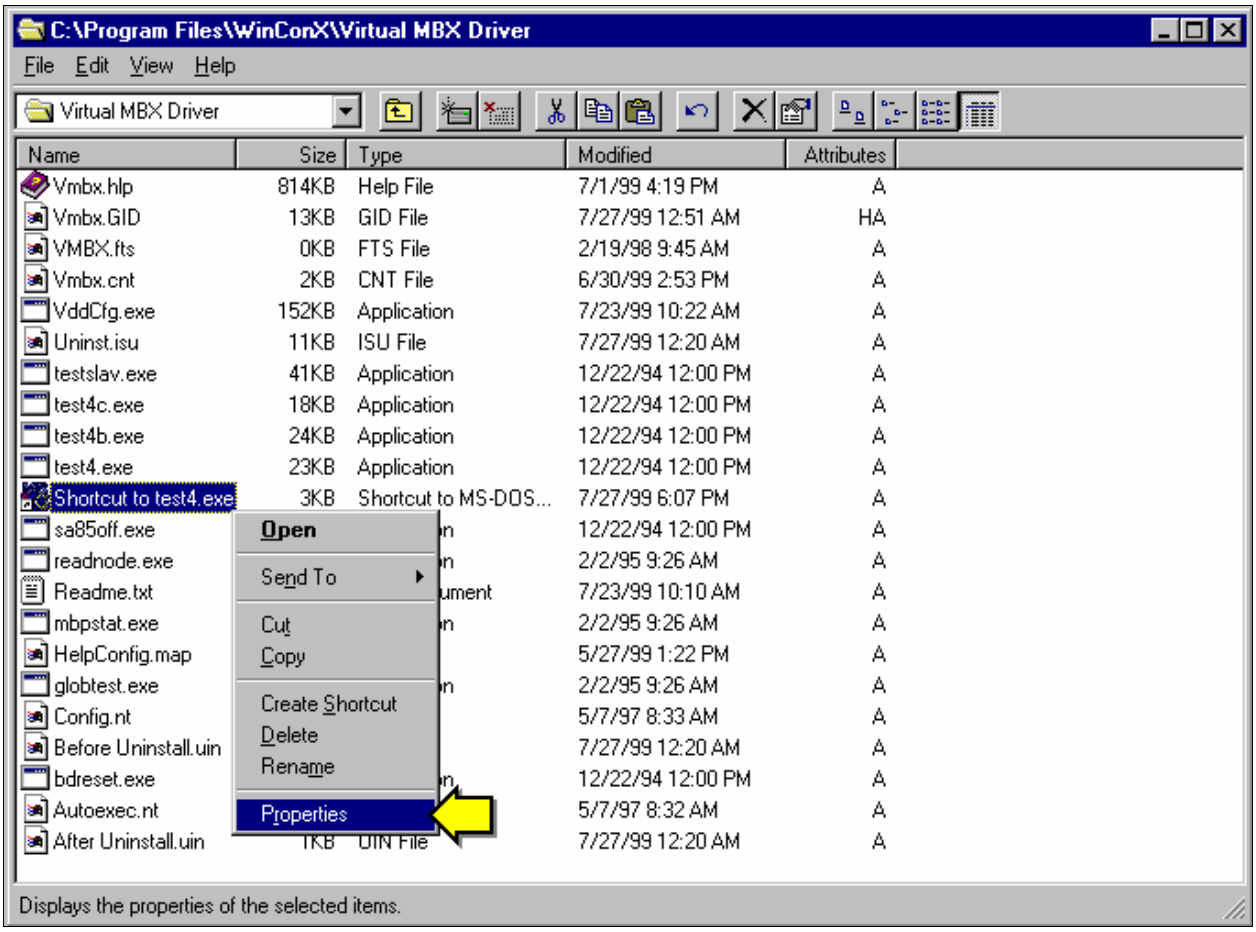
1. Start Windows NT Explorer and browse to the product installation directory (*\Program Files\Cyberlogic\Virtual MBX Driver*). Select *test4.exe*.



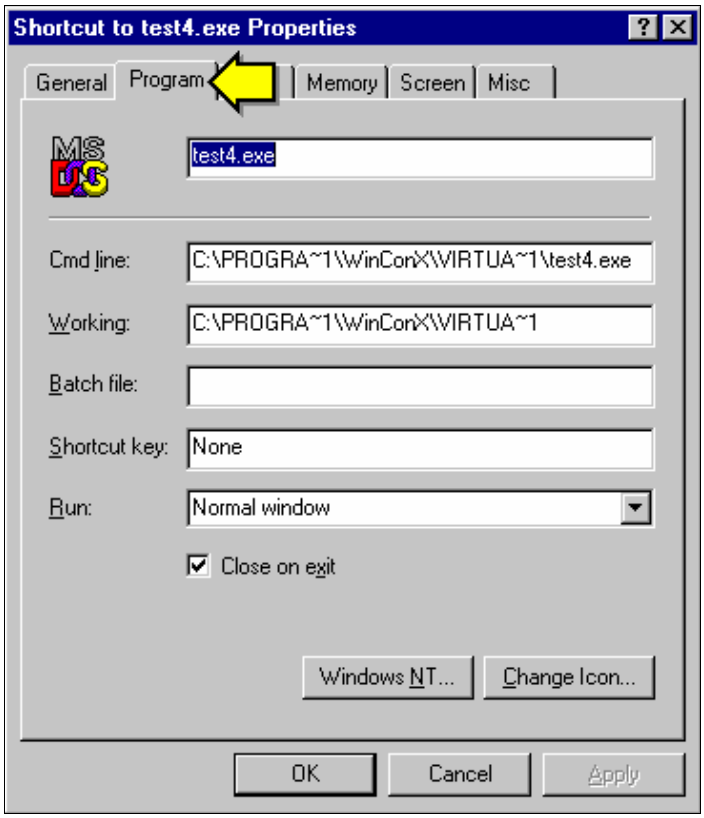
2. Right-click on *test4.exe* and select *Create Shortcut* from the menu. A shortcut to test4.exe will be created.



3. Select the *Shortcut to test4.exe*, then right-click and select *Properties* from the menu.

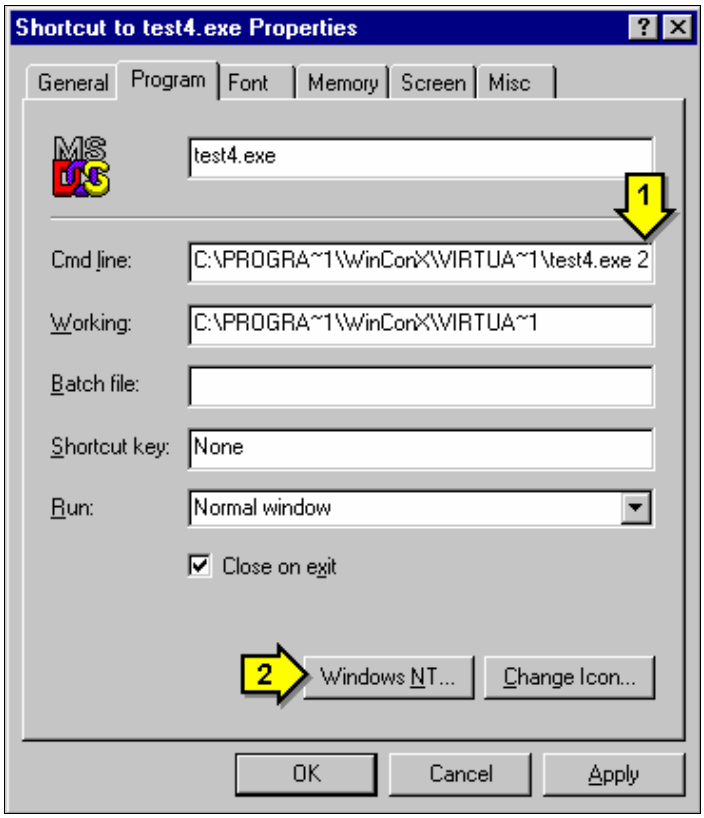


4. Select the *Program* tab.

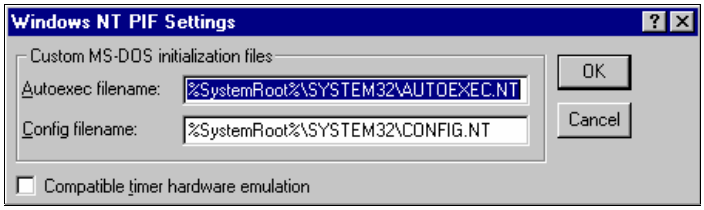



Your application may require some command line switches, which must be added at the end of the *Cmd line* field. In our example, the test4.exe program requires a node address to operate.

5. Append command line switches in the *Cmd line* field.

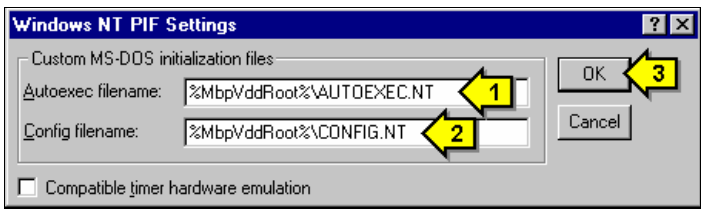



6. Click the *Windows NT* button. You will see the following screen.




By default, when a new shortcut is created, Windows NT selects the autoexec.nt and config.nt files located in the system32 directory. To use private copies of these files, you must modify the directory paths shown in this dialog. In our example, we will use the autoexec.nt and config.nt files located in the product installation directory.

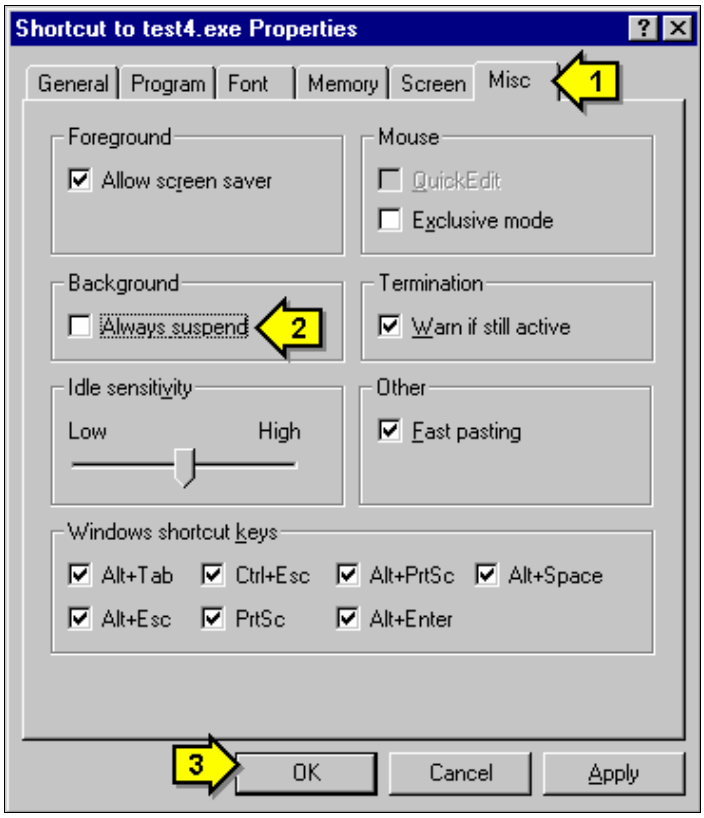
7. Enter *%MbpVddRoot%\AUTOEXEC.NT* in the *Autoexec filename* field. Then Enter *%MbpVddRoot%\CONFIG.NT* in the *Config filename* field. Click the *OK* button.




8. By default, when a new shortcut is created, your application will suspend when the focus is moved away from the application's window. If your application is constantly communicating to some devices,

you will most likely prefer that it runs continuously, even when the focus is moved away. To do that you must deselect the *Always suspend* check box in the *Misc* tab.

Select *Misc* tab. Deselect the *Always suspend* check box. Click the *OK* button to complete the shortcut configuration. You can move your shortcut to another location by dragging it with a mouse and dropping it at the new location.




# Typical Driver Configuration for Win16 Applications

This section shows a typical configuration session. Use it only as a guideline. Only the most common features are shown here. For detailed descriptions, refer to the Virtual MBX Driver Configuration Editor section.

16-bit Windows applications always use the config.nt and autoexec.nt files located in the system32 directory.

To run the Virtual MBX Driver with your Win16 application, the global config.nt file must contain the following line:

*device=%SystemRoot%\System\mbp16.sys*

Also, the global autoexec.nt must contain the following line:

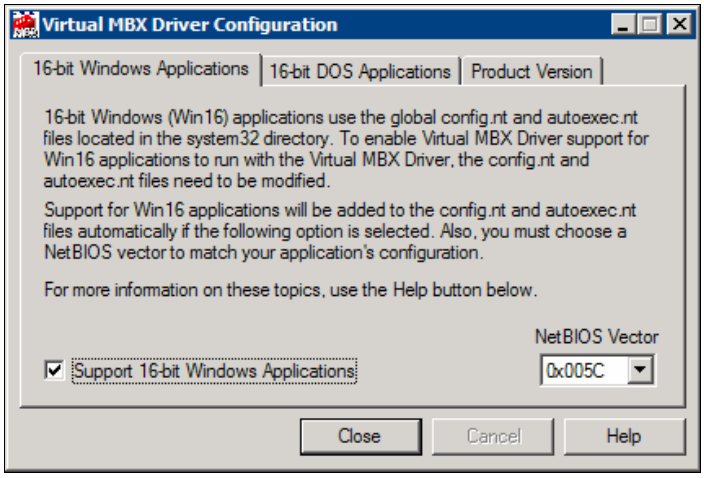
*%SystemRoot%\System\mbp16vec {NetBIOS_vector}*

The NetBIOS_vector identifies the software interrupt vector used by the NetBIOS interface. If not specified, the default value, 5C hex, is used. When your application starts, these lines will automatically load the Virtual MBX Driver.

The above modifications to the autoexec.nt and config.nt files can be done with a text editor, such as Notepad. However, it is much easier to do it by using the Virtual MBX Configuration Editor. The following shows a typical configuration session when this editor is used.

1. From the Windows Start menu, locate the MBX Virtual Driver submenu and select the *Virtual MBX Configuration* menu item.

   Select the *16-bit Windows Applications* tab.




2. Check the *Support 16-bit Windows Applications* checkbox. Then, select the *NetBIOS Vector* to match your application's configuration, normally vector *0x005C*. You must reboot the system to ensure that the added lines will take effect.

# Virtual MBX Driver Configuration Editor

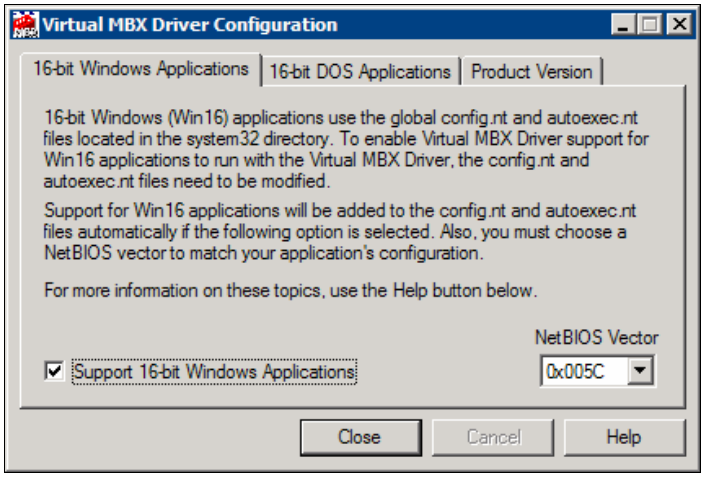
This section provides detailed information about the Virtual MBX Driver Configuration and the Virtual MBX Configuration editor.

If you need a quick-start guide or a step-by-step tutorial of the most common features, go back to the Typical Driver Configuration for DOS Applications or Typical Driver Configuration for Win16 Applications sections.

The Virtual MBX Driver Configuration editor consists of three tabs: 16-bit Windows Applications Tab, 16-bit DOS Applications Tab and Product Version Tab.

## *16-bit Windows Applications Tab*

16-bit Windows (Win16) applications use the global config.nt and autoexec.nt files located in the system32 directory. To enable Win16 applications to run with the Virtual MBX Driver, the config.nt and autoexec.nt files must be modified. This tab can modify these files to add or remove support for Win16 applications.




### *Support 16-bit Windows Applications*

Check this box to add Virtual MBX Driver support for 16-bit Windows Applications.

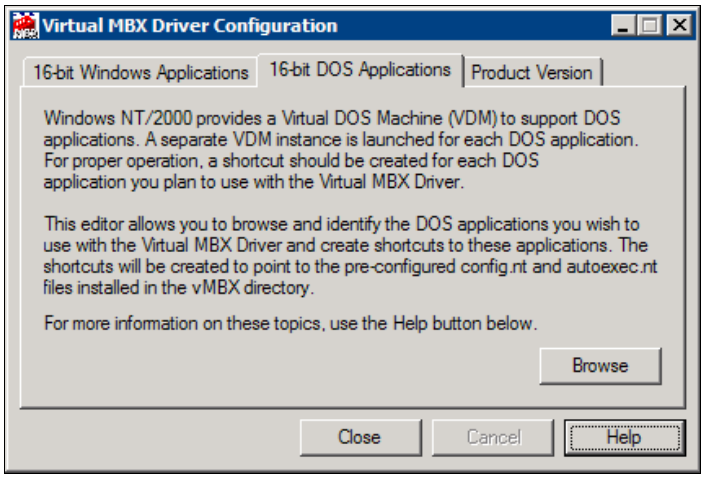
### *NetBIOS Vector*

Select the NetBIOS vector to match your application's configuration, typically vector 0x005C.

# *16-bit DOS Applications Tab*

Windows XP/2000/NT provides a Virtual DOS Machine (VDM) to support DOS applications. A separate VDM instance launches for each DOS application. For proper operation, a shortcut should be created for each DOS application used with the Virtual MBX Driver.

This tab allows you to browse and identify the DOS applications that you wish to use with the Virtual MBX Driver and create shortcuts to these applications. The shortcuts will point to the pre-configured config.nt and autoexec.nt files installed in the vMBX directory.



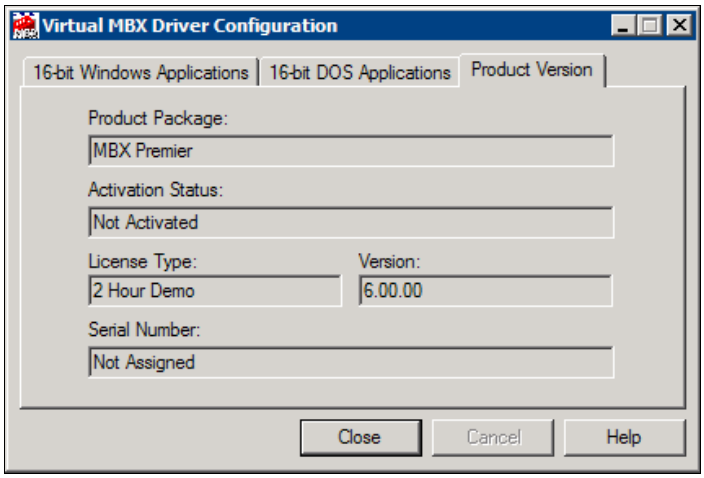

### *Browse*

Click this button to browse to DOS applications. Then click the *Open* button to create a shortcut.

# *Product Version Tab*

This tab shows the Virtual MBX Driver information installed on your system. This information, including the version number, may be requested if you call for technical support.

This tab also tells you if the Virtual MBX Driver has been activated or if it is running in demo mode.



# VALIDATION & TROUBLESHOOTING

The Virtual MBX Driver Setup program installs example shortcuts to the Mbpstat and Modsoft utilities. Use these shortcuts to confirm that the Virtual MBX Driver works. Verify that a 32-bit driver, for example, the MBX Driver, MBX Gateway Driver or Ethernet MBX Driver, is installed and properly configured. Choose various selections from the menu and verify that you can communicate successfully without any errors.

If you are having difficulties communicating through the Virtual MBX Driver the following FAQ section can help you determine the nature of the problem.

## Frequently Asked Questions

***When I double-click on the Modsoft or MBPSTAT icons provided with the Virtual MBX Driver, these programs fail to communicate over Modbus Plus.***

Virtual MBX Driver requires that at least one of the 32-bit driver products (for example, the MBX Driver, MBX Gateway Driver or Ethernet MBX Driver) is installed and properly configured. Verify that your 32-bit driver is operating successfully. Refer to the Validation section for your 32-bit driver product for more information.

***I start my DOS application from the DOS prompt, but it fails to communicate over Modbus Plus.***

To run your application from the command prompt, you must use the *Command Prompt with vMBX* icon that is located in the Virtual MBX Driver group created by the installation program. Refer to the Typical Driver Configuration for DOS Applications section for more information.

***My 16-bit Windows application fails to communicate over Modbus Plus.***

16-bit Windows applications always use the config.nt and autoexec.nt files located in the system32 directory. Verify that these files are properly configured. Refer to the Typical Driver Configuration for Win16 Applications section for more information.