EXPC-1319 Windows Embedded Standard 7 User Manual

First Edition, July 2013

www.moxa.com/product

EXPC-1319 Windows Embedded Standard 7 User Manual

Copyright Notice

Trademarks

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of Moxa.

Moxa provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Moxa reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Moxa assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Technical Support Contact Information

www.moxa.com/support

Moxa Americas
Toll-free: 1-888-669-2872
Tel: +1-714-528-6777
Fax: +1-714-528-6778

Moxa China (Shanghai office)
Toll-free: 800-820-5036
Tel: +86-21-5258-9955
Fax: +86-21-5258-5505

Moxa Europe
Tel: +49-89-3 70 03 99-0
Fax: +49-89-3 70 03 99-99

Moxa Asia-Pacific
Tel: +886-2-8919-1230
Fax: +886-2-8919-1231

Moxa India
Tel: +91-80-4172-9088
Fax: +91-80-4132-1045

Table of Contents

1

Introduction

Thank you for buying Moxa's EXPC-1319-STS panel computer. It comes with the Windows 7 Embedded software platform, providing a simple and familiar development environment for various industrial applications.

- **Windows Embedded Standard 7 OS Components**

Windows Embedded Standard 7 OS Components

Refer to the following content for the software components of the Windows Embedded Standard 7 pre-installed on the EXPC-1319-STS computes.

Core OS:
- 32-bit support
- Remote Client
- Remote Procedure Call

Applications and Services Development:
- .Net Framework 3.5
- Remote Desktop Protocol 7.1
- COM OLE Application Support
- COM+ Application Support
- MSMQ

Internet Services:
- Internet Explorer 8.0
- IIS 7.0

File Systems and Data Store:
- Windows Data Access Components
- Windows Backup and Restore

Diagnostics:
- Common Diagnostic Tools
- Problem Reports and Solutions

Fonts: Chinese (Trad. and Simp.), Japanese, Korean, Western, Middle Eastern, South East Asian, and South Asian Fonts

Graphics and Multimedia:
- MPEG DTV-DVD Audio Decoder (MPEG-2, AAC)
- MPEG Layer-3 Audio Codecs(MP3)
- MPEG4 Decoders
- Windows Media Video VC-1 (WMV) Codecs
- DirectX and Windows Device Experience
- Windows Media Player 12

International:
- IME Simplified Chinese Support
- IME Traditional Chinese Support
- IME Japanese Support
- IME Korean Support

Management:
- Group Policy Management
- Windows Management Instrument (WMI)
- Windows Update

Networking:
- Extensible Authentication Protocol (EAP)
- Internet Authentication Service
- Telnet Server
- Bluetooth
- Domain Services
- Network Access Protection
- Network and Sharing Center
- Quality of Service
- Remote Access Service (RAS)
- Telephony API Client
- Windows Firewall
- Wireless Networking

Security:

- Credential Roaming Service
- Credentials and Certificate Management
- Windows Authorization Manager (AZMAN)
- Windows Security Center
- Active Directory Rights Management
- Security Base
- Encrypted File System (EFS)

Embedded Features:

- Enhanced Write Filter (EWF)
- File-Based Write Filter (FBWF)
- Message Box Default Reply
- Registry Filter
- WSDAPI for .NET

Embedded Self-Health Diagnostic Software:

SNMP-based remote scripting layer for monitoring, reporting, and control

2

System Initialization

This chapter describes how to use the initial boot procedure to set up Windows 7 Embedded Standard system user account settings on your EXPC-1319-STS computer.

The following topics are covered in this chapter:

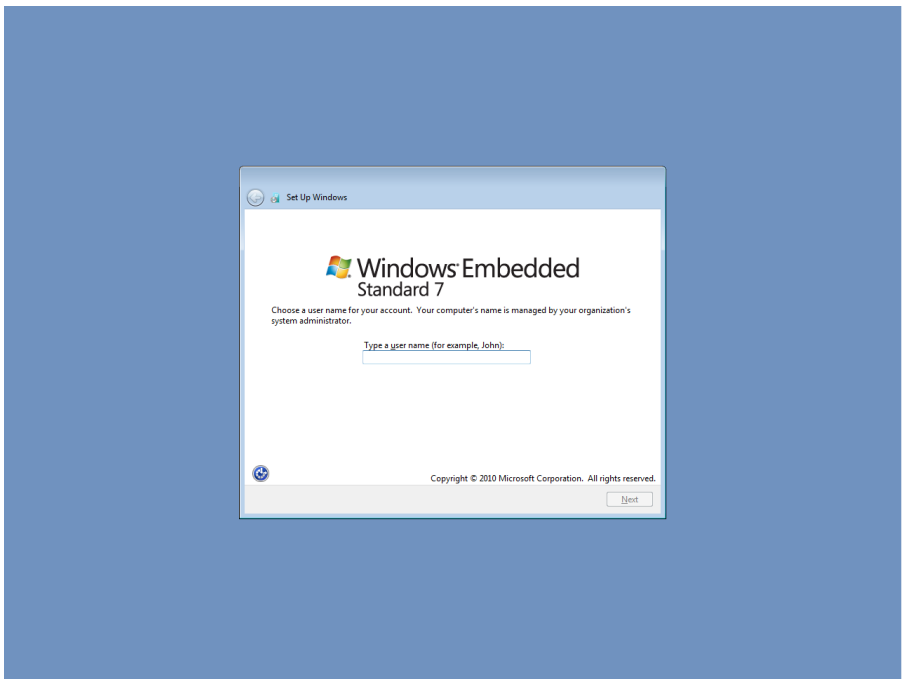
- **Overview**
  - Setting Up a New User Account

Overview

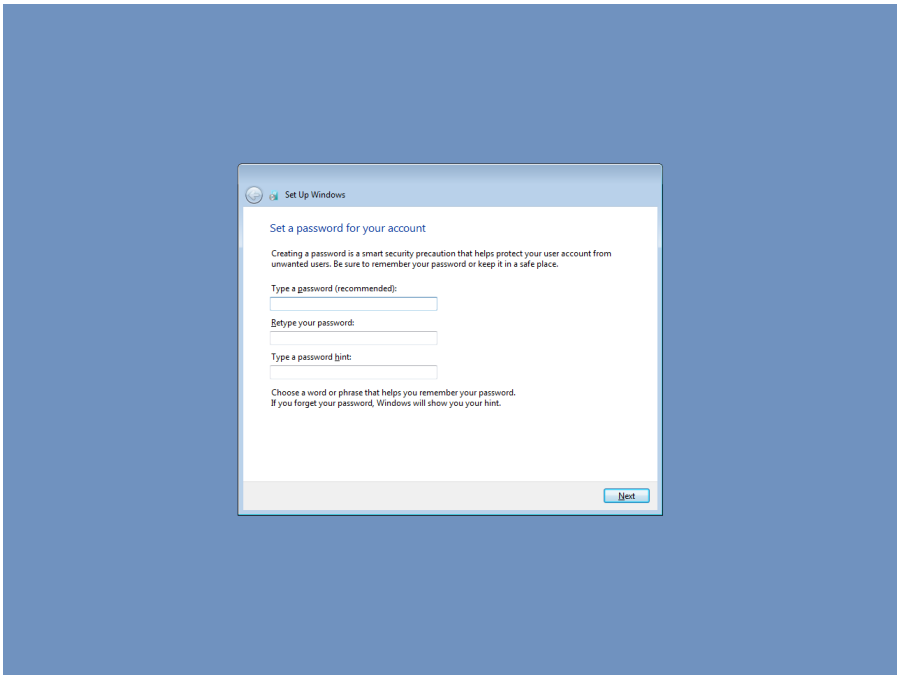
Like most laptop computer, you need to type a user name to create your user account to enable the embedded computer to work, follow the steps below:

Setting Up a New User Account

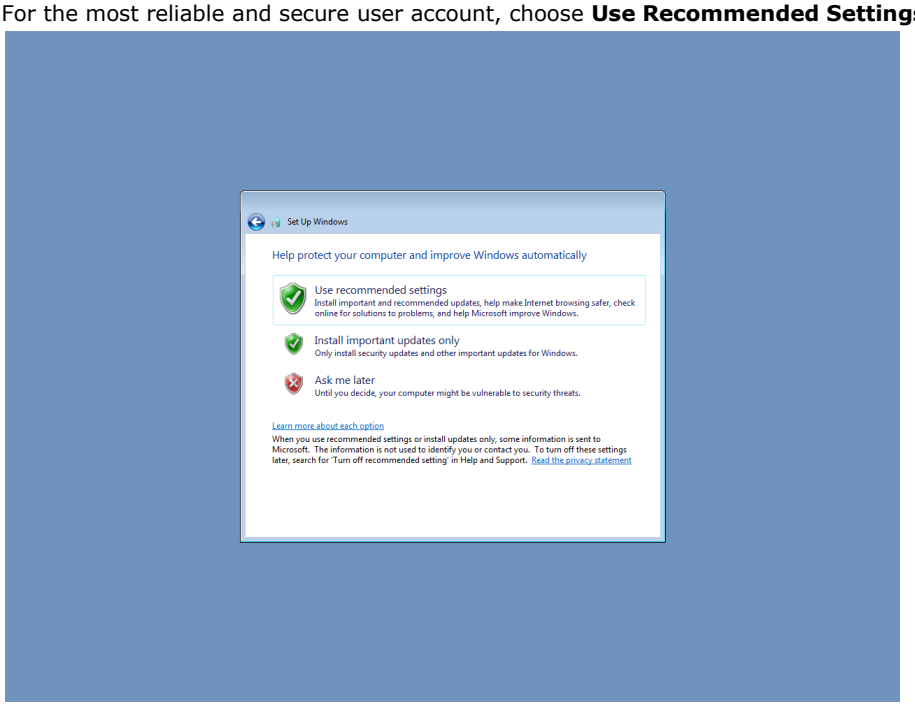
1. After booting up the computer for the first time, a new user account will need to be created. Choose a user name and enter it into the login screen you are presented with immediately following the completion of the boot procedure.



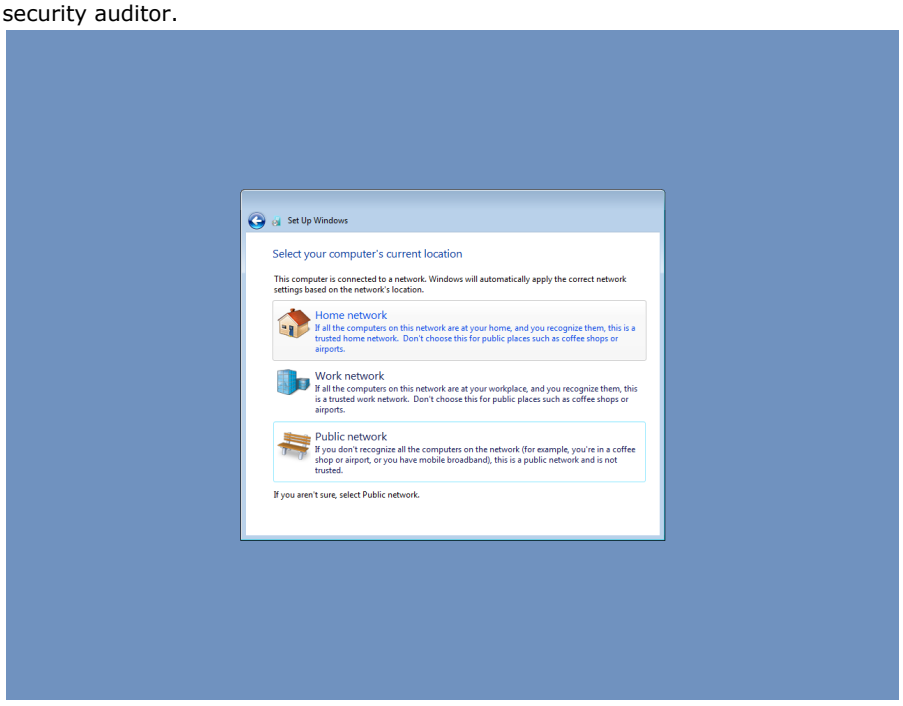
2. Enter a suitably strong password, and then retype the password to verify you have not entered it incorrectly. A password hint may also be entered in the lowest dialog, which Windows will present you in the event you forget your password. If you do not want to set a password, leave it blank and click **Next**.



3. For the most reliable and secure user account, choose **Use Recommended Settings**.



4. Choose a security profile for the computer. The strictest security settings will be applied when choosing **Public Network**; however, some Windows conveniences may be disabled when using this profile. If problems arise with certain applications on the local network, consult your systems admininstrator and/or security auditor.



5. You may now use this user account to operate your EXPC-1319-STS embedded computer.

3

Panel Control Buttons and OSD

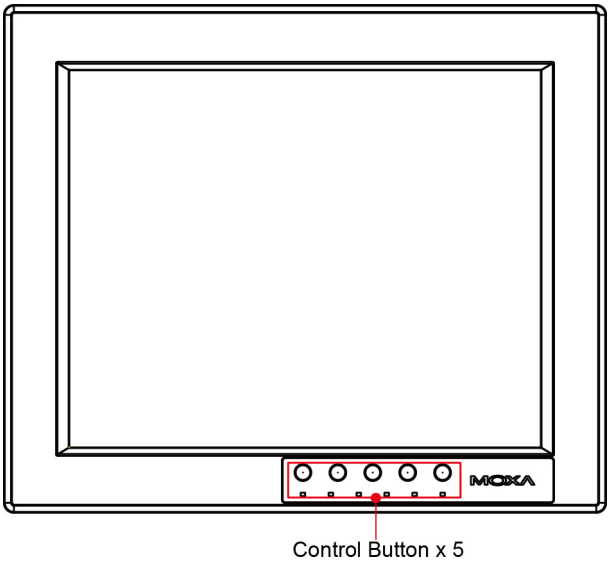
This chapter describes how to use the panel control buttons and OSD (On-screen Display) for the EXPC-1319-STS panel computer.

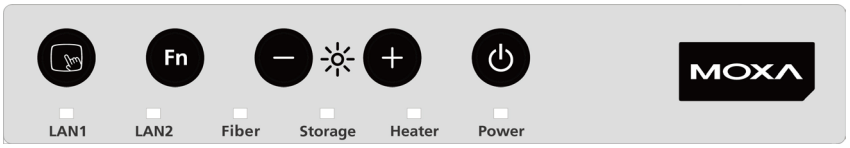
The following topics are covered in this chapter:

- **Overview**
- **Power**
- **Adjusting Brightness**
- **Configuring the Function Key**
- **Enabling and Disabling the Touch Screen**

Overview

There are five control buttons on the bottom of the front panel.





See the following figures and description of the functions of each control button.

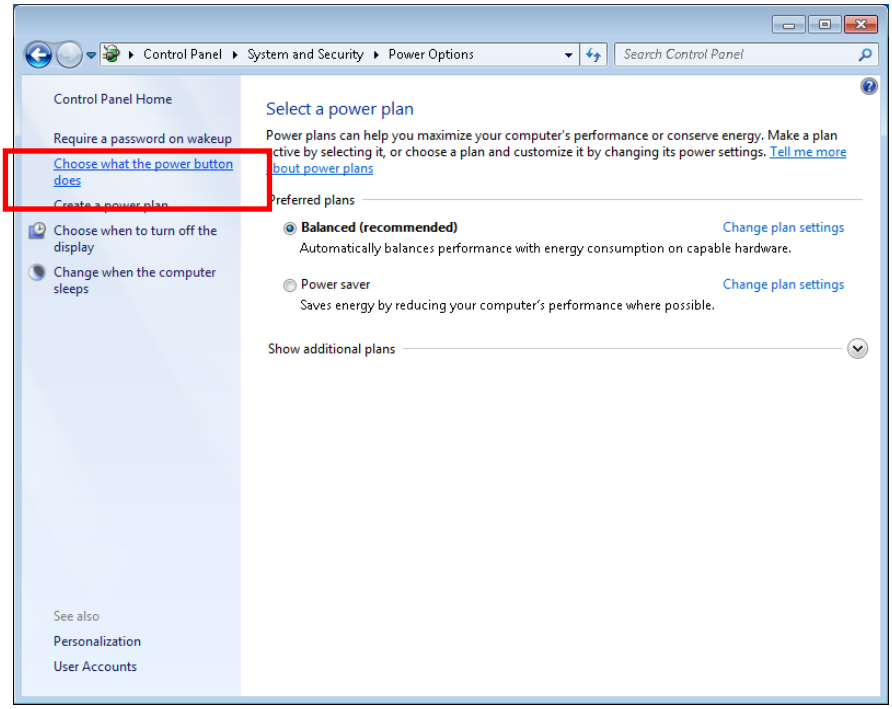
Control Buttons

	Power	Press to turn on the computer. Press again to turn off the computer.
	Brightness	Increase (+) or decrease (-) display brightness.
	Function	This is a customizable function key. By default, this button enables a virtual on-screen keyboard.
	Touch screen	Touch to enable or disable the touch screen. The touch screen is enabled by default. Use this button to turn the touch screen off.

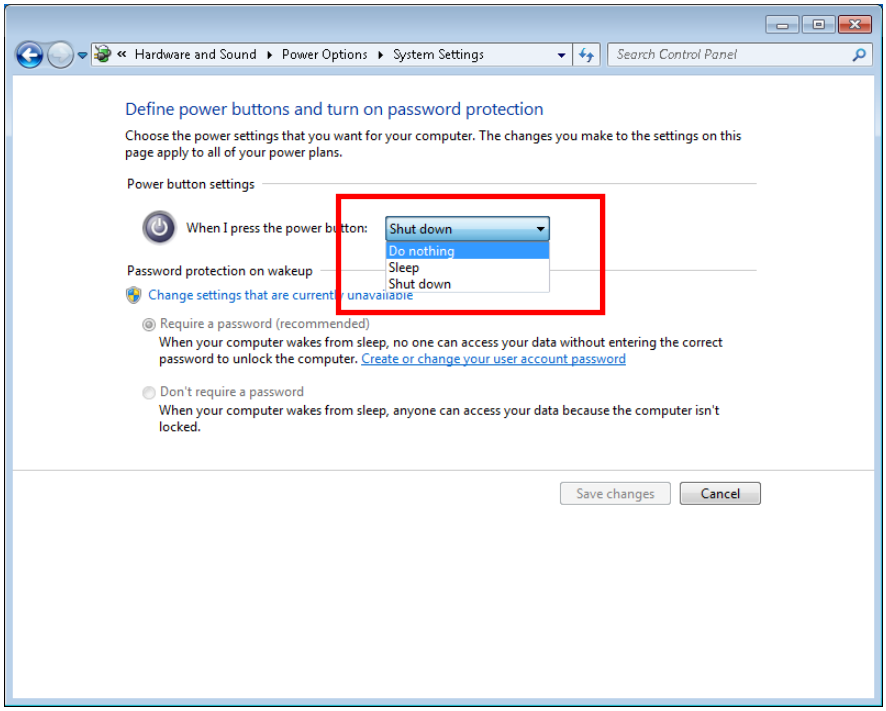
Power

It is possible to use Windows to configure different power profiles. These can help you conserve power usage.

Step 1: Navigate to **Control Panel→System and Security→Power Options,** and click on the selection in the right column, **Choose what the power button does**.

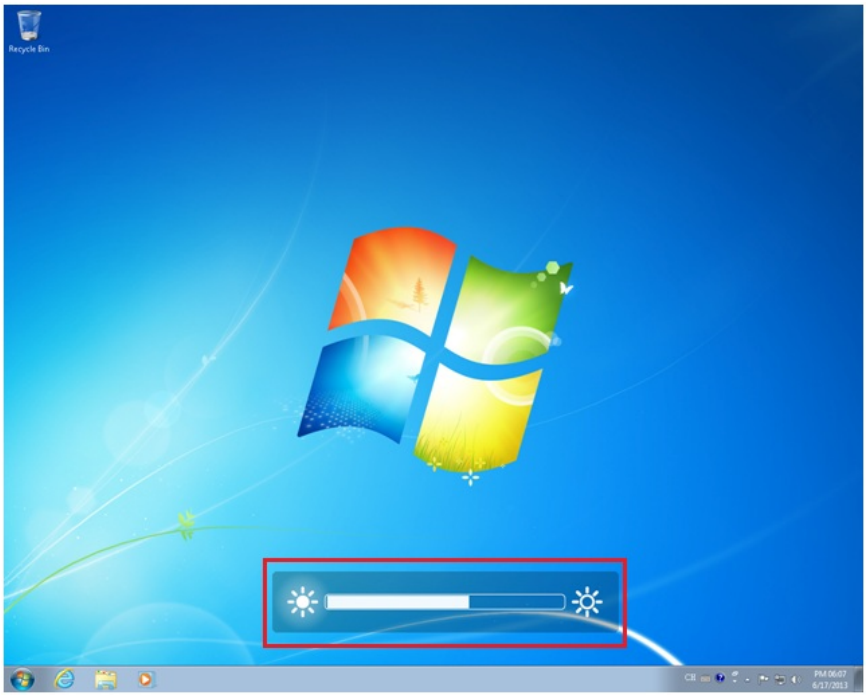


Step 2: You may set the power button to **Do nothing, Sleep,** or **Shut down**, and configure a password that must be used to re-activate the computer from the sleep state.



Adjusting Brightness

You may adjust the brightness using the **+** and **–** buttons. When pressing the button, an adjustment scale will display in the lower middle of the display. Please note that when in the **DirectDraw Full Screen Mode**, this scale will not display.
(Refer to http://en.wikipedia.org/wiki/DirectDraw for detailed description for DirectDraw.)

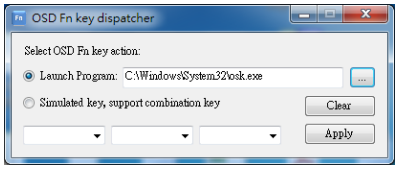


Configuring the Function Key

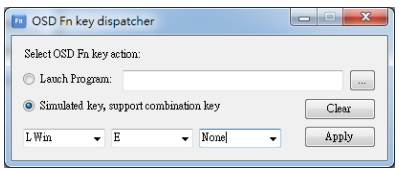
The factory default for the **Function (Fn)** key is the launch of a virtual on-screen keyboard.

However, you may also configure the **Function** key to enable a program or simulate a function with a combination of several keys. Follow these steps:

Step 1: To launch the **Function** key configuration program, select the **mxFnKey** icon from the **System Tray** on the desktop taskbar.



Step 2: To associate a program with the **Fn** key, select **Launch Program** and enter the path of the program you would like to trigger, or browse the directory tree and select the path from **Windows Explorer**. When finished, click **Apply**.



Step 3: To simulate a series of key strokes, select **Simulate key, support combination key** and then configure the keystrokes you wish to associate with the Fn key from the drop-down lists to configure the functions you want to trigger. For example:

L Win + **E** + **None** will launch **Windows Explorer**. When finished, click **Apply**.

To entirely disable the function key, click **Clear**.

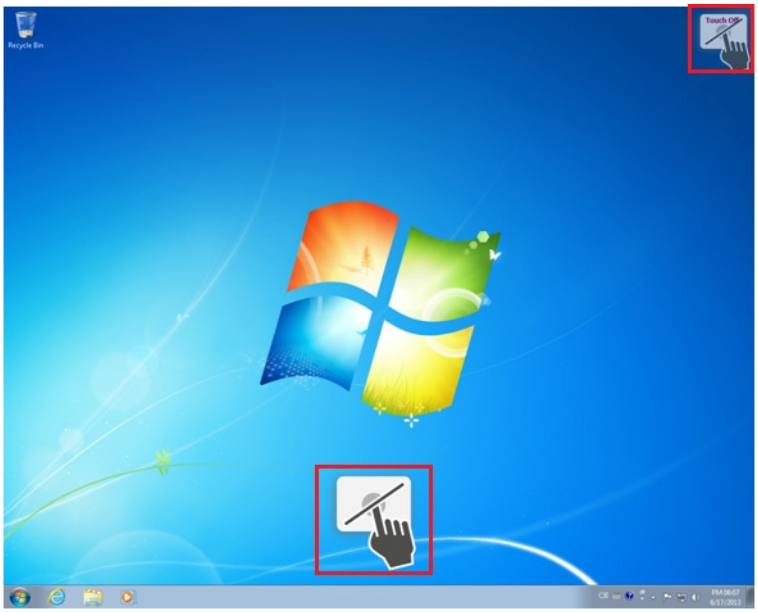
For detailed descriptions of available hot key combinations, you may refer to the following links:

http://technet.microsoft.com/en-us/magazine/ee851673.aspx

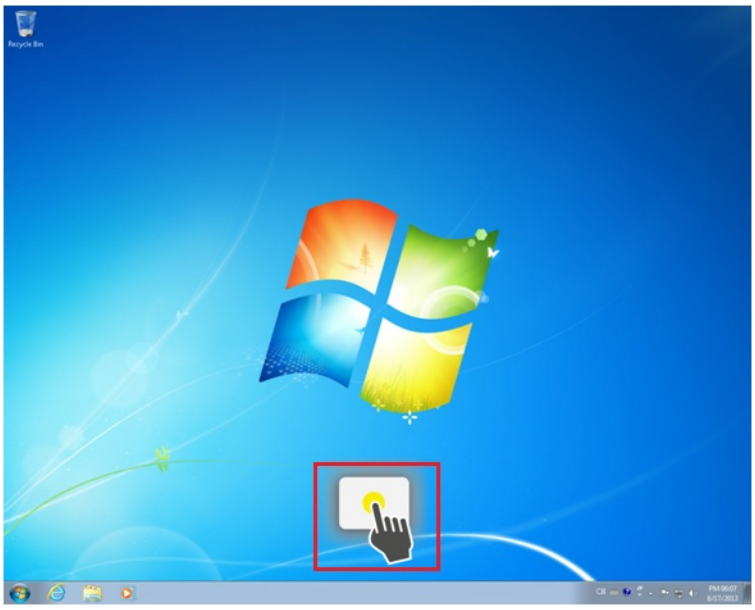
http://en.wikipedia.org/wiki/Table_of_keyboard_shortcuts

Enabling and Disabling the Touch Screen

You may enable or disable the touch screen by depressing the touch screen button. When the touch screen is disabled an icon will continuously display in the upper right corner of the screen, while another icon will always display for about 1.5 second in the lower middle of the screen.



To enable the touch screen function, simply depress the touch screen button again, and an icon indicating the touch screen has been enabled (shown below) will display for about 1.5 seconds in the lower middle of the screen.



4

Touch screen Calibration

This chapter describes how to calibrate the touchscreen function.

The following topics are covered in this chapter:

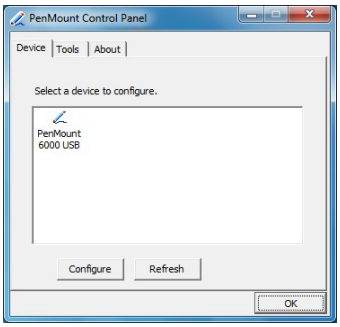
- **Calibrating the Touch Screen**
  - Standard Calibration
  - Advanced Calibration
- **PenMount Calibration Utility Parameters**
  - Turn off EEPROM Storage
  - Touch Screen Cursor Settings
  - Edge Compensation

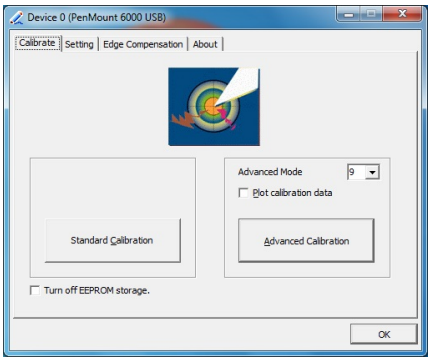
Calibrating the Touch Screen

This chapter describes the calibration process for the EXPC-1319 touch panel.

First, Open the PenMount control panel. This may be found under the **Windows 7 Start Menu**, in the **Programs** list in the **PenMount Windows Universal Driver(WHQL)** folder. From the PenMount folder, navigate to the **Utility** folder and open the **PenMount Control Panel.**

Next, the **PenMount Control Panel** should appear as in the screen shot to the right, with the **Device** tab as its default display. Double-click on the device you want to calibrate, or select the device and click **Configure**. If you do not see your device offered on the menu, click **Refresh** to refresh the list.

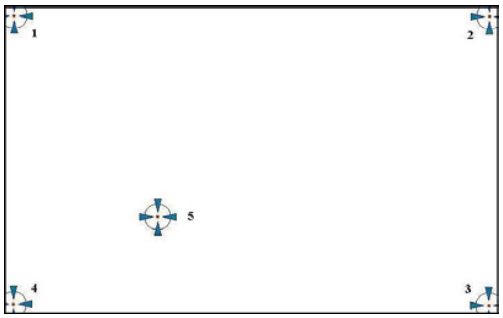




The final preparatory step is to choose what sort of calibration you want. Most will choose **Standard Calibration**, which is a basic touch screen calibration using five reference points. For most situations, a standard calibration should be adequate. As the touch screen ages, users will find that the standard calibration is not adequate for re-establishing screen accuracy and precision. If problems are still encountered following a standard calibration, you may choose **Advanced Calibration** to calibrate the touch screen to a greater number of reference points.

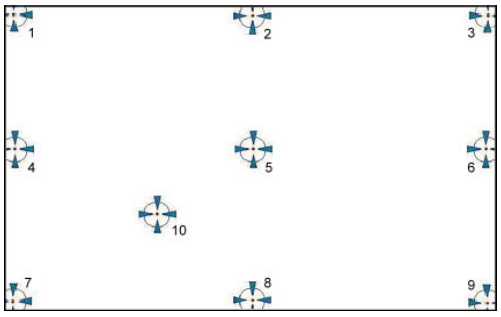
Standard Calibration

For a standard five point calibration, five spots will appear one after another on the display. Use your finger or stylus to touch the five points in order. After you have completed the sequence, hit **ESC** on your keyboard to save the result and exit the calibration process.



Advanced Calibration

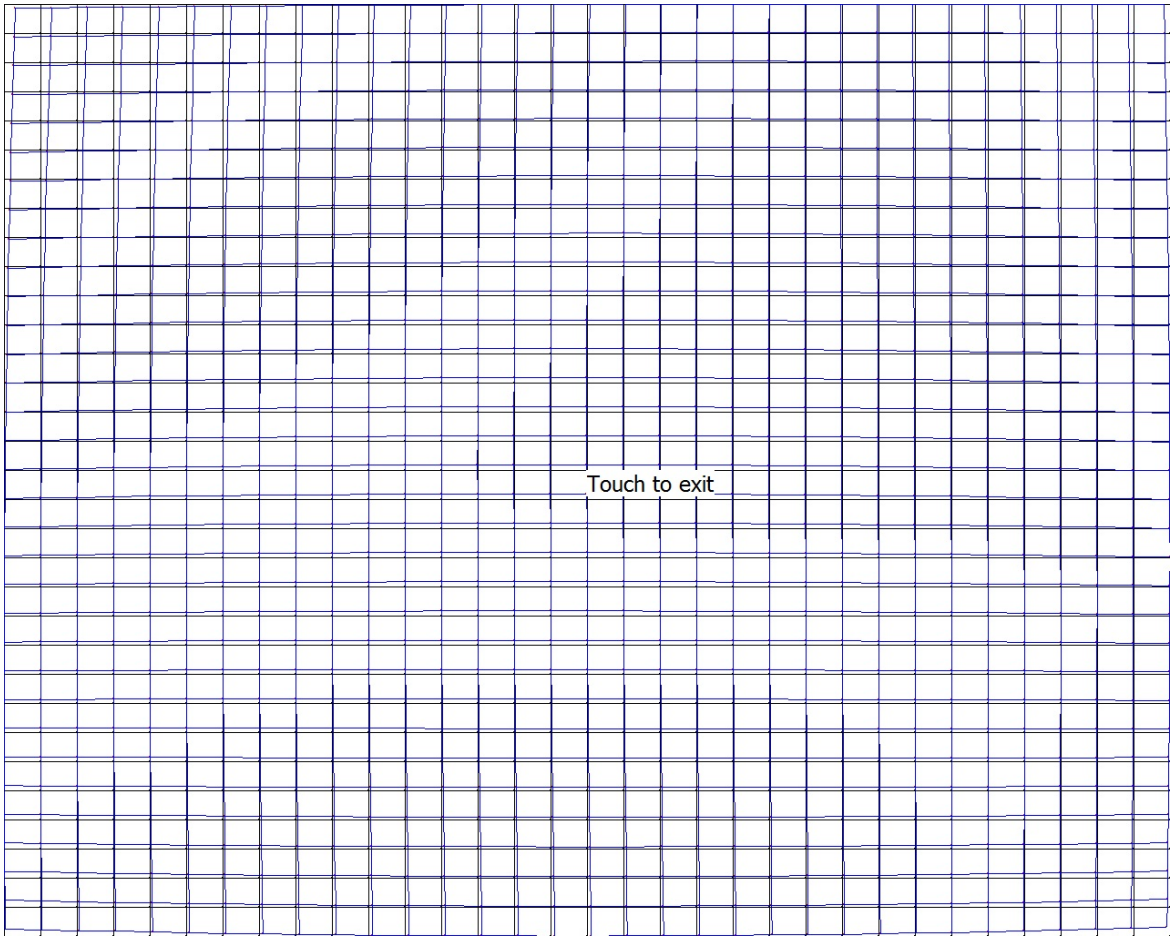
An advanced calibration uses 9, 16, or 25 points to calibrate touch panel linearity; select the number of reference points from the drop-down menu offered on the calibration utility main dialog. You may also instruct the calibration utility to plot detailed calibration data onto a graph. For more information about the data graph, see the next section, **Calibration Data Graph**.



Just as with the standard calibration, to complete the calibration use your finger or stylus to touch the points in order, as they appear. After you have completed the sequence, hit **ESC** on your keyboard to save the result and exit the calibration process.

Advanced Calibration: Calibration Data Graph

If you performed an advanced calibration and ticked the **Plot Calibration Data** selection, then after you complete an advanced calibration the calibration utility will provide you with a graph comparing ideal panel linearity as assumed by the PenMount utility (the black lines) plotted against the approximate linearity derived by the PenMount utility from the user calibration process (the blue lines).



Please note that this function is mainly used by the panel manufacturer for troubleshooting. To exit the graph, simply touch the screen. If you feel you have discovered problems with calibration that you cannot solve using the PenMount calibration utility, please contact Moxa's Embedded Computing Technical Support staff.

PenMount Calibration Utility Parameters

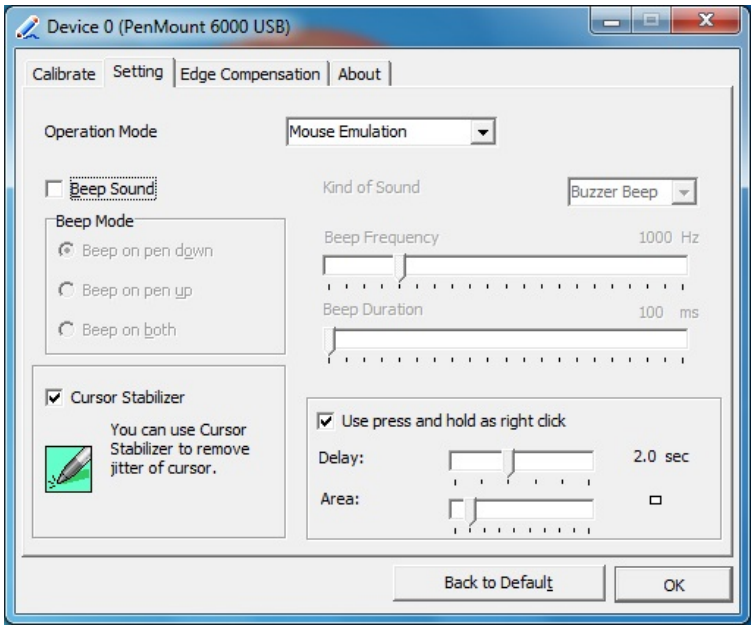
Turn off EEPROM Storage

Ticking this box disables the storage of calibration data in the permanent EEPROM screen controller; instead, the calibration data is saved to the system drive. If you turn off EEPROM storage, the value will be stored and available from one restart to the next, but the changes will be lost should you perform a system software recovery, forcing you perform a touch panel recalibration.

Touch Screen Cursor Settings

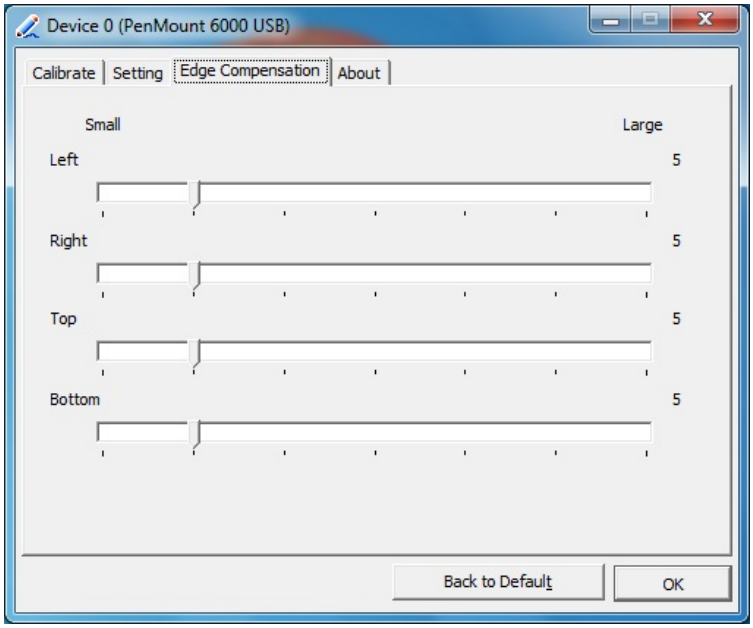
The **Settings** tab allows for configuration of four main touch screen cursor features: cursor behavior (mouse emulation or stylus mode), a beep that sounds when contact with the touch screen is made or broken, a cursor stabilizer, and press-and-hold in place of right clicking.
The top drop-down may only be set to **mouse emulation**. No other modes are available.

Beep mode allows you to configure a beeping sound to play whenever contact is made (or broken) with the screen. The beep may be configured for tone, frequency, and duration.
The **cursor stabilizer** removes jitter from the cursor when the computer is being used in high vibration environments.
To enable **right-click capability** for the touch screen, users may enable the **press-and-hold-as-right-click**, which allows users to press on the cursor and hold their finger in place, without moving, to call up the right-click menu available in most Windows applications.
Back to defaults resets all of the touch screen interface settings to their factory defaults.
When finished, click **OK**.



Edge Compensation

This page allows users to calibrate the touch screen so that software features at the edges of the display are easier to access. This is often a serious problem when, for instance, users are touching the screen with fingertips that are too thick to conveniently access scroll bars, or to manipulate objects on the Windows task bar, or in the system tray located on the bottom of the screen. The edge compensation interface consists of four sliders one for each edge of the screen. The far right represents the largest possible edge area, while the far left represents the smallest possible (unmagnified) edge area.



5

Configuring Serial Interface

This chapter describes how to configure the serial interfaces of the EXPC-1319 panel computer.

The following topics are covered in this chapter:

- **Overview**
- **Configuring the Serial Interfaces**

Overview

The EXPC-1319-STS features two software-selectable serial ports that support three different serial interfaces: **RS232, RS485 (2-wire)** and **RS422/RS485 (4-wire)**. The device handles for the serial ports are COM1 and COM2.

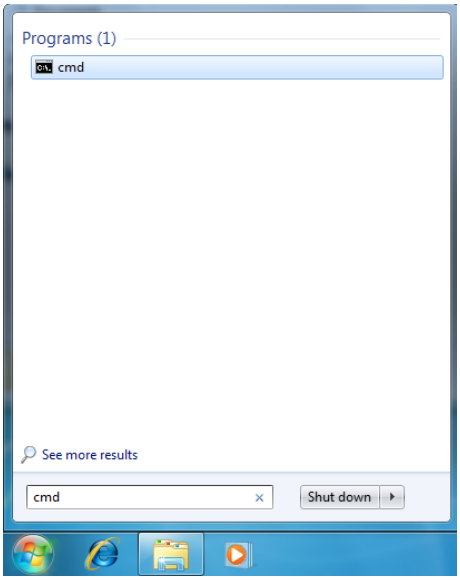
Please see the following notes for these serial interfaces:

1. COM1 and COM2 support baudrate up to 115200 bps, but 38400 or less is recommended, as the FIFO will be overrun when throughput is high.
2. However, you can still see COM3 and COM4 in Device Manager. Please note that these ports are reserved, do not use.

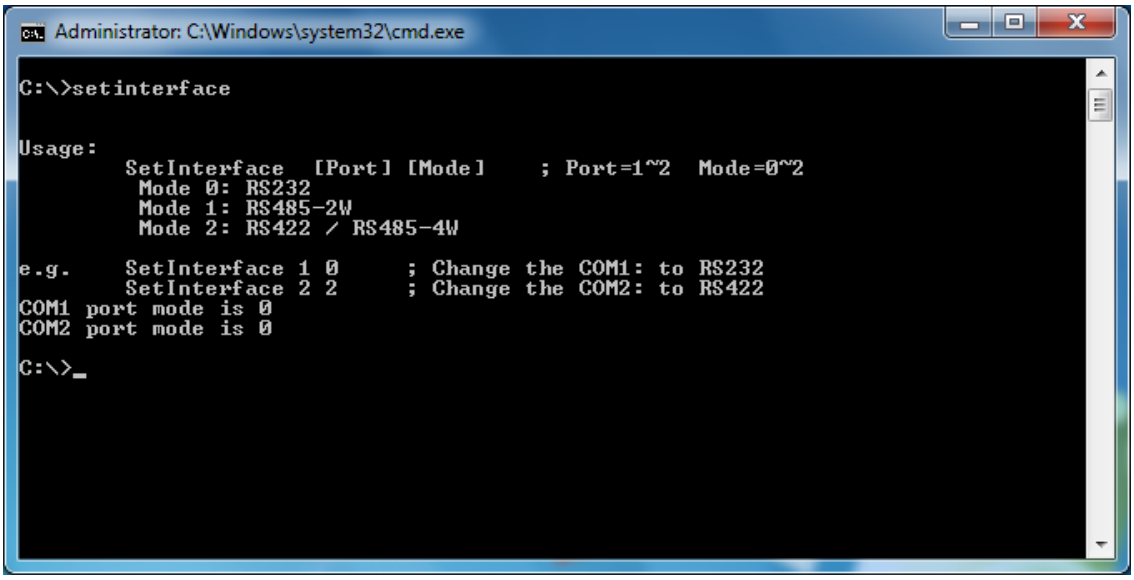
Configuring the Serial Interfaces

Follow these steps to change serial interface mode.

1. Open command console by running **cmd.exe**.



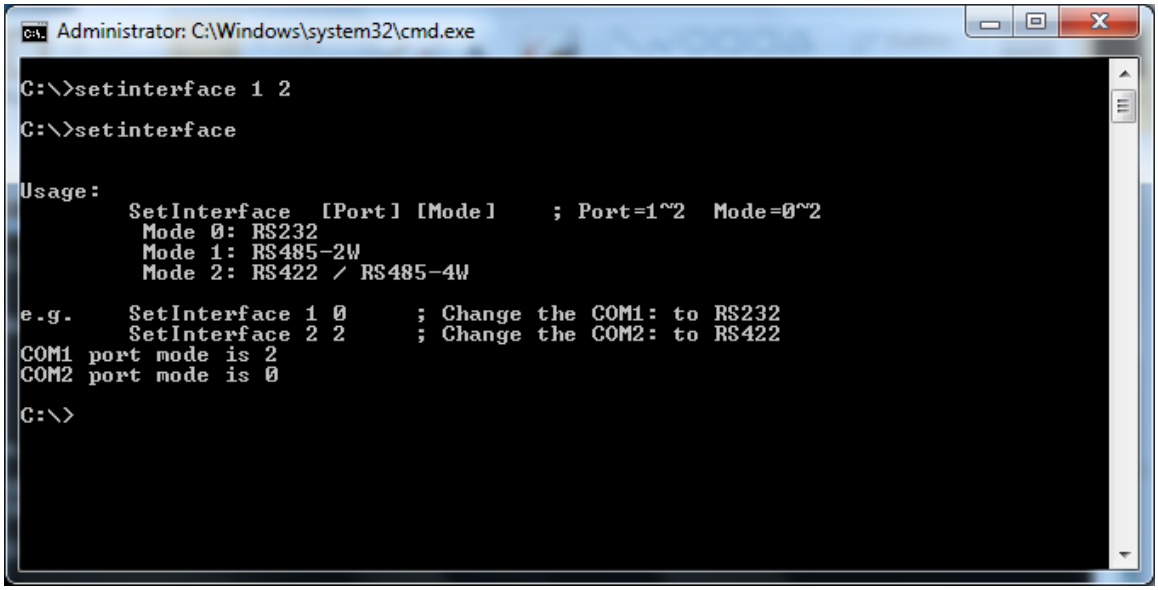
2. To verify what the current serial interface is that the port is set for, type **SetInterface** with no additional arguments. This will return the current interface for which the physical port has been set.



3. To change the a COM port to a different serial interface, type **SetInterface [Port] [Mode]**, where **[PORT]** is either **1** (for **COM1**) or 2 (for **COM2**), and **[Mode]** is either **0** (for **RS-232**), **1** (for **RS-485 2-wire**), or **2** (for **RS-422 / RS-485 4-wire**). For example typing:

   ```
   C:\> SetInterface 1 2
   ```

   will change serial port 1 (i.e. COM1) as a **RS422/RS485-4-wire** interface.



6

Enabling Embedded Filters

This chapter describes how to operate the embedded enabling features on the EXPC-1319-STS panel computer.

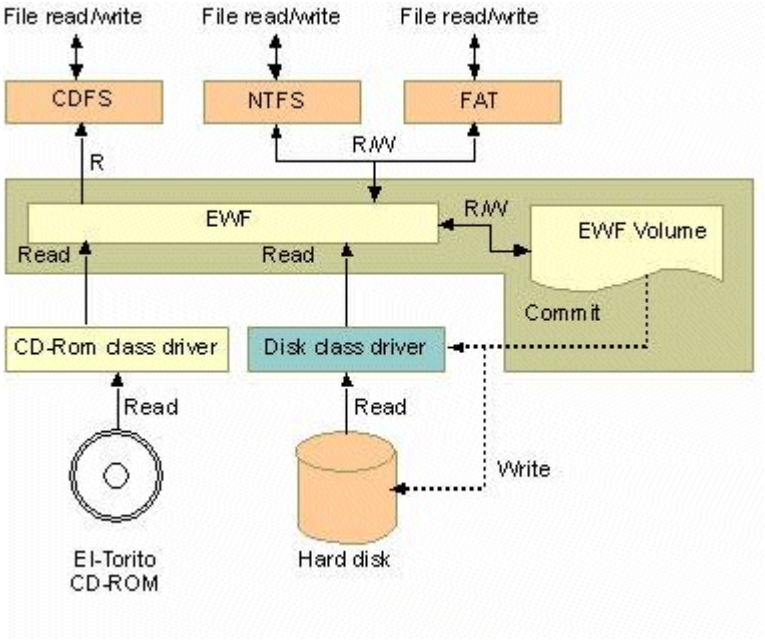
The following topics are covered in this chapter:

- **Enhanced Writer Filter**
- **File-Based Write Filter**

Enhanced Write Filter

Overview

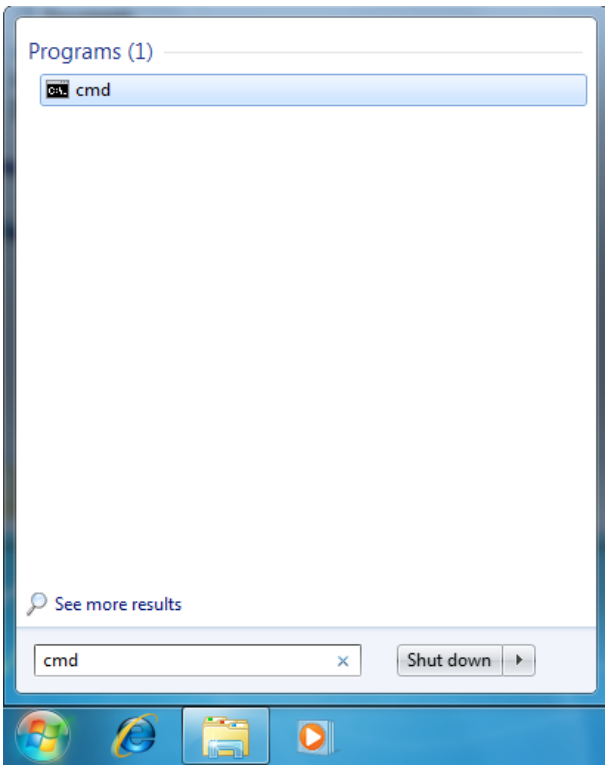
Enhanced Write Filter (EWF) provides a means for protecting a volume from writes. This allows the operating system (OS) to boot from write-protected hard disks. All written data to an EWF-protected volume (The Hard disk in the following figure) are redirected to an overlay (EWF Volume in the following figure). Because EWF does not write data to hard disk directly, it can protect the hard disk from sudden power loose. These written data are cached in the overlay and made available as part of the volume. This gives the appearance that the volume is writeable. The overlay is an independent storage location which exists in random access memory (RAM). If desired, the data stored in the overlay may be committed to the protected volume. Refer to the following figure for the overview of the EWF structure.



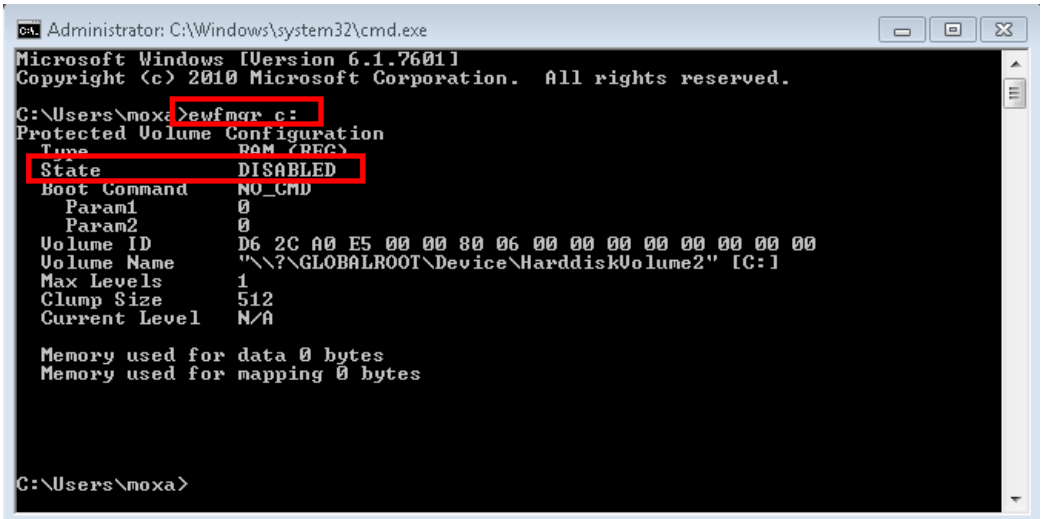
Enabling Enhanced Write Filter

Follow these steps to enable the Enhanced Write Filter

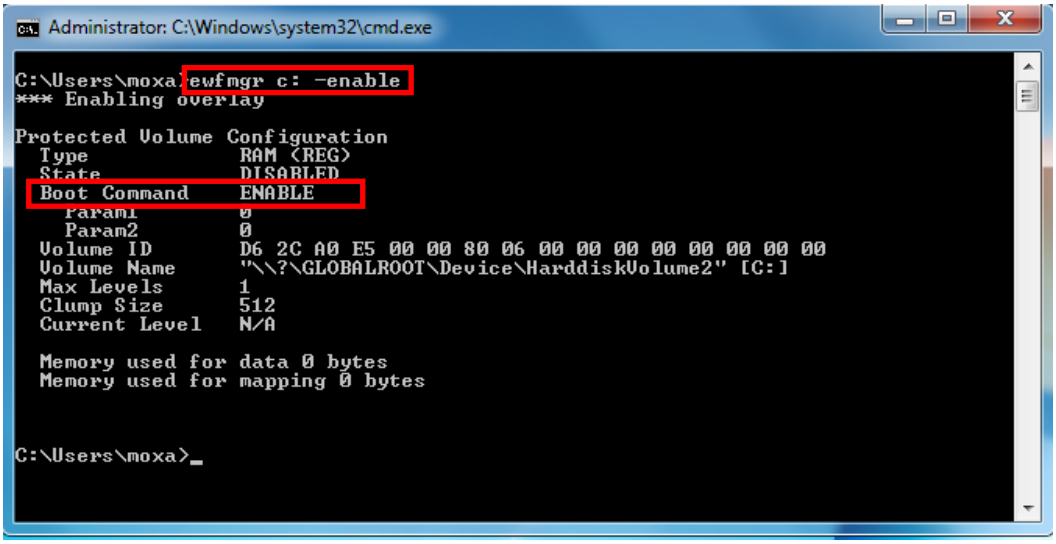
1. First open command console by running **cmd.exe**.



2. To verify that Enhanced Write Filter is disabled, type **ewfmgr c**:.

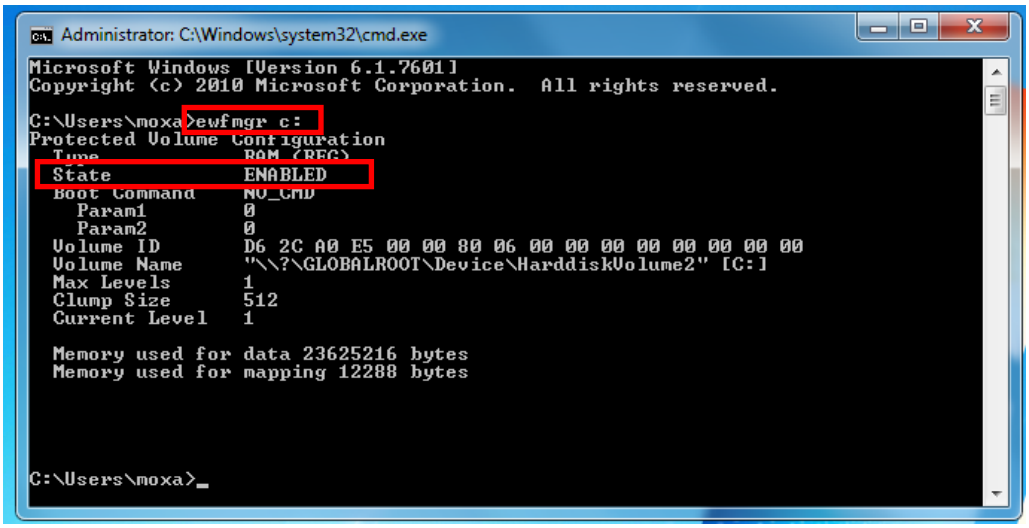


3. To enable the Enhanced Write Filter, type **ewfmgr c: -enable.**

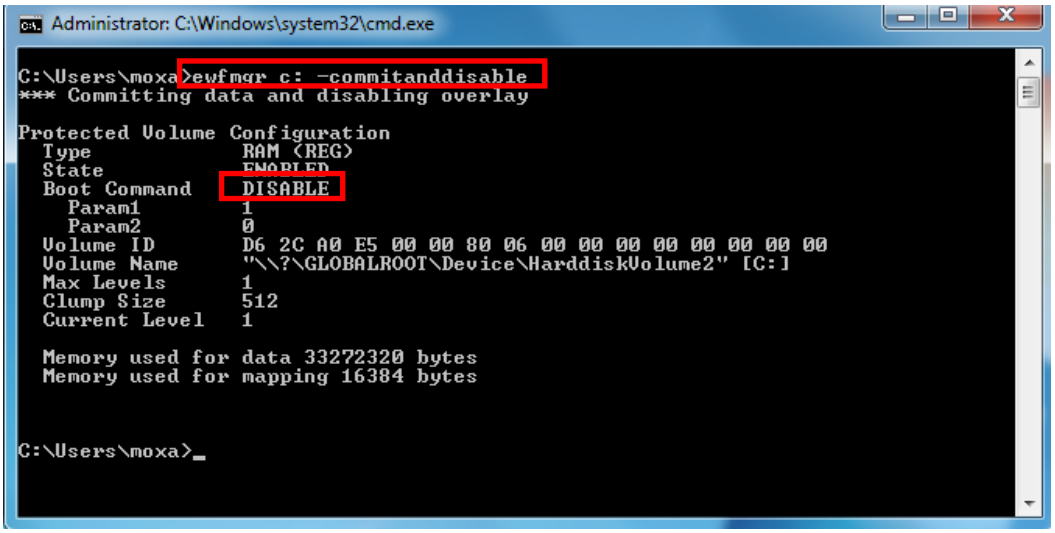


4. Reboot the system to take effect.

5. To verify again that Enhanced Write Filter is enabled, type **ewfmgr c**:.



6. To disable the Enhanced Write Filter, type **ewfmgr c: -commitanddisable**.



For the EWF commands, refer to the MSDN web site:

http://msdn.microsoft.com/en-us/library/ms940853%28v=winembedded.5%29.aspx

File-Based Write Filter

Overview

According to Microsoft:

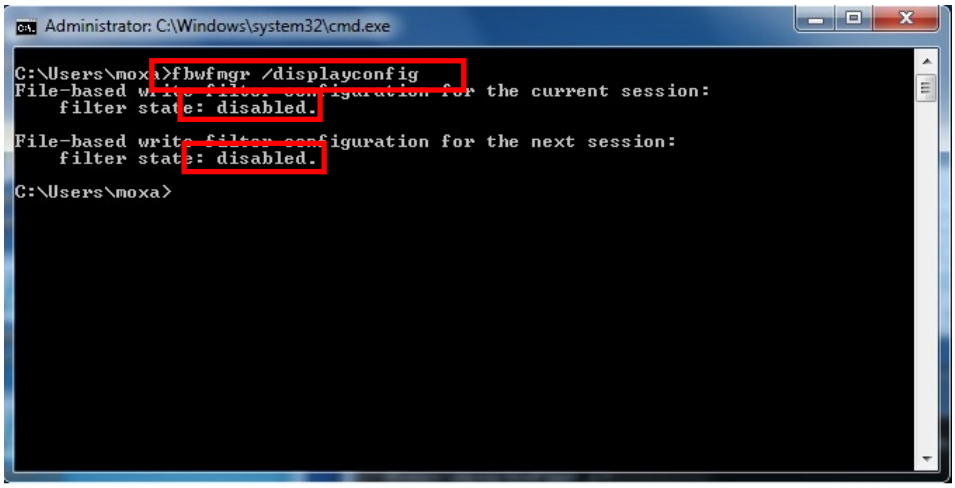
> *File-Based Write Filter (FBWF) allows the Windows Embedded platform to maintain the appearance of read and write access on write-sensitive or read-only storage. FBWF makes read and write access transparent to applications.*
> *Writing to storage media may be undesirable or impossible in embedded devices. FBWF redirects all writes targeted for protected volumes to a RAM cache called an overlay. Used in this context, an overlay is similar to a transparency overlay on an overhead projector. Any change made to the overlay affects the picture as seen in the aggregate, but if the overlay is removed, the underlying picture remains unchanged.*

FBWF provides the advanced feature than EWF to let user specify the directory to write the data to disk drive directly, in our default setting, the default directory is under c:\temp, which means you can read/write the data into disk without commit action.

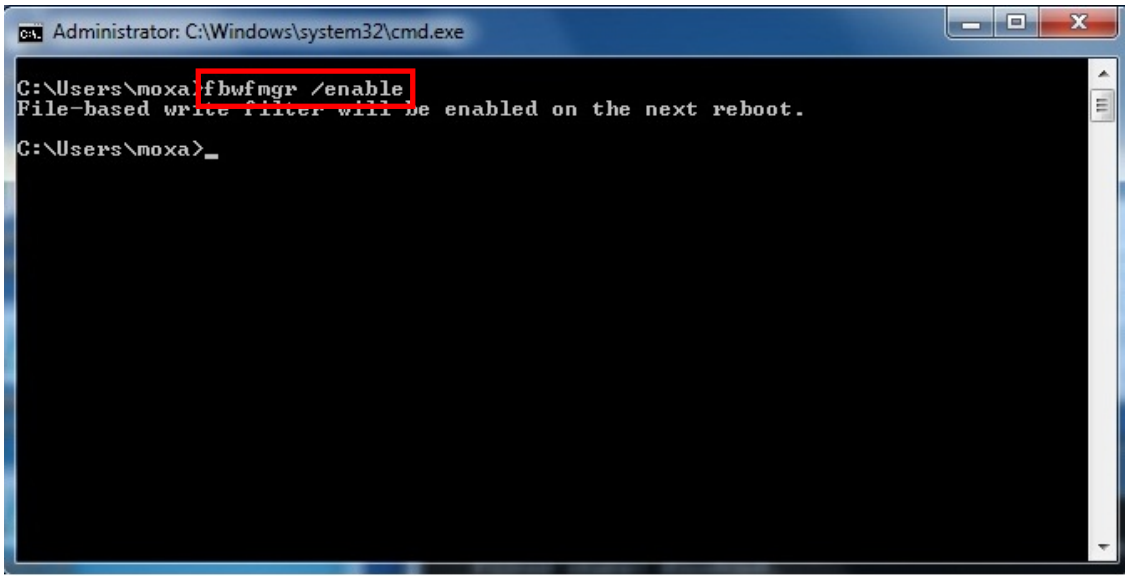
Enabling File-Based Write Filter

To enable file-based write filtering, do the following:

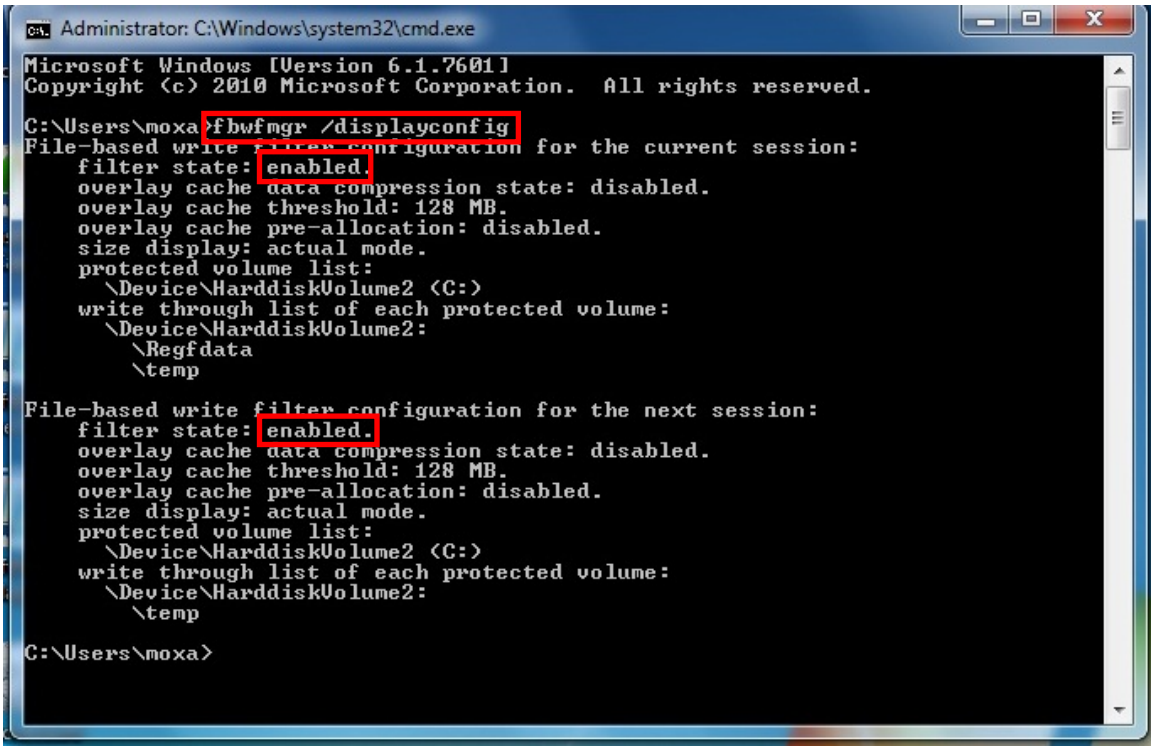
1. To verify that Enhanced Write Filter is disabled, type **fbwfmgr /displayconfig** to check the current status.



2. Type **fbwfmgr /enable** to enable the FBWF, and then reboot the system to take effect.



3. When system reboots, under command prompt, type **fbwfmgr /displayconfig** again to check if the status has been changed to "enabled".



7

Moxa Software Package

This chapter describes the software package for users to easily control and monitor the EXPC-1319-STS computers.

The following topics are covered in this chapter:

- **The Synmap™ Virtualization Layer: Full Software Interoperability with Any Moxa Device**
  - Overview
  - The Synmap Design Concept
  - Moxa Synmap OIDs
- **Installing the Synmap Virtualization Layer**
- **Installing and Using an NMS**
  - Installing Moxa MxView
  - Basic Configuration of MxView
  - Loading the Synmap MIB File
  - Using Synmap OIDs to Control the EXPC-1319
  - Using Synmap to Read the Voltage Sensor
- **Using the Host Resources MIB**
  - Checking CPU Load Using the Host Resources MIB
  - Checking Data Storage Stats Using the Host Resources MIB
  - Checking Network Status

The Synmap™ Virtualization Layer: Full Software Interoperability with Any Moxa Device

Overview

Synmap™ is Moxa's revolutionary software virtualization, an evolutionary advance in network device control that adapts solid, reliable SNMP into a fully portable remote procedure interface. Synmap allows engineers to automate remote processes using SNMP object identifiers (OIDs) rather than device- or OS-specific API addressing, making a scripted Synmap procedure fully interoperable with any other Synmap device. This means that a script created for one Synmap device may be directly copied to another, immediately conferring the same functionality. This eliminates the need for rewriting and compiling code for newly configured devices, significantly reducing maintenance and deployment times.

SNMP is lightweight and easy-to-configure, and is already long-popular with IT professionals; it also enjoys comprehensive native support in high-level languages like .NET, Java, Python, or Ruby. For these reasons, the Synmap framework has re-imagined SNMP as a universal configuration and control interface for remote procedures, adapting it to not only monitor and control device internals like temperature, BIOS parameters, and local interfaces, but also to report on and automate tasks at the process layer, as well. Easily integrated into any existing Network Management System (NMS), Synmap devices are a flexible and cost-effective upgrade that returns obvious benefits to any IA network.

Synmap currently allows you to use SNMP for remote monitoring and control of a select set of computer processes, but its list of features is rapidly growing. Using Synmap's fully portable scripts, engineers will soon be able to:

- Access, monitor, control, and report on digital I/O at both the process and hardware layers
- Use OIDs to monitor, configure, and give process control over serial ports and other interfaces
- Monitor and control system attributes and process events via any NMS
- Build automated remote procedures using Synmap OIDs called by simple shell scripts, or a preferred high-level language like Python, Perl, or VBScript—all without any need for low-level APIs, or platform-specific libraries
- Significantly simplify and reduce development times for custom utilities and automated executables
- Gain scripting and automation independence from OS-dependent libraries

All of this may be achieved using simple, reliable, and familiar SNMP, the easily accessible standard that IT engineers are already familiar with.

The Synmap Design Concept

Synmap is a software design concept that offers programmers a wholly unique and superior conception of infrastructure development for IA control. Instead of using low level APIs, Synmap adapts the higher level SNMP protocol to serve as a universal API across all machines. With Synmap, application developers gain several benefits, the two biggest being a significantly reduced learning curve for control APIs and remarkable code portability. For example, if a user wants to control GPIO in a Linux environment, an application developer needs to generate code that follows the pseudo code shown below:

1. Open() the device node
2. Read() the file descriptor
3. Read() the return value, and make a logical decision
4. Perform an ioctl function on the file descriptor
5. Close() the file descriptor

The above example shows how this is done in a *NIX environment. In a Windows environment, it looks a little different, but the process is essentially the same, and of equal complexity:

1. Open a required file handle using **mxgpio_open**
2. Get data using the file handle, an assigned port, and **mxgpio_get_data**
3. Evaluate the returned data, and make a logical/control decision
4. Use **mxgpio_set_data** with the file handle to set a value
5. Use **mxgpio_close** to close the file handle

These examples show, in concise form, the difficulties application developers face when dealing with low level APIs. Developers must understand each system's API and track down various device node IDs from within the user manual, the sample code, or the general system. Synmap significantly simplifies this situation. In comparison to the example just shown, the pseudo code that replaces it will look something like this:

- GET an OID using SNMP and the localhost connection (127.0.0.1)
- Evaluate the returned data, make a logical decision
- SET an OID using SNMP and the localhost connection (127.0.0.1)

The benefits of using SNMP in this way should be clear.

- First, the code is easily migrated across different computers and even different operating systems, because Moxa's SNMP libraries are supported on both Windows XPE and Linux, as well as a host of other platforms.
- Second, the program can just as easily be ported to the network for remote operations simply by changing the localhost connection (127.0.0.1) to the target IP address and hostname.
- Third, the time needed to learn how to control a peripheral is drastically cut; all one needs to do is understand how to use an SNMP OID, and start scripting.
- Fourth, Developers are free to choose any kind of programming languages or utilities with which they might be familiar, so long as they are apropos to the platform(s) on which they will be used. For example, in place of the C API, Microsoft developers might want to use the SNMP libraries in .NET or Java to control remote Linux devices, or it can be flipped around so that Linux developers use Net-SNMP libraries to control remote Windows XPE machines.

All of these things mean that the Synmap virtualization makes the work of programming custom applications much faster and simpler, and dramatically increases code interoperability. Complex controls such as USB notify, mounting information, and BIOS settings have been integrated into the Synmap engine, so that creating a customized monitoring or control application now only requires the coordination of a few SNMP SET/GET calls, potentially allowing developers to save on hundreds of lines of code when authoring new applications.

Moxa Synmap OIDs

The full list of SynMap OIDs is reproduced as **Appendix A: The Moxa Synmap OID Table**. The table below lists the Synmap OIDs that are currently enabled on the EXPC-1319 panel computer.

Item Name	OID	Access	Description
productName	1.3.6.1.4.1.8691.17.1.1.1	read-only	Returns the product name
productDesc	1.3.6.1.4.1.8691.17.1.1.2	read-only	Returns a short device description
productVersion	1.3.6.1.4.1.8691.17.1.1.3	read-only	Returns product version
productBuildDate	1.3.6.1.4.1.8691.17.1.1.4	read-only	Returns the last software build date, YYMMDDHH format
voltSensorsIndex	1.3.6.1.4.1.8691.17.1.5.1.2.1.1	read-only	Returns a list of numbers that correspond with the voltage sensors, used by SNMP for identification; begins with 1
voltSensorsDevice	1.3.6.1.4.1.8691.17.1.5.1.2.1.2	read-only	Returns a list of string values identifying the voltage sensors by name/location.

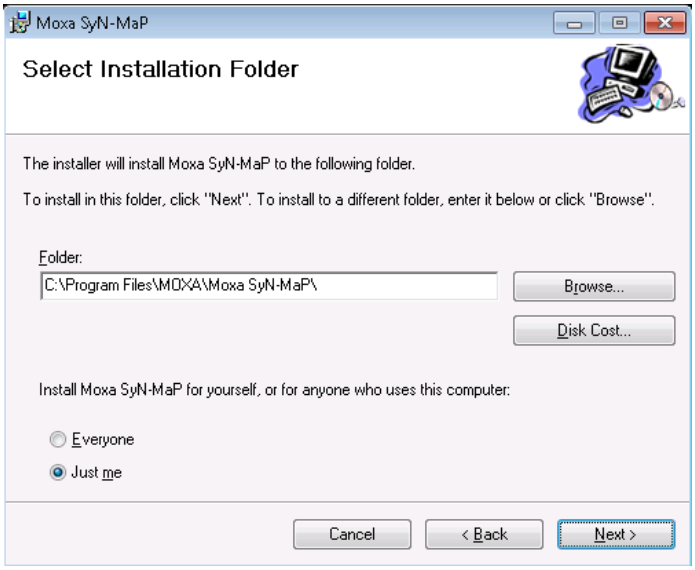
			Possible values are **Vcore**, **V1.05**, **V1.5_S3**, **V1.5**.
voltSensorsValue	1.3.6.1.4.1.8691.17.1.5.1.2.1.3	read-only	Returns/sends a value indicating or changing the sensor's state
usbDeviceProductID	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.3	read-only	Returns the USB's hexadecimal product ID
usbDeviceActiveClass	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.4	read-only	Returns the USB device class for any connected device

Installing the Synmap Virtualization Layer

The following steps will install Synmap.

1. Double click `mxSynmap_setup.msi`, found in the **Utility** folder located on the software DVD under `\utility\3.mxSynmap\`. Then click **Next** to start the Synmap setup wizard.

2. In the middle of the dialog, the button **Disk Cost** will display how much space the Synmap software package will occupy on your storage drive, as well as the remaining storage space on the drive where the system is stored.
   At the bottom of the dialog, select whether Synmap will be installed for every user across the entire system, or just for the current user account. Above that (in the text dialog), you may click the **Browse** button to browse the file tree and select the folder where you want to install the package, or simply click **Next** to install Synmap to the default folder.



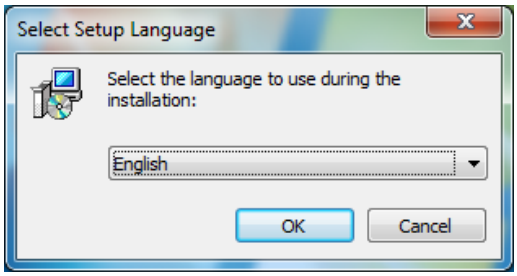
3. Click through the next few dialogs to complete the installation of the Net-SNMP agent. The SNMP agent will not begin working until you reboot the TC-6110 computer.

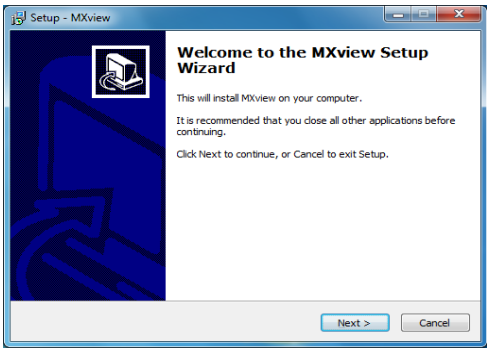
Installing and Using an NMS

For full implementation, Synmap requires (like any SNMP-based system) an NMS to become fully functional; an NMS with an MIB browser also makes using SNMP a far simpler task. If you already have your own MIB browser, you can skip this section. However, if your network is lacking an NMS then you may install a free version of Moxa's **MXview** to get Synmap up and running. MXview provides an MIB browser and an interface that will allow you to monitor and control any Synmap enabled device. This section will walk you through a basic MXview installation, and show you how to use the MXview MIB browser to start working with the TC-6110's MIB.

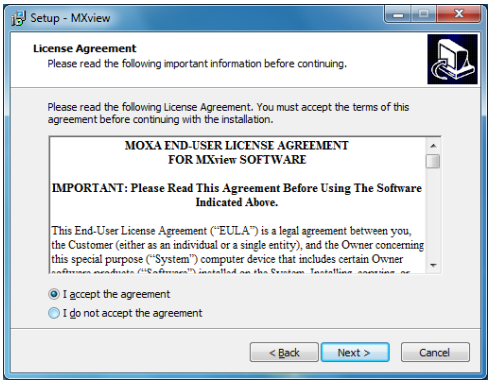
Installing Moxa MxView

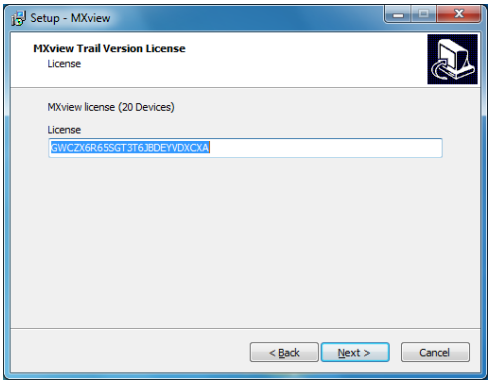
1. MXview is included on your TC-6110 software DVD. Double click **`mxView_Trial_V2.3.msi`** in the Utility folder, which you can find on the software DVD in **`\utility\4.mxViewTrial`**. Select **OK** to choose the language, and when the next dialog appears click **Next** to continue.



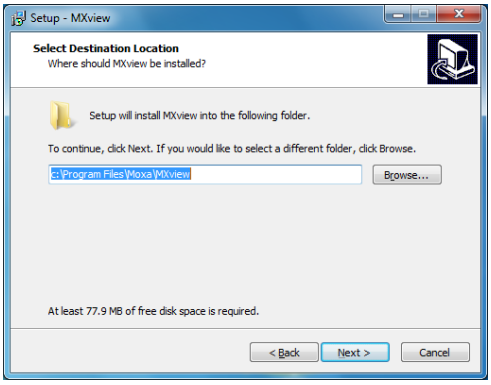


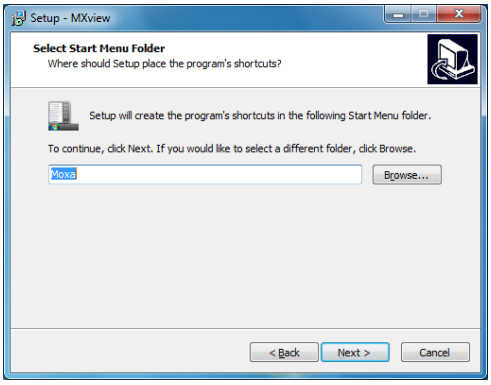
2. Accept the licensing agreement and click **Next** to move to the licensing dialog.



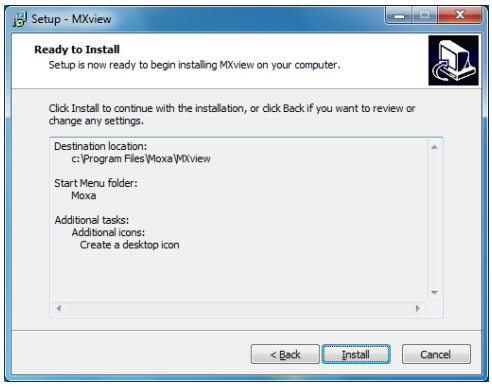


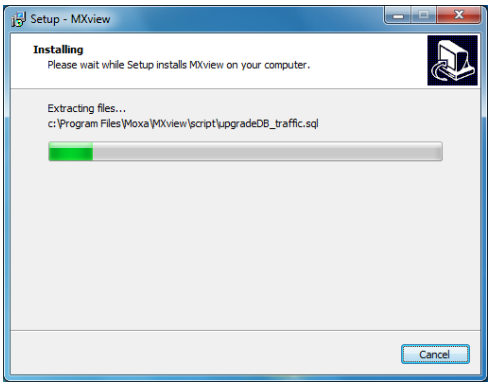
3. On the next dialog you may change the folder and path where MXview will be installed. On the next, you may select where MXView shortcuts will be stored in the Windows Start Menu.



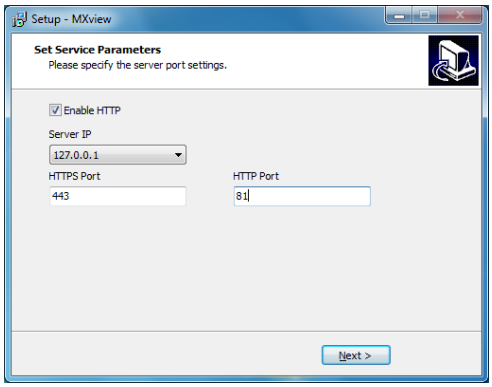


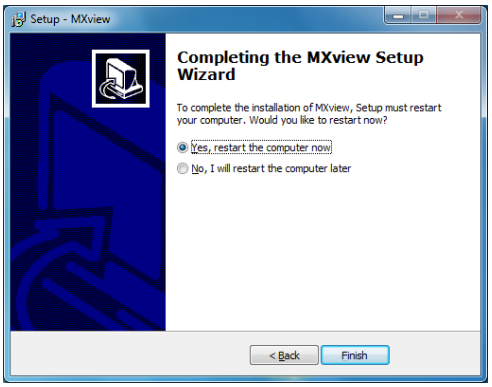
4. Next, you may register MXview as a Windows service and create a **desktop shortcut**.
5. After you have completed the pre-install configuration, click **Install** to transfer MXview to disk and wind up the installation.



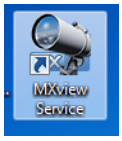


6. After MXview has installed, you must enter the IP address of the machine on which it is located. This may be the **localhost** address, **127.0.0.1**, or if you are connecting to MXview over a LAN it will be a remote IP address. Additionally, you must configure the ports which MXview will use for HTTP and HTTPS communications. Once the installation is complete, you may choose to restart the computer to get MXview up and running.



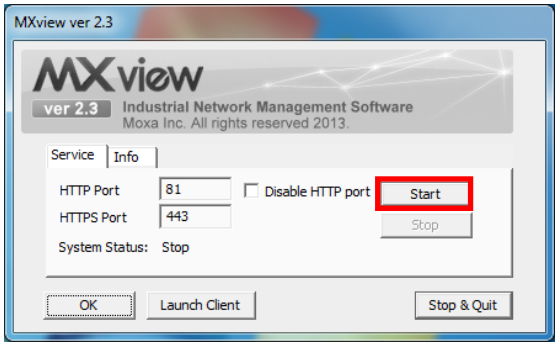


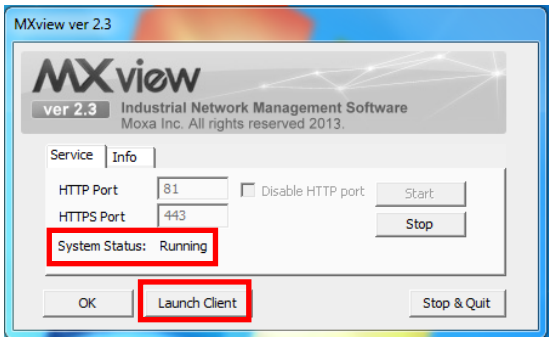
7. After rebooting, the MXview shortcut will appear on your desktop (shown at right). Click on the shortcut to continue on to the next section and begin the MxView setup.



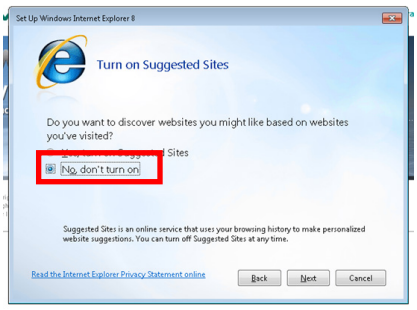
Basic Configuration of MxView

1. Open MXview (see step 7 of the last section, immediately above) and select **Start** to initialize the MXview NMS; wait for the **System Status** notification to change to **Running**, then click **Launch Client**.





2. If opening Microsoft Internet Explorer for the first time, make sure to turn off the **suggested sites** feature (shown at right). If you wish to use another browser you may, and IE's other settings may be configured to your own preferences.



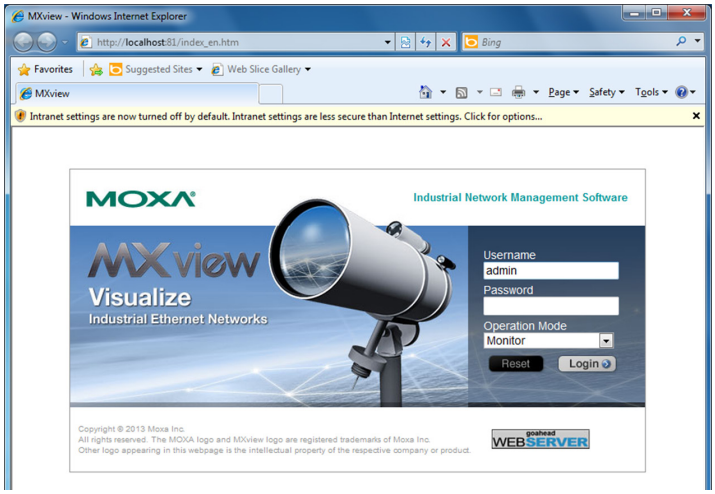
3. The IP address for MXview will be 127.0.0.1 followed by a colon and the HTTP port you have configured MXview to communicate over (in **step 6 of MXview**). If you have used

the suggested settings above, then to login using HTTP would be 127.0.0.1:81, and using HTTPS you would use 127.0.0.1:443.

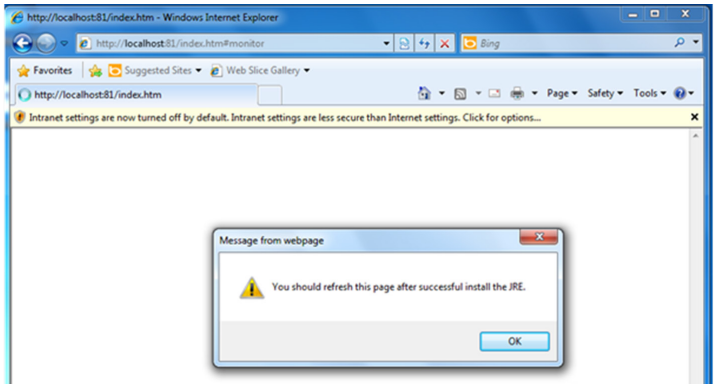
WARNING

For security's sake, Moxa strongly recommends resetting the password to a strongly secure password of at least 8 characters, mixing numbers and symbols in a non-word series.

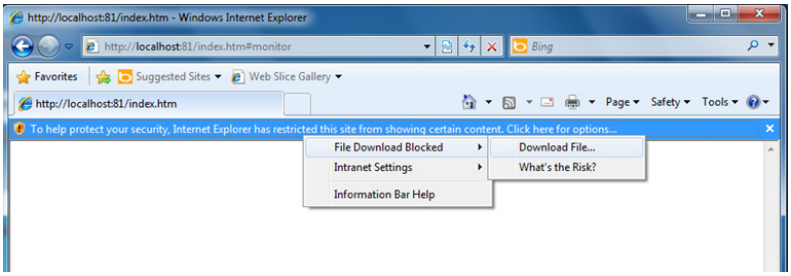
For the login, the default username is **admin**, with a blank password.



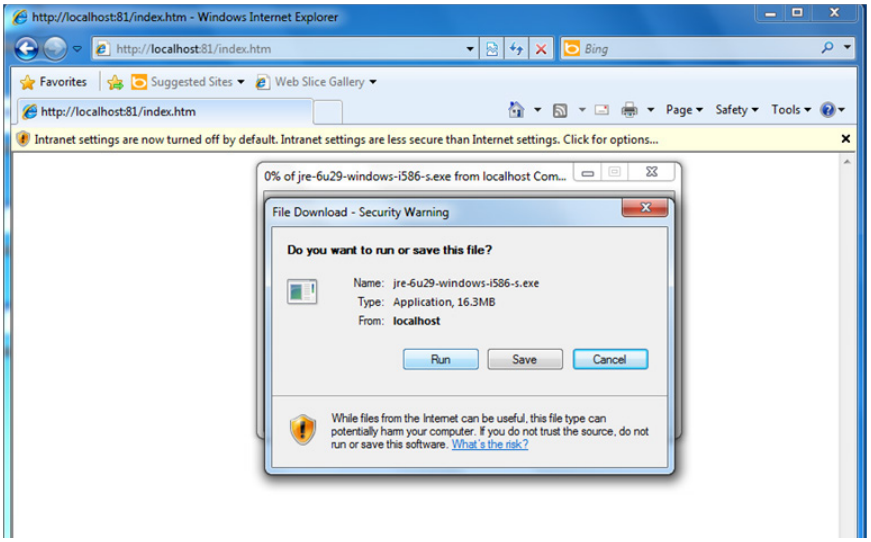
4. When opening your browser for the first time, a warning message will pop up telling you to install the Java runtime environment. Click **OK** to continue.



5. Click the title bar, and select **File Download Blocked-->Download File** to continue.

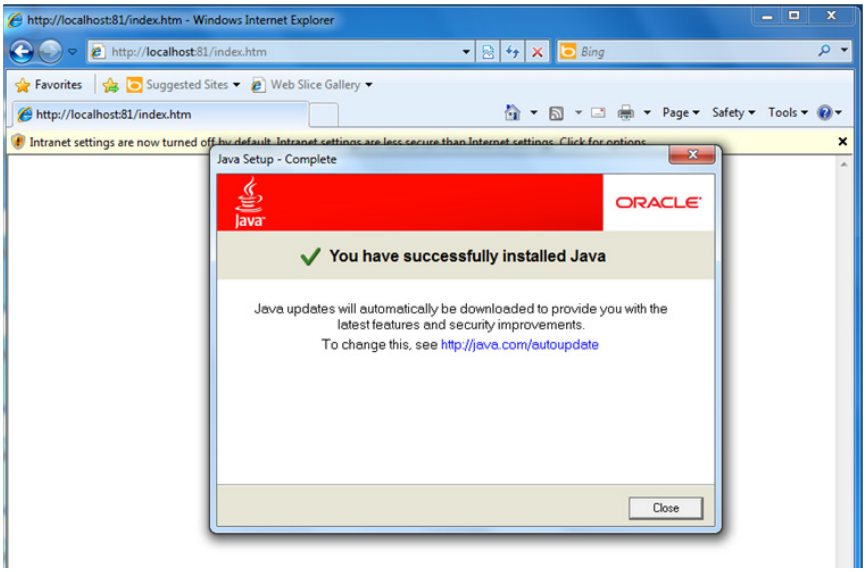


6. Select **Run** to download and install the Java Runtime Environment (JRE), and when Windows posts a security warning asking if you wish to run the installer, click **Run** again.

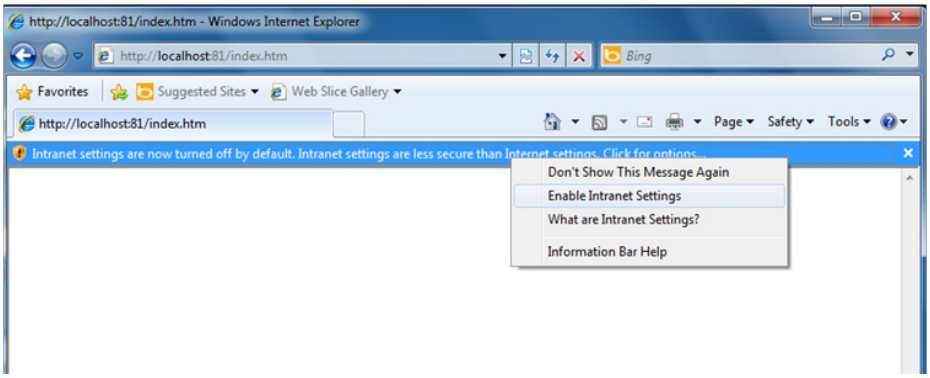


7. Click **Install** to continue.

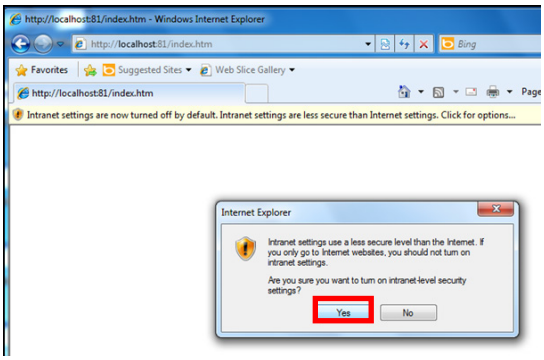
8. Click **Close** to complete.

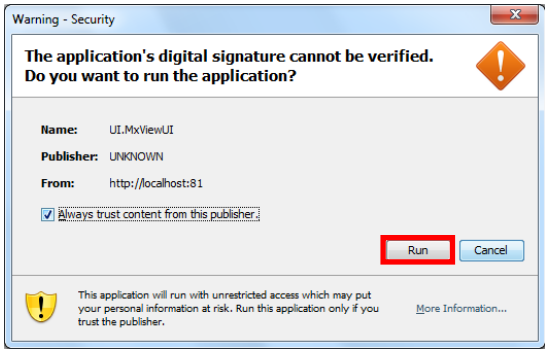


9. In Windows IE, a banner will appear at the top of the browser window. Click the message and select **Enable Intranet Settings**.

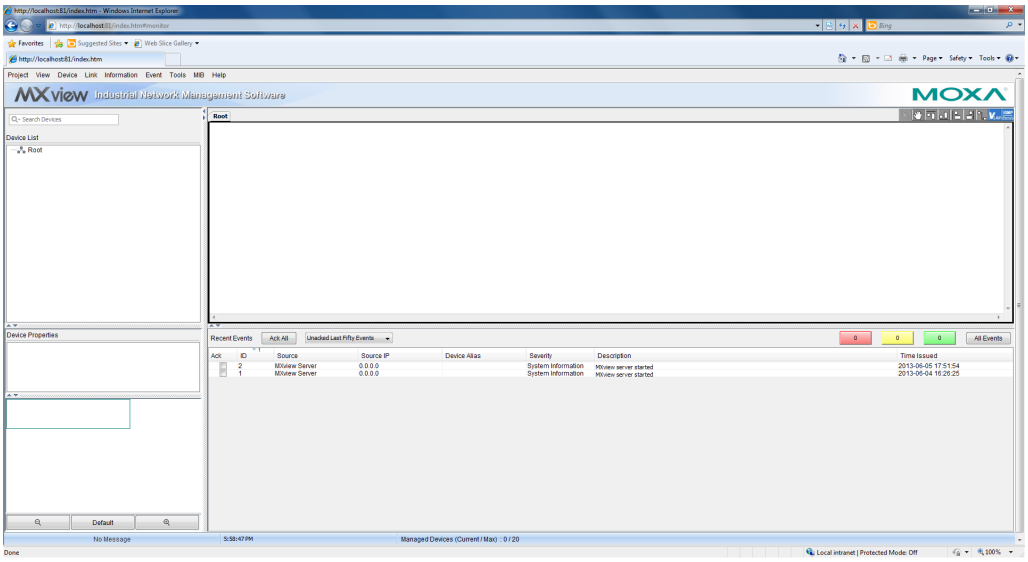


10. A security warning will appear, telling you that intranet settings are not secure enough for the open Internet. Click **Yes** to ignore this, and when another security warning appears telling you that the application's digital signature is not recognized, click **Run**.





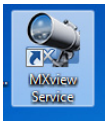
11. The **Moxa MXview Setup Wizard** will now appear. You may click **Next** if you wish to enter the setup routine, or select **Cancel** to launch the program immediately.
If you click **Cancel**, The program will be launched. It should look like the screenshot below.



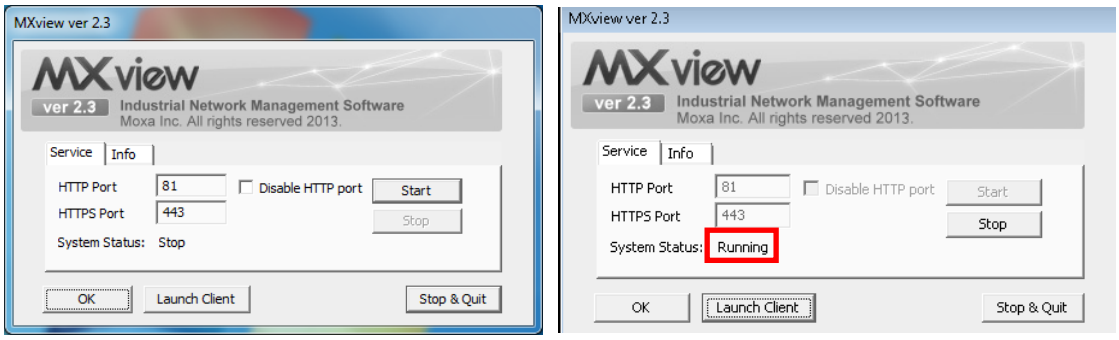
Loading the Synmap MIB File

To load the Synmap MIB file you must first have a running NMS; if you do not have an NMS, you may install the free version of MXview included with your TC-6110 computer software. If you have already started MXview, go directly to step 5 of this section.

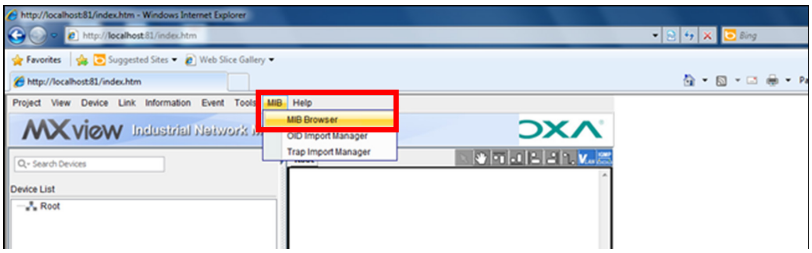
1. Click the MxView Service shortcut on the desktop.



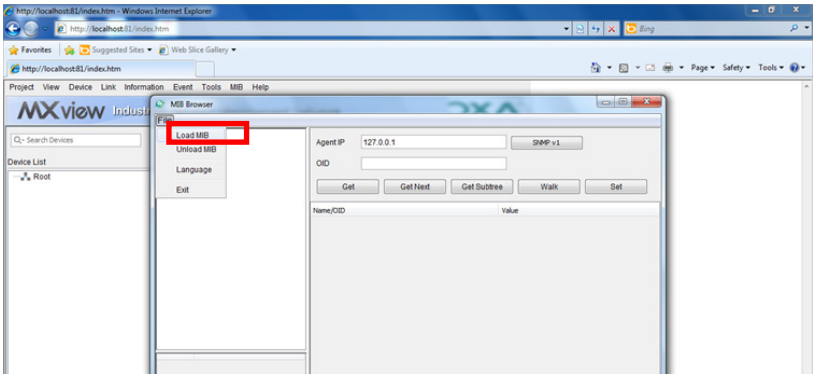
2. Click **Start**, wait for the **System Status** indicator to show **Running**, and then select Launch Client. When the MXview Setup Wizard appears, click Cancel to skip the setup process and directly open the MXview interface.



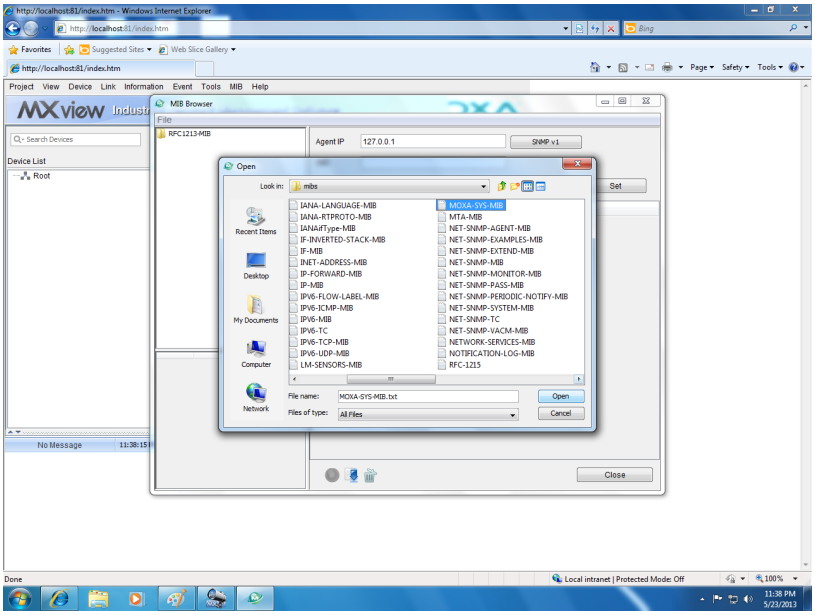
3. Select MIB-->MIB Browser.



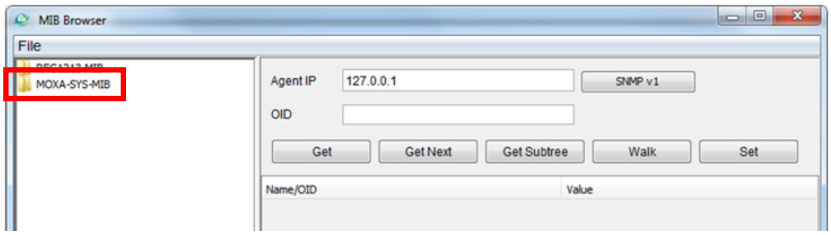
4. After the MIB browser has opened, select **File** from the browser's upper left corner, and then **Load MIB**.



5. Navigate to **`c:\usr\share\snmp\mibs\`** and select **`MOXA-SYS-MIB.txt`**.



6. After opening the Synmap MIB in the browser, check that it appears in the **File** window. If it is not, then it is likely because the MIB file is corrupted. To remedy this, re-copy the MIB file from the software DVD, and re-load the MIB file following the instructions above.

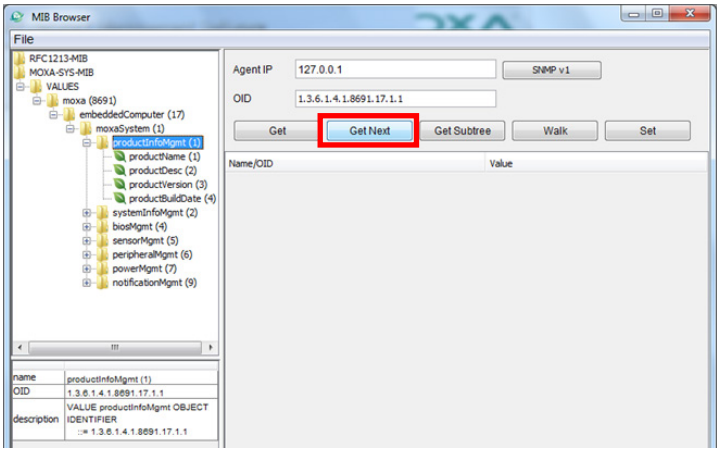


Using Synmap OIDs to Control the EXPC-1319

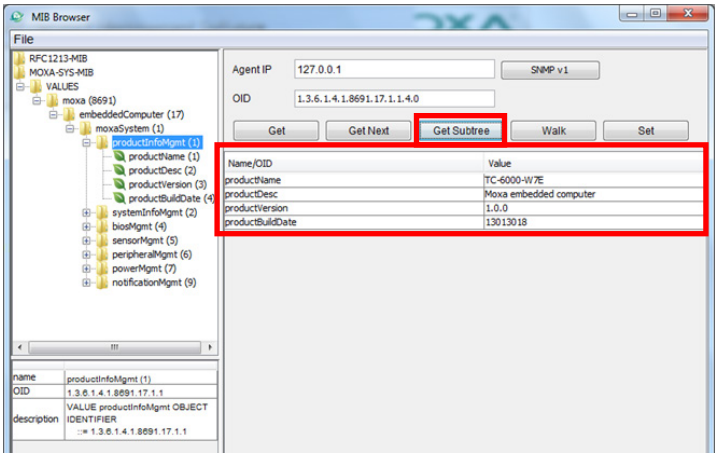
Follow these steps to use Synmap to use the Moxa MIB to set up automated controls for the EXPC-1319.

Retrieving Basic Device Information

1. In this first step, we will use Synmap to retrieve specific device information about the TC-6110. First, use the **Get Next** button to navigate the OID tree by clicking through these items: **`MOXA-SYS-MIB\VALUES\Moxa\embeddedComputer\MoxaSystem\productInfoMgmt`**



2. When you reach the final layer of OIDs, you will need to select **GetSubTree** to display the available information. When you use the MIB viewer to select the **productInfoMgmt** OID, you will see the following information displayed in the MIB viewer's information window: **Product Name** (TC-6110), **Product Description** (Moxa embedded computer), **Product Version** (1.0.0), and **Product Build Date** (13013018).

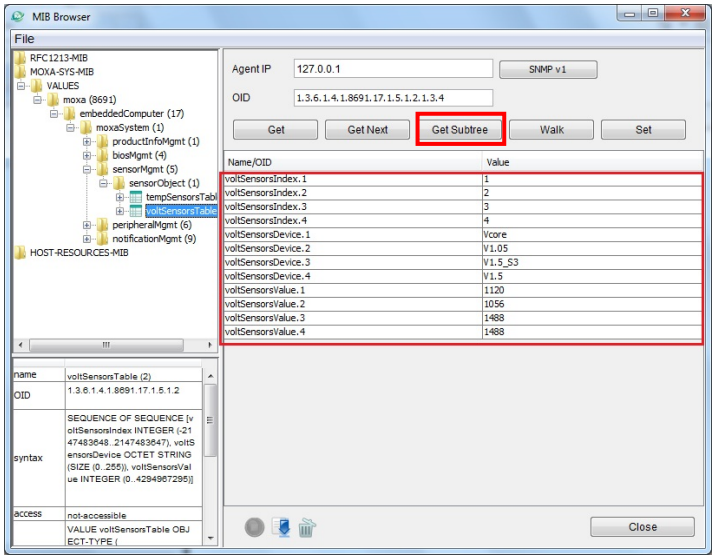


Using Synmap to Read the Voltage Sensor

The following table shows the OID of the voltage sensor, read/write option and available values.

Item Name	OID	Access	Description
voltSensorsIndex	1.3.6.1.4.1.8691.17.1.5.1.2.1.1	read-only	Returns a list of numbers that correspond with the voltage sensors, used by SNMP for identification; begins with 1
voltSensorsDevice	1.3.6.1.4.1.8691.17.1.5.1.2.1.2	read-only	Returns a list of string values identifying the voltage sensors by name/location. Possible values are **Vcore**, **V1.05**, **V1.5_S3**, **V1.5**.
voltSensorsValue	1.3.6.1.4.1.8691.17.1.5.1.2.1.3	read-only	Returns the sensor's reading, in volts

1. Start up MXview (or some other NMS) and open the MIB browser. For detailed instructions on loading MXview, you may refer to **Loading the Synmap MIB File**, steps 1 to 3.
2. In the MIB Browser, navigate to:
   MOXA-SYS-MIB\VALUES\moxa\embeddedComputer\moxaSystem\sensorMgmt\sensorObject\voltSensorTable.
3. Retrieve the MIB subtree.



4. Verify you are properly receiving the sensor information. The units displayed by the sensor are micro-volts (μV). For example, when the voltage is at 1.12 V, it will be diplayed as 1120 μV.

Using the Host Resources MIB

The **Host Resources MIB** is a mainstay industry standard, defined by RFC2790. The steps for using the EXPC-1319 Host Resources MIB with your chosen NMS are described below.

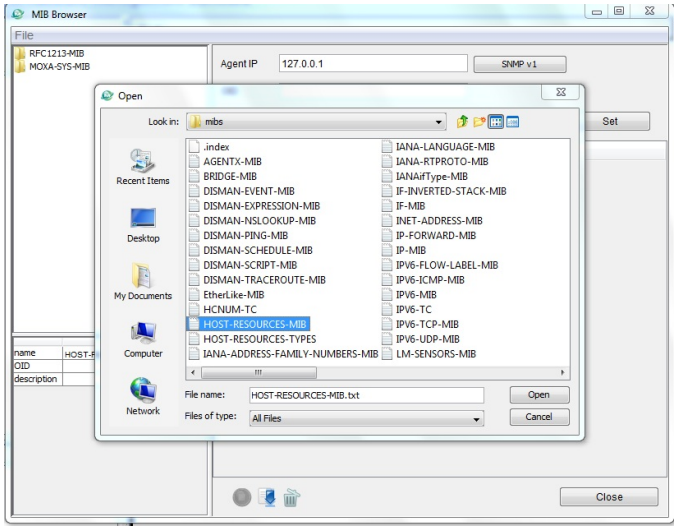
MXView does not provide table views of MIB data. If your NMS does, however, you may use any of the available OIDs that are provided in the **Host Resources MIB**. Use of MXView will, however, limit your available OIDs.

ATTENTION

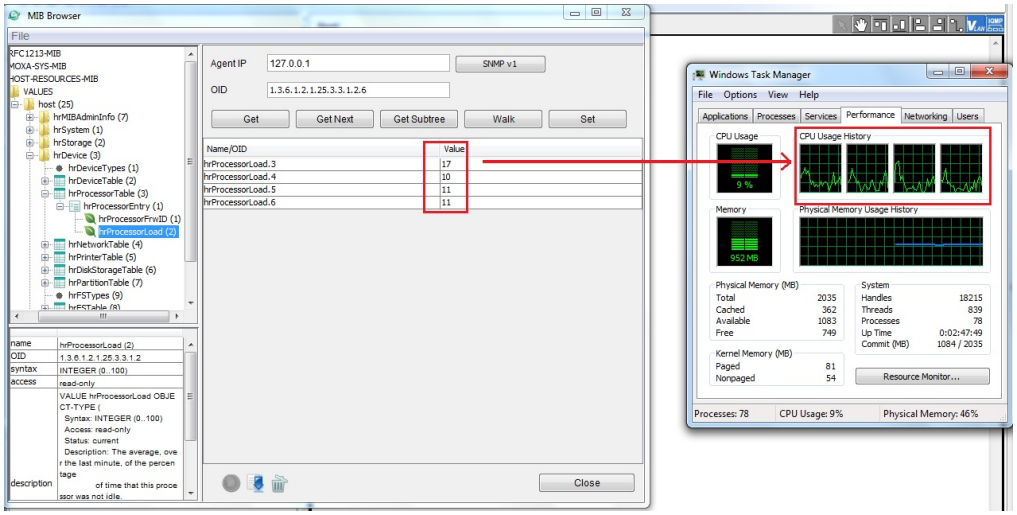
All of the information reproduced below regarding OIDs of the Host Resources MIB—as well as much more regarding—may be found using the **Cisco SNMP Object Navigator**, which may currently be found at this web address: http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en

Checking CPU Load Using the Host Resources MIB

1. Start up MXview (or some other NMS) and open the MIB browser. For detailed instructions on loading MXview, you may refer to **Loading the Synmap MIB File**, steps 1 to 3.
2. Load the **Host Resources MIB** file from **c:\usr\share\snmp\mibs\HOST-RESOURCES-MIB.txt**.



3. To check the CPU load, select **hrProcessorLoad(2)**. If you wish, you may evaluate these stats by comparing them stats with the CPU usage indicated by the **Windows Task Manager**.



Item Name	OID	Access	Description
hrProcessorLoad	1.3.6.1.2.1.25.3.3.1.2	read-only	The average, over the last minute, of the percentage of time that this processor was not idle. Implementations may only approximate this one minute smoothing period.

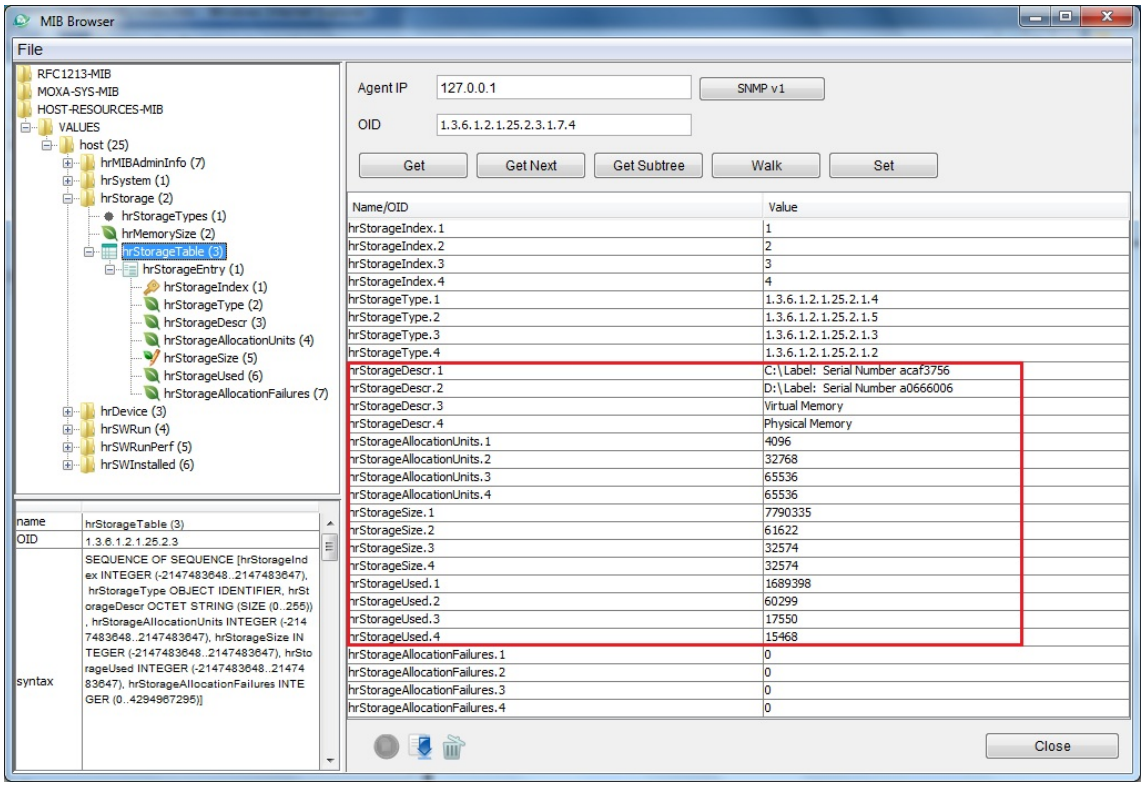
ATTENTION

All of the information reproduced below regarding OIDs of the Host Resources MIB—as well as much more regarding—may be found using the **Cisco SNMP Object Navigator**, which may currently be found at this web address: http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en

Checking Data Storage Stats Using the Host Resources MIB

There are several OIDs available for monitoring and/or manipulating a system's storage devices, whether virtual memory, RAM, fixed disks, or externally mounted disks (among others). These OIDs are available under the **hrStorageTable (3)**, which is itself found in the **Host Resources MIB** referenced above.

1. If you have not done so already, start up MXview (or some other NMS) and open the MIB browser. For detailed instructions on loading MXview, you may refer to **Loading the Synmap MIB File**, steps 1 to 3.
2. The Host Resources / Storage Table is located one level down the Host Resources tree, at the first stage.



The hrStorageTable OIDs

Below is a full list of all the OIDs available under the **hrStorageTable** tree.

Item Name	OID	Access	Description
hrStorageIndex	1.3.6.1.2.1.25.2.3.1.1	read-only	Returns a unique integer for each logical storage area available, beginning with 1
hrStorageType	1.3.6.1.2.1.25.2.3.1.2	read-only	Returns the OID of the type of storage associated with the **hrStorageIndex** number (above) that is appended to it.
hrStorageDescr	1.3.6.1.2.1.25.2.3.1.3	read-only	Returns a description of the type and instance of the associated storage area
hrStorageAllocationUnits	1.3.6.1.2.1.25.2.3.	read-only	The size, in bytes, how data objects on

	1.4		this device are allocated. This number will indicate if this device is allocating data in multiples of sectors, blocks, buffers, or packets.
hrStorageSize	1.3.6.1.2.1.25.2.3.1.5	read-write	The size of the storage represented by this entry, ***in units of hrStorageAllocationUnits*** (see above entry for details). This object is writable to allow remote configuration of the size of the storage area in those cases where such an operation makes sense and is possible on the underlying system.
hrStorageUsed	1.3.6.1.2.1.25.2.3.1.6	read-only	The amount of storage area that is already allocated, ***in units of hrStorageAllocationUnits***."
hrStorageAllocationFailures	1.3.6.1.2.1.25.2.3.1.7	read-only	This returns the number of requests that could not be honored due to not enough storage space. It should be noted that as this object has a SYNTAX of Counter32 it does not have a defined initial value.

Explaining Allocation Units

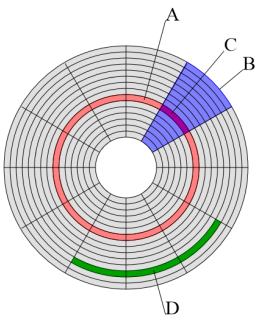
Below is a diagram showing how data is logically organized on a hard disk. As file systems such as NFTS have been adapted for use on solid state drives, the same basic logical organization remains relevant for SSDs.

A) **Disk track:** A track is like a groove in a record player, or a laser track on a CD. Tracks are arranged concentrically on a hard disk, and represent the pathways among which the scanner will shift as it navigates the surface of a disk. Tracks are divided into data sectors, which may or may not represent successively ordered blocks of data.

B) **Geometrical Disk Sector:** A geometrical sector represents the mathematical concept of a sector. It is not the same as a data sector.

C) **Track, or Data Sector:** Data sectors are the smallest block of information on a disk. Currently, data sectors are standardized at 4096 bytes per sector, or 4 kibibytes (kB) per sector.

D) **Cluster:** A cluster represents a series of sectors that lie upon the same track of a disk. **Allocation units** represent the fewest number of data sectors that may be used to store a block of information (i.e., a file). If the storage system designates three successive data sectors as the basic allocation unit, then all data stored on the disk will be stored in continuous sections of 12,1288 bytes. If 4 sectors are used, then the smallest possible block of data will be 16,384 bytes (or 16 kB).



Using the **hrStorageTable** OIDs, it is possible to determine all relevant statistics regarding your system's memory storage performance. The following example is a brief summary of the EXPC-1319's storage system:

	Description	Allocation Unit (byte)	Total Storage/Memory Size (allocation unit)	Used Storage/Memory Size (allocation unit)
1	C:\ (C drive)	4096 (1 sector)	7790335	1689398
2	D:\ (D drive)	32768 (8 sectors)	61622	60299

3	Physical Memory	65536	32574	15468

Ex. 1: To derive the total capacity of the C drive, use the formula:

hrStorageAllocationUnits.1 * hrStorageSize.1

4096 * 7790335 = 31909212160 bytes, or **30 GB**

Ex. 2: If you want to know the physical memory (RAM) capacity, use the following formula:

hrStorageAllocationUnits.4 * hrStorageSize.4

65536 * 15468 = 1013710848 byte, or **1 GB**

Ex. 3: If you want to calculate what percent of physical memory (RAM) is currently allocated, use the formula:

hrStorageSize.4 / hrStorageUsed.4

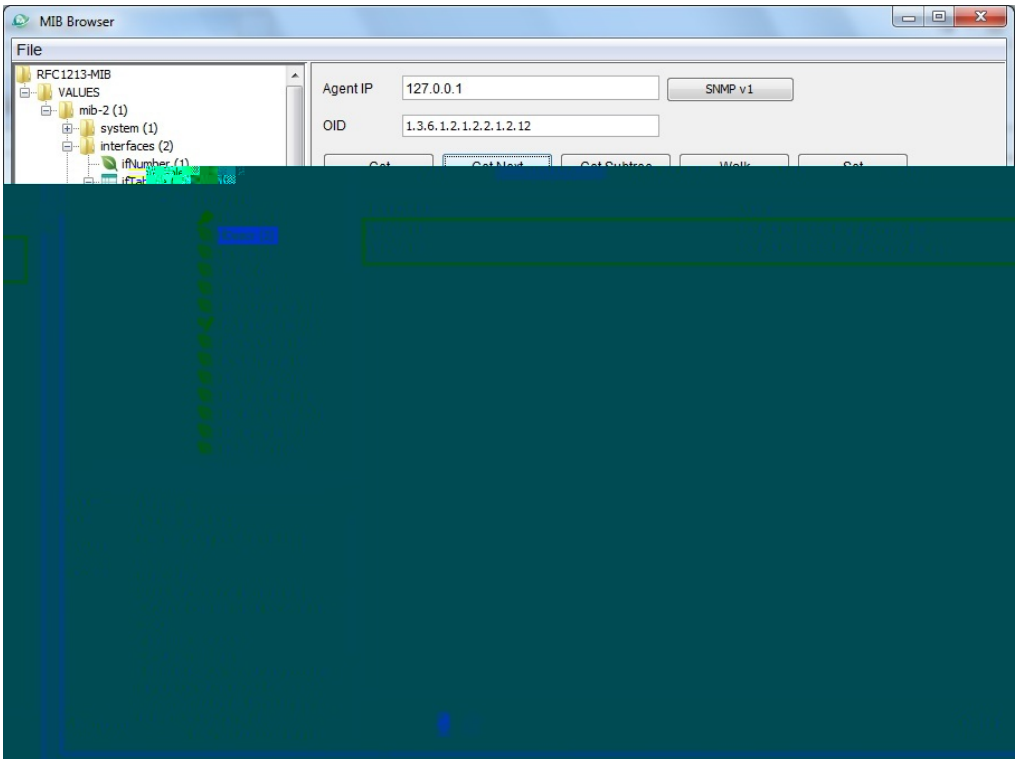
15468 / 32574 = 0.474, or **47.4%**

Checking Network Status

Use these OIDs to check the network status:

Item Name	OID	Access	Description
ifDescr	1.3.6.1.2.1.2.2.1.2	read-only	Returns a textual string containing information about the interface. This string should include the name of the manufacturer, the product name, and the version of the interface hardware/software.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	read-only	

1. Based on the RFC1213-MIB, select **ifDescr (2)** from the list of the left side in the MIB browser. You can view the Ethernet controller on the right column.



2. Select **ifOperStatus (8)** to check the status of the Ethernet controller. **1** indicates that the Ethernet port is connected, while 2 indicates that the Ethernet port is disconnected.

8

Sample Code

This chapter describes how to use various examples on the EXPC-1319-STS computers for different functions.

The following topics are covered in this chapter:

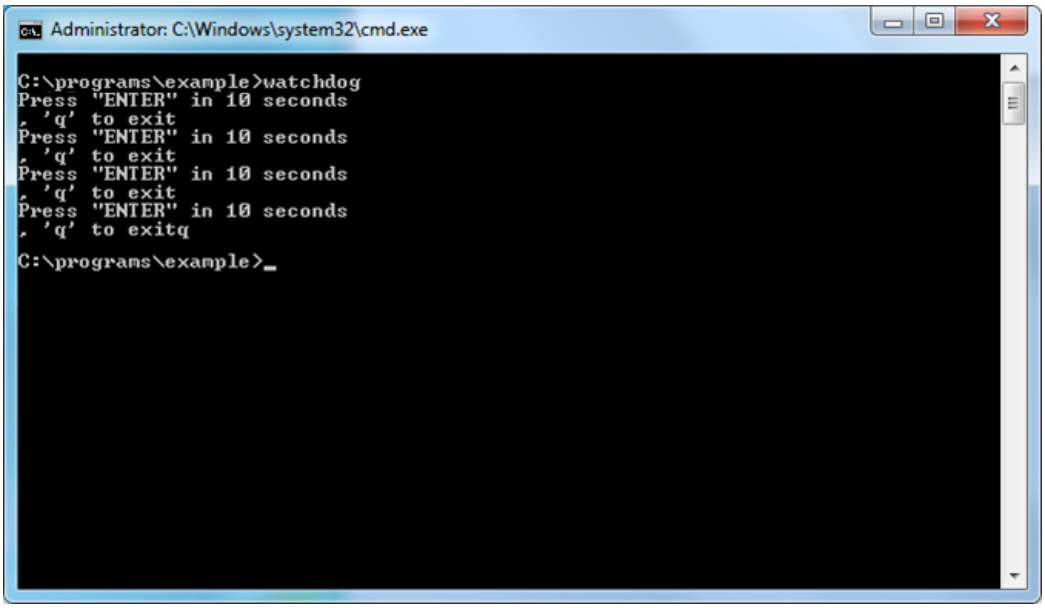
- **Watchdog**
  - Enabling Watchdog Function

Watchdog

EXPC-1319 computers provide sample code for enabling the watchdog timer, found under <Software DVD>**\examples\C++\WatchDog**. The executable file **Watchdog.exe** is under <Software DVD>**\examples\Release**.

Enabling the Watchdog Timer

1. If you haven't already, create a `c:\programs\example` folder and copy **Watchdog.exe** into the folder.
2. Execute **Watchdog.exe**; once the watchdog is running, you will need to press **Enter** in every 10 seconds or the system will reboot.
3. To stop the watchdog timer, press **q** to exit the program.



9

System Recovery

The EXPC-1319-STS ready-to-run embedded computers are a Windows Embedded Standard 7 platform. This chapter describes the recovery process in the event of system instability.

The following topics are covered in this chapter:

- **Recovery Environment**
- **Recovery Procedure**
- **Saving the System to the USB Drive**

Setting Up the Recovery Environment

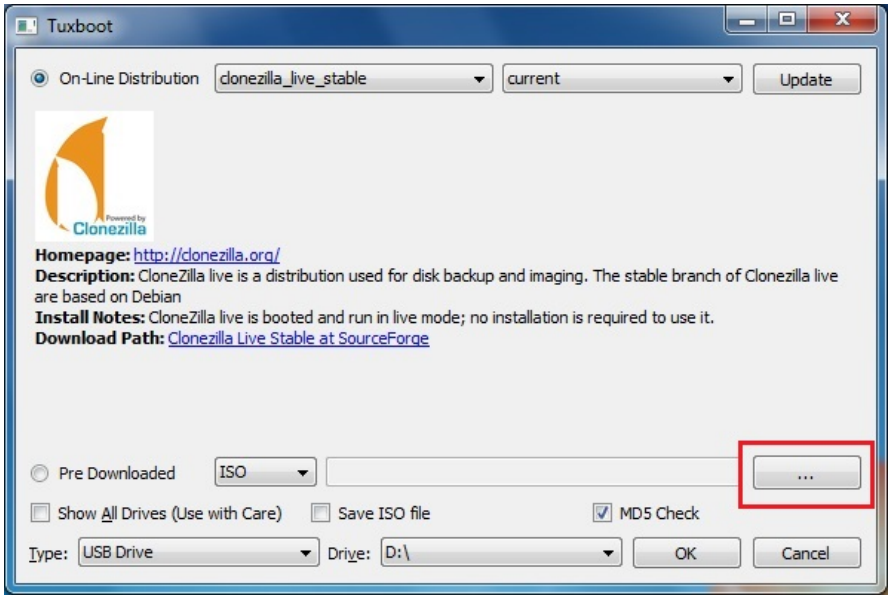
In this section, you will learn how to prepare a USB drive with the recovery environment and system image, and how to set up the system for a system recovery. The EXPC-1319's system recovery is built on the **Clonezilla** system recovery solution.

To create a system recovery image you will need to create a bootable USB drive containing the recovery environment and a duplicate image of the platform software. The USB drive should be at least 2GB, though larger USB drives will provide better performance.

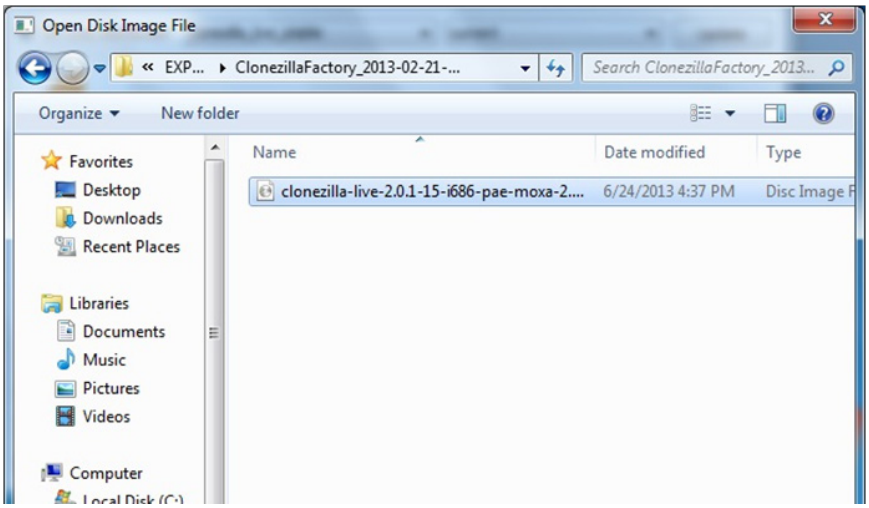
Creating the Recovery Environment

In this section, you will copy the Clonezilla recovery environment over to a USB drive, and then copy a full image of your Windows 7 operating system into the Clonezilla file system. After completing this section, your USB drive will be able to function as a system recovery environment.

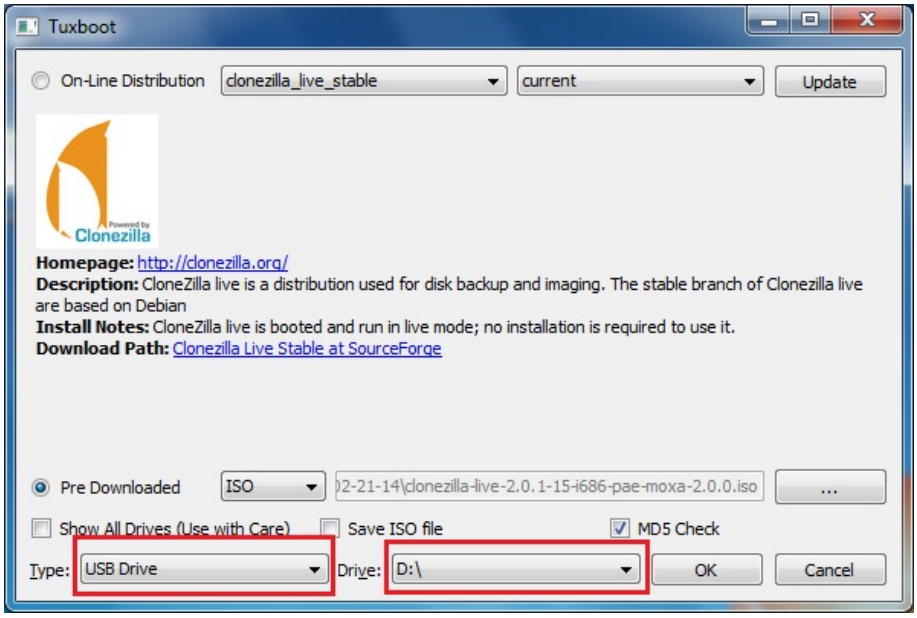
1. Execute **tuxboot-windows-23.exe** from the <Software DVD>**\recovery\EXPC-1319-STS-W7E**
2. Select **Pre Download**
3. Click the ellipses ("**...**").



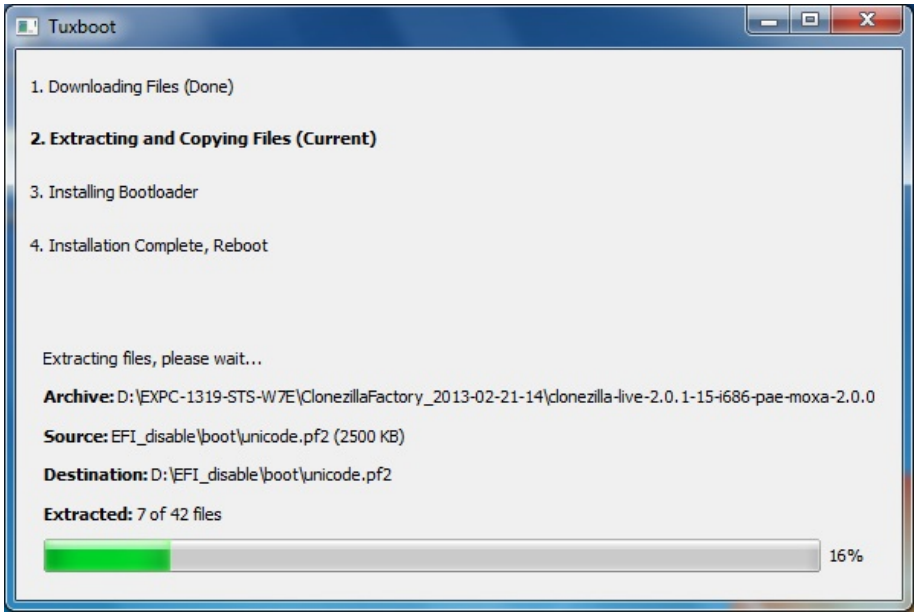
4. Select the ISO file from <Software DVD>**\recovery\ EXPC-1319-STS-W7E \ClonezillaFactory_2013-02-21-14**.



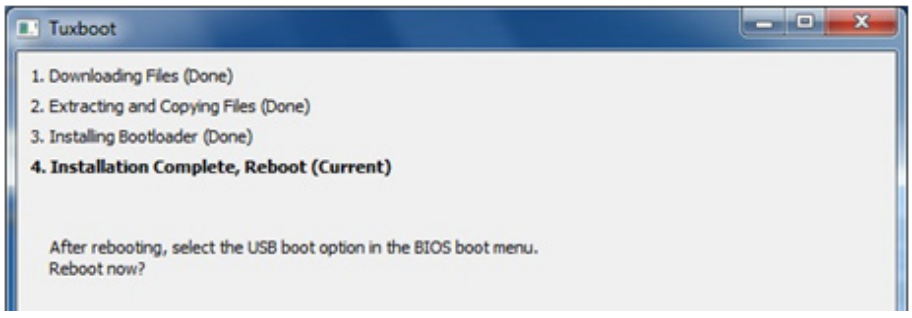
5. Select the type of device to be used to store the recovery environment. For this example that will be **USB Drive**.



6. Next, select what **Drive** the USB is mounted under. This will most likely be **D:**.
7. Click **OK** to continue; the boot files will begin to be copied to your USB drive.



8. When finished, click **Exit** to stop the program.

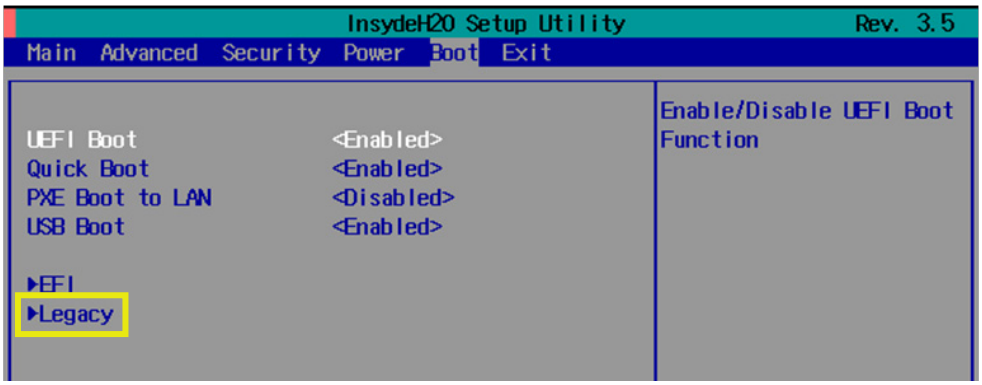


9. Finally, you should manually copy the Windows 7 Embedded operating system image into the Clonezilla file system. To do this, copy the **os_image** directory from <Software DVD>**\EXPC-1319-STS-W7E** `recovery` folder over to the **\home\partimag** on the USB drive.

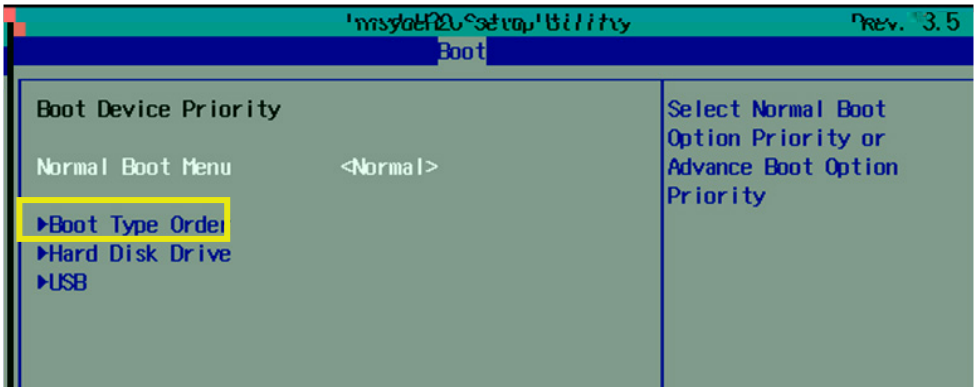
Setting up the BIOS

To enable the system to boot from the USB drive you will need to change the BIOS settings.

1. Turn on the computer and when you hear a beep press **F2**; this will take you to the BIOS setup menu.
2. Select **Boot** from the ribbon of tabs that runs across the top of the screen.
3. Next, select **Legacy**.

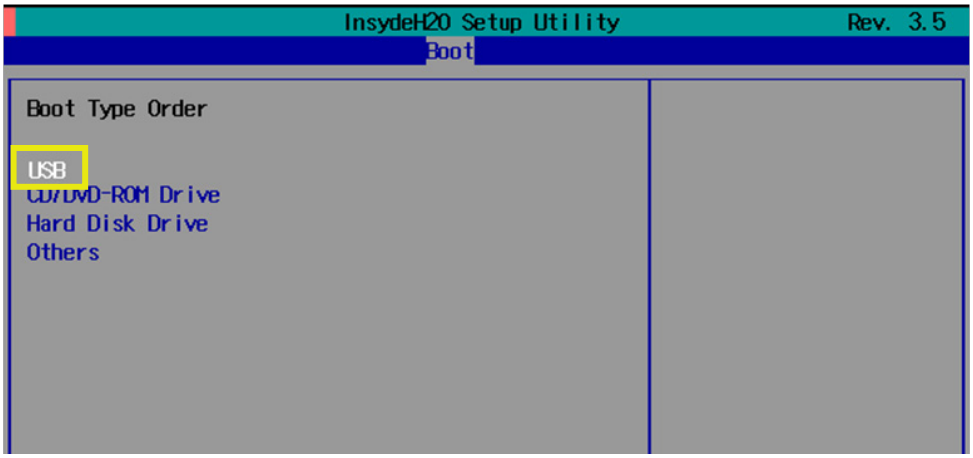


4. Press **Enter** to continue.
5. Select **Boot Type Order**.



6. Select USB drive
7. Press the plus sign (**+**) to move the USB entry into the first boot device position.

   Warning: An incorrect boot priority will lead to recovery failure.

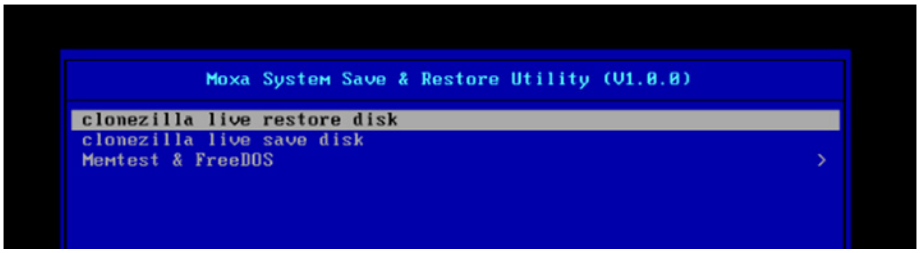


8. Press **F10** and then press **Enter** to save and exit theBIOS setup.

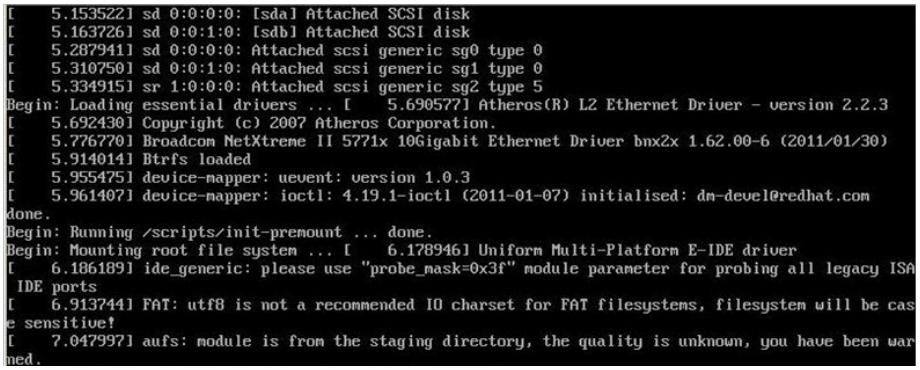
Restore a System Image to the Main System Hard Drive

This section will show you how to boot into the **Clonezilla** rescue environment to re-copy the system image over to your platform's main hard drive. To do this, you will need to have completed the above section, **Setting up the BIOS**. Connect the USB drive to any of the EXPC-1319-STS's USB ports and then reboot the computer. The system should now boot from the USB drive into the Clonezilla rescue environment.

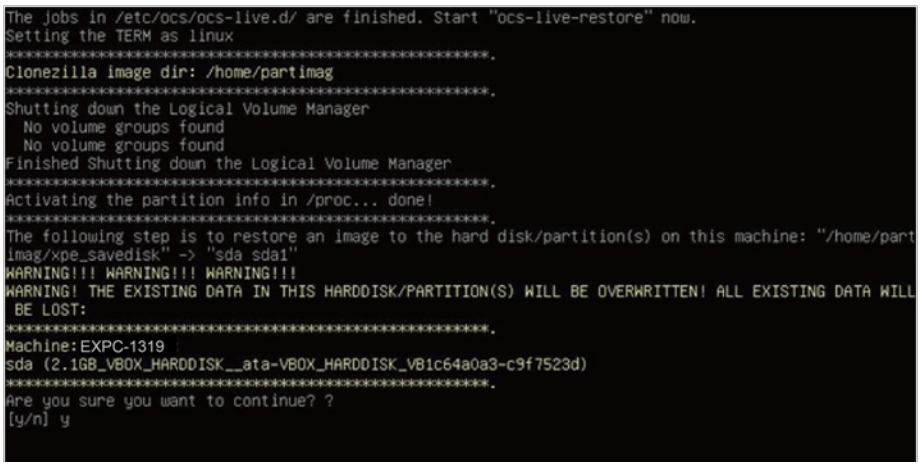
1. Once Clonezilla has fully booted, select the first option, **clonezilla live restore disk**.



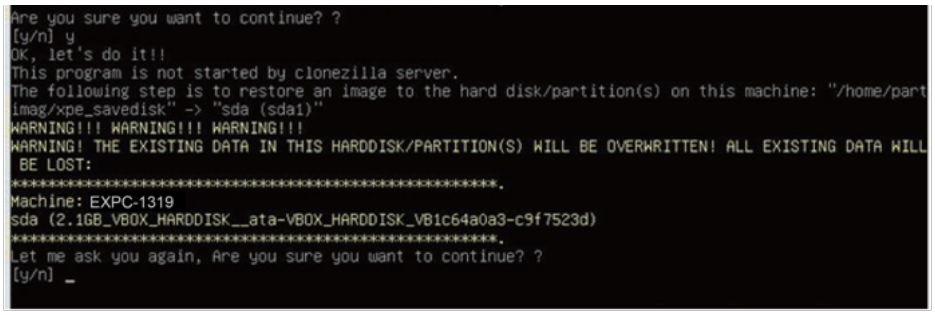
2. The USB drive will then serve the full rescue environment to the computer.



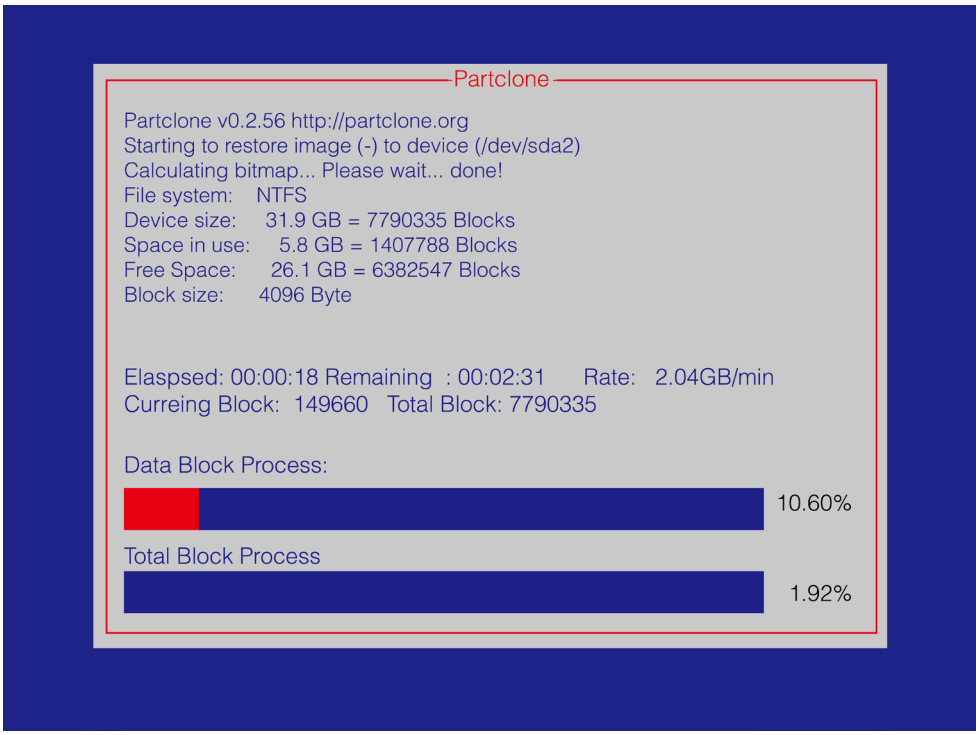
3. Clonezilla will warn you that you are about to erase your entire OS. Enter **y** to continue.



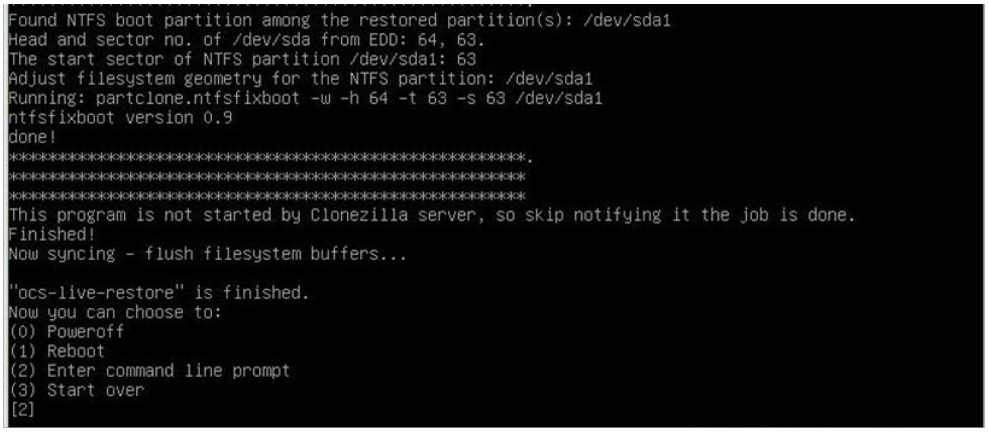
4. Clonezilla will warn you again. Enter **y** to confirm again.



5. Wait for the files to be copied over. Depending on the speed of your USB, this could take some time.



6. Select **(0) Poweroff** to power off the computer.

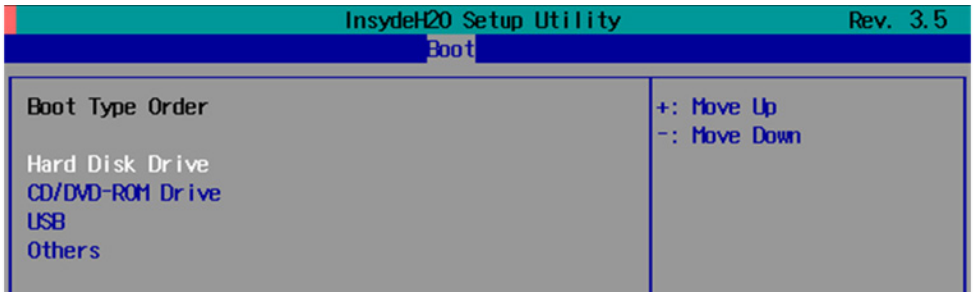


7. Remove the USB drive after the computer has been powered off.

Return the BIOS to Its Original Setup

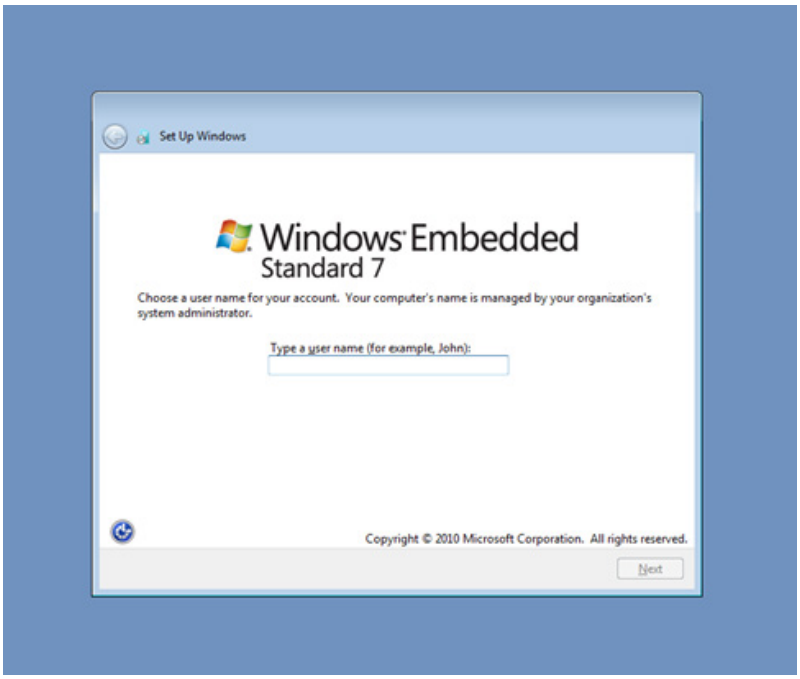
For security's sake, you should change the boot priority so that the system will now boot from the main system drive. Turn on the system, and as it reboots press **F2** to enter the BIOS setup menu.

1. Select **Hard Disk Boot Priority** and then press the plus sign (**+**) to move the Hard Disk Drive entry to the priority boot position



2. Press **Enter.**
3. Press **F10** and then **Enter** to save and exit BIOS settings.
4. Reboot the computer.

After the system re-install you will need to wait about 10 to 15 minutes for the system to restart. This is because it will automatically go through the boot process two time, to re-initiate the system configuration files. **Do not turn off the computer or shut down the computer while the system is restarting**; otherwise, the IIS service will be terminated, and you will likely need to restart the restoration process to return the computer to its original, full operating state. When the operating system has successfully launched, you will need to restart your computer so that the new settings can be activated.

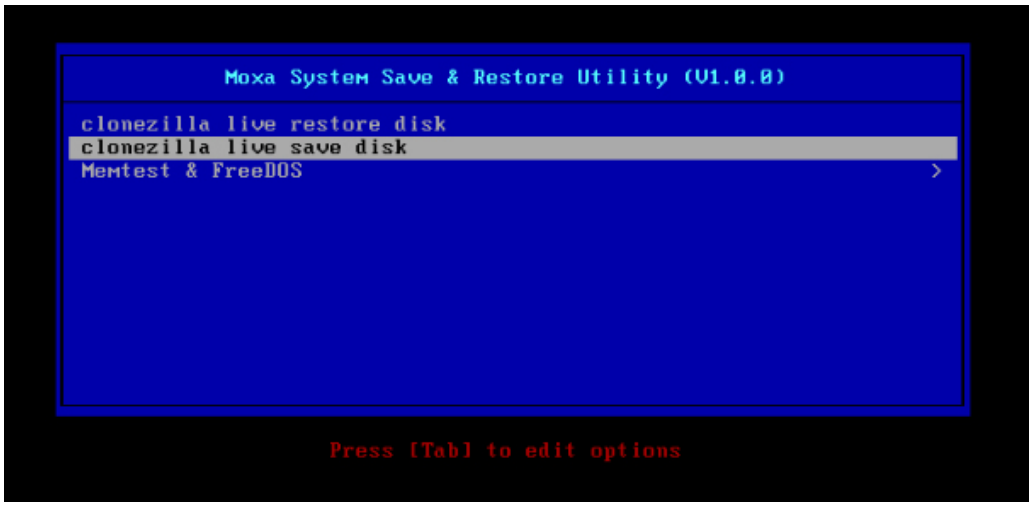


Saving the System to the USB Drive

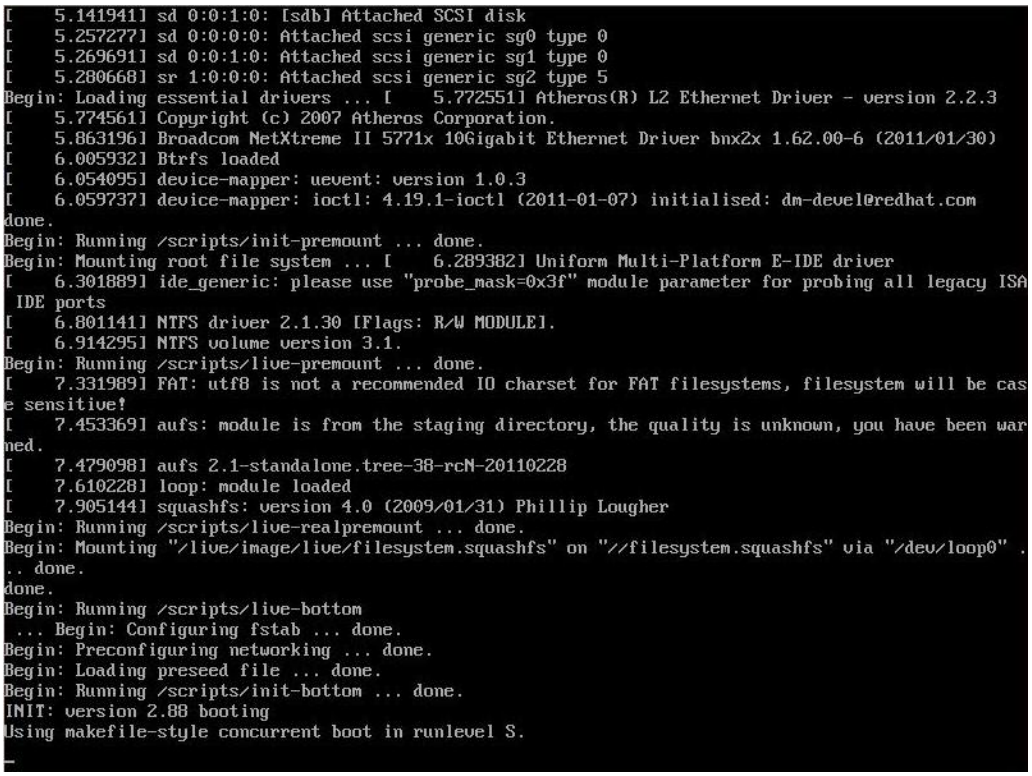
In this section you will learn how to use Clonezilla to save the entire system to the USB drive. Before saving the system to the USB drive, we suggest you remove all files under **\home\partimag** on the USB drive. Be sure to return the BIOS settings (for details, see **Setting up the BIOS**) to make the USB drive the first boot priority.

When the system has booted into the Clonezilla operating environment, take the following steps:

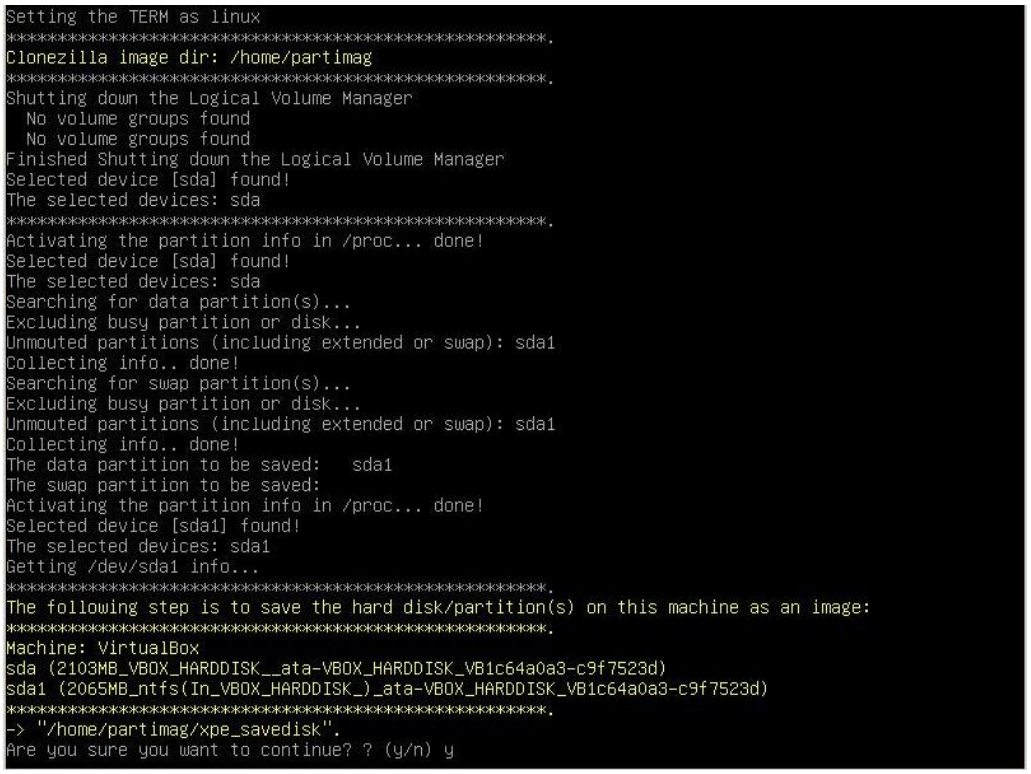
1. Select **clonezilla live save disk**. This will take you into the Clonezilla image generation process, where will the EXPC-1319's entire software platform will be automatically copied over to your USB. It is very important to make sure your USB drive is large enough to accommodate all of the system data.



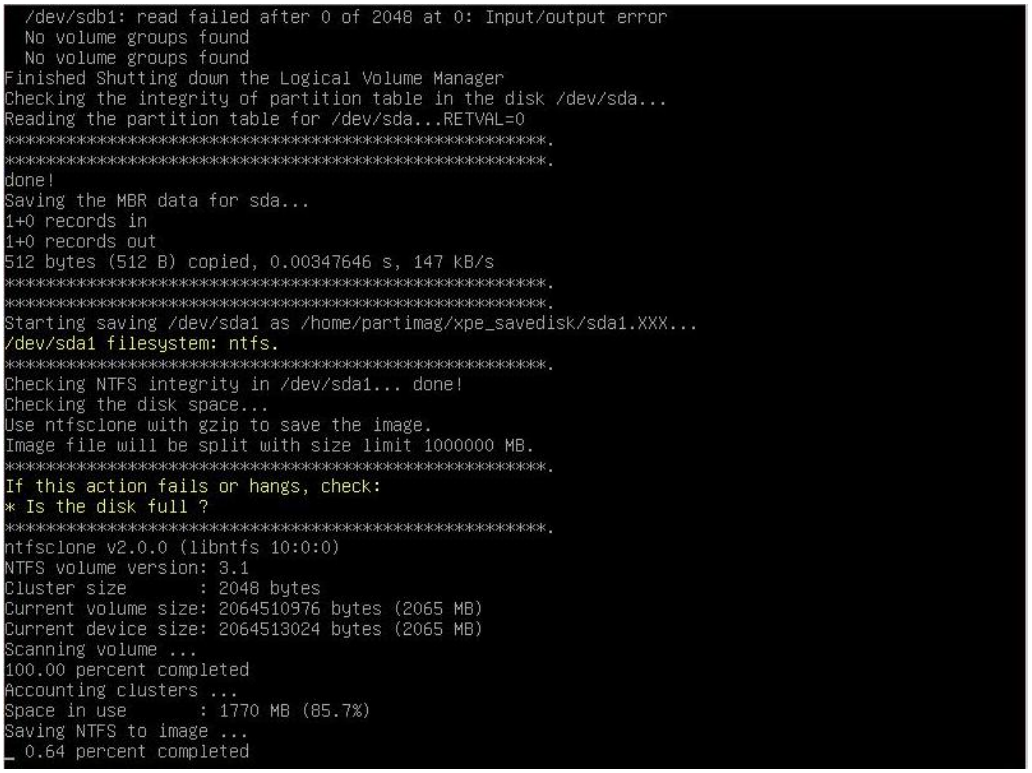
2. Wait for the USB drive boot process to finish.



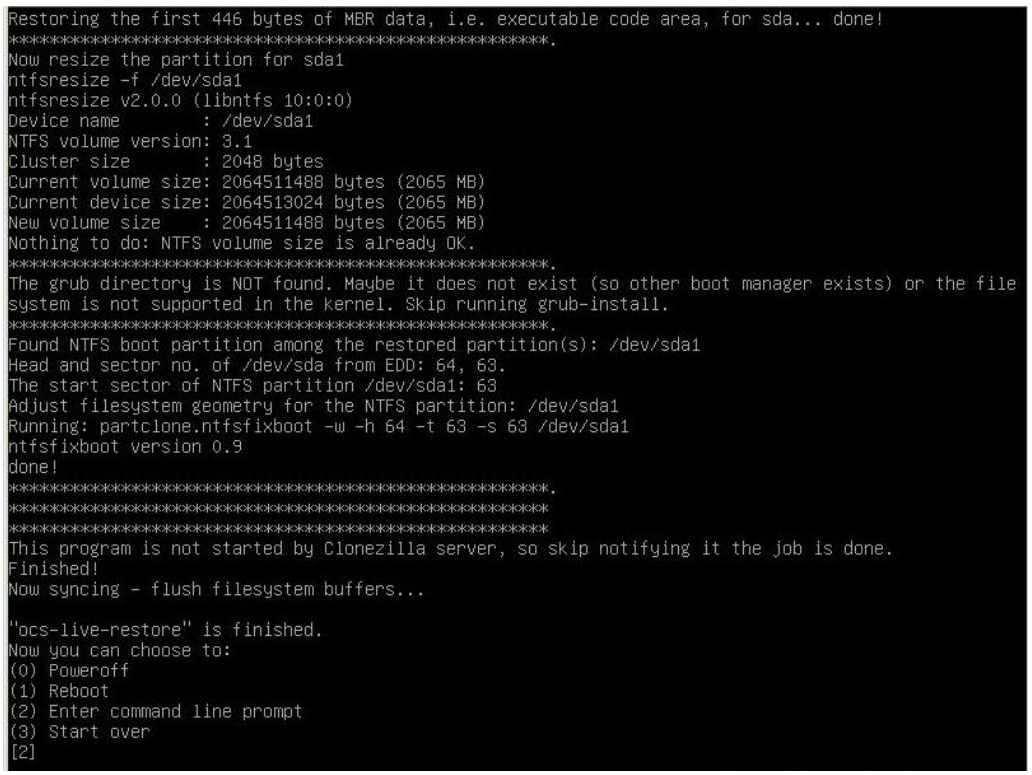
3. Clonezilla will warn you that you are about to erase all files currently located in the USB's image directory. Enter **y** to confirm that you want to continue.



4. Wait for the copying process to finish.



5. Once the new image is completed, select **(0) Poweroff** to turn off the computer.



6. After the computer has turned off, remove the USB from the port and return the BIOS to its original operating state (for details, see **Returning the BIOS to Its Original Setup**, above).

A

Moxa Synmap OID Table

This appendix describes the Moxa SynMap OID Table

The following topics are covered in this appendix:

- **Moxa SynMap OID Table**

The Moxa Synmap OID Table

Item Name	OID	Access	Description	Supported
productName	1.3.6.1.4.1.8691.17.1.1.1	read-only	Returns product name.	■
productDesc	1.3.6.1.4.1.8691.17.1.1.2	read-only	Returns product short description.	■
productVersion	1.3.6.1.4.1.8691.17.1.1.3	read-only	Returns product version.	■
productBuildDate	1.3.6.1.4.1.8691.17.1.1.4	read-only	Returns product last build date, the format is YYMMDDHH.	■
systemCpuUsage	1.3.6.1.4.1.8691.17.1.2.1.1	read-only	Show CPU usage rate (0-100 %).	
systemMemUsage	1.3.6.1.4.1.8691.17.1.2.1.3	read-only	Show memory usage rate (0-100 %).	
systemUptime	1.3.6.1.4.1.8691.17.1.2.1.5	read-only	The amount of time since this host was last initialized.	
systemTotalUptime	1.3.6.1.4.1.8691.17.1.2.1.6	read-only	The amount of time from total boot up time.	
systemMemorySize	1.3.6.1.4.1.8691.17.1.2.3.1	read-only	The amount of physical main memory contained by the host.	
systemVolumeCount	1.3.6.1.4.1.8691.17.1.2.3.2	read-only	Show total volume count.	
systemVolumeIndex	1.3.6.1.4.1.8691.17.1.2.3.3.1.1	read-only	Reference index for each observed device.	
systemVolumeName	1.3.6.1.4.1.8691.17.1.2.3.3.1.2	read-only	The name of the volume.	
systemVolumeLabel	1.3.6.1.4.1.8691.17.1.2.3.3.1.3	read-only	The label of the volume.	
systemVolumeSize	1.3.6.1.4.1.8691.17.1.2.3.3.1.4	read-only	The total size of the volume.	
systemVolumeAvail	1.3.6.1.4.1.8691.17.1.2.3.3.1.5	read-only	The available size of the volume.	
biosVersion	1.3.6.1.4.1.8691.17.1.4.1	read-only	Returns the BIOS version.	
biosSaveSetting	1.3.6.1.4.1.8691.17.1.4.2	read-write	Write 1 to save bios setting, and read 0 mean setting had been applied.	
biosSettingStatus	1.3.6.1.4.1.8691.17.1.4.3	read-only	Returns compare of bios CMOS setting and bios new setting.	
bootDeviceStatus	1.3.6.1.4.1.8691.17.1.4.4.1	read-only	Returns the current support boot device.	
firstBootDevice	1.3.6.1.4.1.8691.17.1.4.4.2	read-write	read show current first boot device, write set boot device.	

pwrOnAfterPwrFail	1.3.6.1.4.1.8691.17.1.4.8.1	read-write	Select power on after power fail behavior.	
pwrLanWakeUp	1.3.6.1.4.1.8691.17.1.4.8.3	read-write	Enable/Disable wake on LAN functionality.	
tempSensorsIndex	1.3.6.1.4.1.8691.17.1.5.1.1.1.1	read-only	Reference index for each observed device.	■
tempSensorsDevice	1.3.6.1.4.1.8691.17.1.5.1.1.1.2	read-only	The name of the temperature sensor we are reading.	■
tempSensorsValue	1.3.6.1.4.1.8691.17.1.5.1.1.1.3	read-only	The temperature of this sensor in mC.	■
voltSensorsIndex	1.3.6.1.4.1.8691.17.1.5.1.2.1.1	read-only	Reference index for each observed device.	
voltSensorsDevice	1.3.6.1.4.1.8691.17.1.5.1.2.1.2	read-only	The name of the device we are reading.	
voltSensorsValue	1.3.6.1.4.1.8691.17.1.5.1.2.1.3	read-only	The voltage in mV.	
accelerometerIndex	1.3.6.1.4.1.8691.17.1.5.1.3.1.1	read-only	Reference index for each observed device.	■
accelerometerAxis	1.3.6.1.4.1.8691.17.1.5.1.3.1.2	read-only	The name of the accelerometer axis we are reading.	■
accelerometerValue	1.3.6.1.4.1.8691.17.1.5.1.3.1.3	read-only	The accelerometer value in mG.	■
accelerometerTimestamp	1.3.6.1.4.1.8691.17.1.5.1.3.1.4	read-only	The timestamp when accelerometer measured.	■
ioDiNumber	1.3.6.1.4.1.8691.17.1.6.1.1.1	read-only	Number of digital input pin in current system.	
diIndex	1.3.6.1.4.1.8691.17.1.6.1.1.2.1.1	read-only	Reference index for each digital input pin.	
diPort	1.3.6.1.4.1.8691.17.1.6.1.1.2.1.2	read-only	The port number of digital input pin.	
diValue	1.3.6.1.4.1.8691.17.1.6.1.1.2.1.3	read-only	The digital input status, 0 is low, 1 is high.	
diTrapEnable	1.3.6.1.4.1.8691.17.1.6.1.1.2.1.4	read-write	Agent will send trap message when digital input pin status changed and this object enbeled.	
ioDoNumber	1.3.6.1.4.1.8691.17.1.6.1.1.3	read-only	Number of digital output pin in current system.	
doIndex	1.3.6.1.4.1.8691.17.1.6.1.1.4.1.1	read-only	Reference index for each digital output pin.	
doPort	1.3.6.1.4.1.8691.17.1.6.1.1.4.1.2	read-only	The port number of digital output pin.	
doValue	1.3.6.1.4.1.8691.17.1.6.1.1.4.1.3	read-write	The digital output status,	

			0 is low, 1 is high.	
ledNumber	1.3.6.1.4.1.8691.17.1.6.2.1	read-only	Number of LED in current system	
ledIndex	1.3.6.1.4.1.8691.17.1.6.2.2.1.1	read-only	Reference index for each LED.	
ledPort	1.3.6.1.4.1.8691.17.1.6.2.2.1.2	read-only	The port number of LED.	
ledValue	1.3.6.1.4.1.8691.17.1.6.2.2.1.3	read-write	The LED status, 0 is low, 1 is high.	
uartNumber	1.3.6.1.4.1.8691.17.1.6.3.1	read-only	Number of internal UART in current system.	
uartIndex	1.3.6.1.4.1.8691.17.1.6.3.2.1.1	read-only	Reference index for each UART port.	
uartType	1.3.6.1.4.1.8691.17.1.6.3.2.1.2	read-write	The UART mode, 0 is RS232, 1 is RS485 2 wires, 2 is RS422, 3 is RS485 4 wires.	
usbNumber	1.3.6.1.4.1.8691.17.1.6.4.1.1	read-only	The number of ports regardless of their current state in the usb general port table.	■
usbDeviceIndex	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.1	read-only	The index is dentical to usbPortIndex for the correspondent USB port.	■
usbDeviceVendorID	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.2	read-only	The USB device port vendor HEX-formatted string as it is provided to the USB host by the USB device.	■
usbDeviceProductID	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.3	read-only	The product ID HEX-formatted string as it is provided to the USB host by the USB device.	■
usbDeviceActiveClass	1.3.6.1.4.1.8691.17.1.6.4.1.3.1.4	read-only	This object returns USB Device Class type of the active configuration	■
usbPlugTrapEnable	1.3.6.1.4.1.8691.17.1.6.4.1.4	read-write	Agent will send trap message when USB device inserted or removed and this object enabled.	
watchdogPeriod	1.3.6.1.4.1.8691.17.1.6.6.2.1	read-write	Watchdog period, 0 means disable watchdog monitor program; otherwise enable watchdog monitor program and configure	■

			the expired time.	
watchdogStatus	1.3.6.1.4.1.8691.17.1.6.6.2.2	read-write	To show the watchdog monitor program status.	■
powerPolicy	1.3.6.1.4.1.8691.17.1.7.2	read-write	Current system power policy.	■
moxaSystemTrapIP	1.3.6.1.4.1.8691.17.1.9.1	read-write	Set Trap IP address.	
moxaSystemTrapCommunity	1.3.6.1.4.1.8691.17.1.9.2	read-write	Trap community.	