Intel® EP80579 Software Drivers for Embedded Applications on Microsoft® Windows® XP Embedded

Getting Started Guide

September 2009
INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or by visiting Intel's Web Site.

Any software source code reprinted in this document is furnished under a software license and may only be used or copied in accordance with the terms of that license.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.


*Other names and brands may be claimed as the property of others.

Copyright © 2009, Intel Corporation. All rights reserved.
Contents—EP80579 Integrated Processor

1.0 Introduction ..............................................................................................................................................6
   1.1 About this Manual ...............................................................................................................................6
   1.2 Additional Information on Software ..................................................................................................6
      1.2.1 Where to Find Current Software and Documentation .................................................................6
      1.2.2 Product Documentation ................................................................................................................6
      1.2.3 Pre-boot Firmware .........................................................................................................................7
   1.3 Related Software and Documentation ..............................................................................................7
   1.4 Conventions .........................................................................................................................................7
   1.5 Software Overview .............................................................................................................................7
      1.5.1 Features Implemented ...................................................................................................................7
      1.5.2 List of Files in Release ..................................................................................................................7
      1.5.3 Package Release Structure ............................................................................................................7

2.0 Configuration Requirements ..................................................................................................................9
   2.1 Development Board Configuration ....................................................................................................9
      2.1.1 Package Components ...................................................................................................................9
      2.1.2 Development Kit Setup ..............................................................................................................9
      2.1.3 Safety ..........................................................................................................................................9
      2.1.4 Connecting the Serial ATA Hard Drive and Cable .......................................................................10
      2.1.5 Connecting the Keyboard and Mouse .........................................................................................10
      2.1.6 Connecting the PCI Express Video Card .....................................................................................10
      2.1.7 Connecting the Serial ATA DVD-ROM Drive (Optional) ............................................................11
      2.1.8 Connecting the Power Cables ......................................................................................................11
      2.1.9 Powering Up the System ............................................................................................................11
   2.2 Development Board Setup Requirements .........................................................................................15
      2.2.1 Stand-alone Target System ........................................................................................................15

3.0 System Requirements for Installing an OS on a Development Board ................................................16
   3.1 Installing Windows XP Embedded ...................................................................................................16
      3.1.1 Build Environment Requirements ..............................................................................................16
      3.1.2 Windows XP Embedded Development Toolset .........................................................................17

4.0 Configuring and Building an XP Embedded Image ..............................................................................18
   4.1 Unpacking the EP80579 Software Package ......................................................................................18
   4.2 Target Analyzer - Analyzing Target Device ......................................................................................18
   4.3 Import a Target Device .pmq File ........................................................................................................19
   4.4 Importing the Embedded Software Release .sld File ........................................................................20
   4.5 Inspecting the Intel® EP80579 Software Drivers for Embedded Applications Release .sld File .......21
   4.6 Creating a Target Device Image ..........................................................................................................22
   4.7 Transferring Windows XP Embedded Image to Target Device ......................................................26
      4.7.1 Dual Boot Windows XP Embedded Image Deployment ...............................................................27

5.0 Runtime Configuration ..........................................................................................................................29
   5.1 Controller Area Network Driver ......................................................................................................29
      5.1.1 Windows XP Embedded Instructions ..........................................................................................29
   5.2 Enhanced Direct Memory Access Driver ...........................................................................................29
      5.2.1 Windows XP Embedded Instructions ..........................................................................................29
   5.3 WDT - Watchdog Timer .....................................................................................................................29
      5.3.1 Windows XP Embedded Instructions ..........................................................................................29
   5.4 GPIO ..................................................................................................................................................29
      5.4.1 Windows XP Embedded Instructions ..........................................................................................29
   5.5 IEEE 1588 Hardware Assist ..............................................................................................................30
5.5.1 Windows XP Embedded Instructions .......................................................... 30
5.6 Global Configuration Unit and Gigabit Ethernet Driver ............................... 30
5.6.1 Windows XP Embedded Instructions .......................................................... 30
5.7 SMBus .............................................................................................................. 30
5.7.1 Windows XP Embedded Instructions .......................................................... 30
6.0 Pre-boot (BIOS) Firmware .............................................................................. 31
6.1 Pre-boot Firmware Setup Menu ....................................................................... 31
6.1.1 Serial Console Redirection ......................................................................... 32
6.1.2 Changing the Boot Device ......................................................................... 32
6.1.3 Maximum Memory Speed Setup ................................................................... 32
6.1.4 Coherent and Non-Coherent Memory Allocation ......................................... 33
6.1.5 Legacy and AHCI SATA Mode .................................................................... 33
6.2 Pre-boot Firmware Image Reflashing Instructions ............................................ 34
6.2.1 Aptio Flash Update Utility (AFUEFI) ......................................................... 34
7.0 Uninstalling the Software ................................................................................ 36
7.1 Windows XP Embedded Module/Driver Dependencies .................................... 36
8.0 Troubleshooting ................................................................................................. 37
9.0 Glossary ............................................................................................................ 38

Figures
1 Software Package Release Structure - Windows XP Embedded ........................ 8
2 Development Board - Component and Connector Locations ............................ 12
3 Side View of the Board Connectors .................................................................. 13
4 Development Board System Setup ...................................................................... 15
5 Development Machine and Target Device Relationship .................................... 16
6 Importing a .pmq File ..................................................................................... 20
7 Importing an .sld File ....................................................................................... 21
8 Inspecting an .sld File ..................................................................................... 22
9 Adding Components to a Target Device Image ............................................... 24
10 boot.ini File Location .................................................................................... 28

Tables
1 Key Development Board Components and Connectors Legend ..................... 13
2 Pre-boot Firmware Setup Main Menu ............................................................... 31
3 Pre-boot Firmware Setup Program Function Keys ........................................... 31
4 Serial Console Redirection Default Settings .................................................. 32
5 Memory Allocation Settings ............................................................................ 33
6 Windows XP Embedded Module/Driver Dependencies .................................... 36
### Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2008</td>
<td>003</td>
<td>In Section 4.1, “Unpacking the EP80579 Software Package” on page 18, removed note on alternate PHY support. Other updates marked with changebars.</td>
</tr>
<tr>
<td>September 2008</td>
<td>002</td>
<td>Added information for alternate PHY support (see Section 6.1, “Unpacking the EP80579 Software Package” on page 35).</td>
</tr>
<tr>
<td>July 2008</td>
<td>001</td>
<td>Initial release of this document.</td>
</tr>
</tbody>
</table>
1.0 Introduction

1.1 About this Manual

This Getting Started Guide documents the instructions to obtain, build (if necessary), install, and execute the software release package for the Intel® EP80579 Integrated Processor product line. Additionally, this document describes brief installation instructions for the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Board.

Note: The “Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Board” will be referred to as the “development board” in this document.

1.2 Additional Information on Software

The Intel® EP80579 Software Drivers for Embedded Applications on Microsoft* Windows* XP Embedded package has been validated with Windows XP Embedded Service Pack 2.

1.2.1 Where to Find Current Software and Documentation

The software release and associated collateral can be found on the Hardware Design resource center.

1. In a web browser, go to http://www.intel.com/go/soc.
2. For Software and Pre-boot Firmware: Click on “Tools & Software” tab.
3. For Documentation: Click on “Technical Documents” tab.

1.2.2 Product Documentation

The following documentation is provided to support this software release:

- This Getting Started Guide
- Intel® EP80579 Software Drivers for Embedded Applications Release Notes

Note: The Release Notes contains important information about each software release, such as the appropriate firmware version. Please review the Release Notes before proceeding with this document.

Please follow the directions in Section 1.2.1 to locate this collateral.
1.2.3 Pre-boot Firmware

The latest release of the development board pre-boot firmware (BIOS) is also located on Hardware Design resource center. Refer to the Intel® EP80579 Software Drivers for Embedded Applications Release Notes for latest version.

Please follow the directions in Section 1.2.1 to locate this firmware.

1.3 Related Software and Documentation

Refer to the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Kit User’s Guide for information on the development board including board layout, components, connectors, jumpers, headers, power and environmental requirements, and pre-boot firmware.

Please follow the directions in Section 1.2.1 to locate this collateral.

1.4 Conventions

The following conventions are used in this manual:

- Courier font - commands and code examples
- Italic - directory names

1.5 Software Overview

1.5.1 Features Implemented

The software provides the following features:

- IEEE 1588 Hardware Assist Driver
- Controller Area Network (CAN) Hardware Access Driver
- Gigabit Ethernet (GbE) Controller Driver for Network Connectivity
- Advanced Host Controller Interface Software Support for SATA for Native Command Queuing and Hot Plug Capability
- SMBus Driver
- General Purpose I/O (GPIO) Hardware Access Driver
- Watchdog Timer Hardware (WDT) Access Driver

1.5.2 List of Files in Release

The Bill of Materials, sometimes referred to as the BOM, is included as a text file in the released software package. This text file is labeled “filelist” and is located at the top directory level for each release.

1.5.3 Package Release Structure

The package release structure is shown in Figure 1.
Figure 1. Software Package Release Structure - Windows XP Embedded
2.0 Configuration Requirements

2.1 Development Board Configuration

Complete details about the development board can be found in the "Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Kit User’s Guide". This document contains details on the design, structure, and function of all development board features.

To facilitate quick start of the EP80579 software drivers for embedded applications package, relevant sections from the Development Kit User’s Guide have been included in this chapter. Please follow the directions in Section 1.2.1 for information on accessing the full User’s Guide.

2.1.1 Package Components

The Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Kit includes the following:

- Development board containing the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology
- ATX12V power supply
- One DDR2-800 DIMM
- PCIe* graphics card
- SATA hard drive with cable
- SATA DVD-ROM with cable
- Two Controller Area Network cable connectors
- Power Cord (USA power cord supplied)

These items are not supplied from Intel. Please note this is not an exhaustive list of items not supplied.

- Mouse
- Keyboard
- Monitor
- Power supply cord (if country or region-specific power cord is required)

2.1.2 Development Kit Setup

Ensure that all components listed in Section 2.1.1 arrive together. Once all components have been identified and located, installation and setup can begin. This section describes how to set up the development board for operation.

Note: This document assumes that the user is familiar with the basic concepts required to install and configure hardware for a PC system.
2.1.3 Safety

The development board is shipped as an open system allowing for maximum flexibility in changing hardware configurations and peripherals in a lab environment. Since the board is not in a protective chassis, the user is required to take safety precautions in handling and operating the board. Some assembly is required before use.

Ensure a safe and static-free work environment before removing any components from their anti-static packaging. The development board is susceptible to electrostatic discharge that may cause failure or unpredictable operation. The development board must be operated on a flame-retardant surface because a chassis is not included with the board.

Caution: Connecting the wrong cable or reversing a cable may damage the board and may damage the device being connected. Since the board is not in a protective chassis, use caution when connecting cables to the board.

Caution: The power supply cord is the main disconnect device to main power (AC power). The socket outlet should be installed near the equipment and should be readily accessible. To avoid shock, ensure that the power cord is connected to a properly wired and grounded receptacle. Do not connect/disconnect any cables or perform installation/maintenance of the boards in this product during an electrical storm. Ensure that any equipment to which this product will be attached is also connected to properly wired and grounded receptacles.

Note: Ensure that setting up the ATX power supply is the final step performed in the process of assembly.

2.1.4 Connecting the Serial ATA Hard Drive and Cable

The development board provides two Serial ATA (SATA) connectors. Connect cables to the appropriate drive sequentially, starting from Port 0 to Port 1. See Figure 2 and Table 1 for the location and identification of the SATA connectors.

Note: Intel recommends connecting the boot drive to SATA port 0.

2.1.5 Connecting the Keyboard and Mouse

Connect a PS/2 mouse and keyboard to the stacked PS/2 connector on the rear panel of the board. The bottom connector is the keyboard connector and the top connector is the mouse connector. Alternatively, a USB keyboard and a USB mouse can be connected to the development board’s USB connectors.

Note: Mouse and keyboard are not supplied by Intel.

Note: The serial redirection feature can be enabled to remotely access the board through a serial cable without attaching a keyboard or mouse to the development board. Refer to the “Connecting the Serial Cable for Console Redirection” section of the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Kit User’s Guide for more information.

2.1.6 Connecting the PCI Express Video Card

Populate the PCIe graphics card in any one of the PCIe slots.
2.1.7 Connecting the Serial ATA DVD-ROM Drive (Optional)

Connect the Serial ATA DVD-ROM drive to SATA Port 1 utilizing the cable that comes with the DVD-ROM drive. See Figure 2 and Table 1 for the location and identification of the SATA connectors.

2.1.8 Connecting the Power Cables

Use the following procedure to connect the power cables:

1. The board supports the use of ATX12V power supplies with either 2 x 10 or 2 x 12 main power cables.

2. Plug the main connector into the board. Ensure that the plug clip lines up with the clip lock and the connector pins easily fit into their appropriate slots. When using a power supply with a 2 x 10 main power cable, attach that cable to the right-most part of the main power connector, leaving pins 11, 12, 23 and 24 unconnected.

3. Plug in the power connectors from each of the SATA drives and disk drives.

2.1.9 Powering Up the System

**Warning:** Ensure the steps in the previous sections were strictly followed before powering up the system.

Use the following procedures to power up the development board:

1. Ensure that the processor heat sink and the fan are installed according to the procedure in the “Connecting the Processor Heatsink and Fan” section of the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Board User’s Guide.

2. Leaving the On/Off switch in the OFF position, plug the power cable into the back of the power supply.

3. Once the board is set up, plug the cord into the power source.

4. Switch on the power supply.

5. Press the power-on button to start the system. Refer to Figure 2 for the location power-on button (item 1, lower-left). Table 1 is a legend for key items labeled in Figure 2.
Figure 2. Development Board - Component and Connector Locations
Figure 3. Side View of the Board Connectors

![Side View of the Board Connectors](image)

Table 1. Key Development Board Components and Connectors Legend

<table>
<thead>
<tr>
<th>Callout</th>
<th>Component/Connector</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Intel® EP80579 Integrated Processor</td>
</tr>
<tr>
<td>B</td>
<td>PEX PCIe Switch Chip</td>
</tr>
<tr>
<td>C</td>
<td>Marvell 8811E1 Quad PHY</td>
</tr>
<tr>
<td>D</td>
<td>Super IO Controller</td>
</tr>
<tr>
<td>E</td>
<td>FPGA</td>
</tr>
<tr>
<td>F</td>
<td>Flash memory 0</td>
</tr>
<tr>
<td>G</td>
<td>Flash memory 1</td>
</tr>
<tr>
<td>H</td>
<td>FWH</td>
</tr>
<tr>
<td>I</td>
<td>Power button</td>
</tr>
<tr>
<td>J</td>
<td>Reset button</td>
</tr>
<tr>
<td>K</td>
<td>Sleep button</td>
</tr>
<tr>
<td>L</td>
<td>PCIe Wake button</td>
</tr>
<tr>
<td>M</td>
<td>Port 80 IC</td>
</tr>
<tr>
<td>N</td>
<td>CMOS battery</td>
</tr>
<tr>
<td>O</td>
<td>On-board speaker</td>
</tr>
<tr>
<td>P</td>
<td>CPU FAN connector</td>
</tr>
<tr>
<td>Q</td>
<td>AUX FAN connector</td>
</tr>
<tr>
<td>R</td>
<td>AUX0 FAN connector</td>
</tr>
<tr>
<td>S</td>
<td>AUX1 FAN connector</td>
</tr>
<tr>
<td>T</td>
<td>ATX power connector</td>
</tr>
<tr>
<td>U</td>
<td>Two 7-segment display (Port 80)</td>
</tr>
<tr>
<td>V</td>
<td>SATA port 0</td>
</tr>
<tr>
<td>W</td>
<td>SATA port 1</td>
</tr>
</tbody>
</table>
Table 1. Key Development Board Components and Connectors Legend

<table>
<thead>
<tr>
<th>Callout</th>
<th>Component/Connector</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>DDR2 DIMM0</td>
</tr>
<tr>
<td>Y</td>
<td>DDR2 DIMM1</td>
</tr>
<tr>
<td>Z</td>
<td>Slot 0 x8 connector 4 lanes PCI Express</td>
</tr>
<tr>
<td>AA</td>
<td>Slot 1 x4 connector 1 lane PCI Express</td>
</tr>
<tr>
<td>BB</td>
<td>Slot 2 x4 connector 1 lane PCI Express</td>
</tr>
<tr>
<td>CC</td>
<td>Slot 3 x4 connector 1 lane PCI Express</td>
</tr>
<tr>
<td>DD</td>
<td>Slot 4 x4 connector 1 lane PCI Express</td>
</tr>
<tr>
<td>EE</td>
<td>Mezzanine connector 0</td>
</tr>
<tr>
<td>FF</td>
<td>Mezzanine connector 1</td>
</tr>
<tr>
<td>GG</td>
<td>Floppy Connector</td>
</tr>
<tr>
<td>HH</td>
<td>CF connector</td>
</tr>
<tr>
<td>II</td>
<td>ITP-XDP connector</td>
</tr>
<tr>
<td>LL</td>
<td>Mezzanine connector 3</td>
</tr>
<tr>
<td>MM</td>
<td>Parallel port</td>
</tr>
<tr>
<td>NN</td>
<td>COM1</td>
</tr>
<tr>
<td>OO</td>
<td>COM2</td>
</tr>
<tr>
<td>PP</td>
<td>PS/2 mouse (top)/keyboard (bottom)</td>
</tr>
<tr>
<td>QQ</td>
<td>USB port 0</td>
</tr>
<tr>
<td>RR</td>
<td>USB port 1</td>
</tr>
<tr>
<td>SS</td>
<td>RJ-45 Ethernet port 0</td>
</tr>
<tr>
<td>TT</td>
<td>RJ-45 Ethernet port 1</td>
</tr>
<tr>
<td>UU</td>
<td>RJ-45 Ethernet port 2</td>
</tr>
</tbody>
</table>

Note: Items JJ and KK are not shown in Figure 2 or Figure 3 and are intentionally omitted.
2.2 Development Board Setup Requirements

2.2.1 Stand-alone Target System

Figure 4 shows the system setup when the target development board is also used for build and install.

Figure 4. Development Board System Setup
3.0 System Requirements for Installing an OS on a Development Board

3.1 Installing Windows XP Embedded

The installation process for Microsoft® Windows® XP Embedded is different than the more traditional OSs such as Linux or even Windows XP Pro. The installation process requires the user to create an OS image on a development machine and subsequently install it on the target embedded device. The following subsections provide some explanation of the Windows XP Embedded image creation and installation process. For detailed instructions on the Windows XP Embedded Toolset, please refer to the online help provided with these tools.

A Windows XP Embedded image is created on a Development Machine, a standard PC. The image is transferred to the Target Device. Figure 5 shows this relationship, as well as listing where the XP Embedded development tools are installed and used.

Figure 5. Development Machine and Target Device Relationship

3.1.1 Build Environment Requirements

Consult the Microsoft web site or the Windows XP Embedded product for minimum system requirements. The following is a link to the Windows XP Embedded system requirements:

3.1.2 Windows XP Embedded Development Toolset

Please acquire a distribution of Windows XP Embedded from Microsoft or your distributor of choice. An evaluation copy of Windows XP Embedded provides access to all the Windows XP Embedded development tools needed and referenced in this guide, however these tools can be licensed on a permanent basis from Microsoft or one of their distributors. The evaluation copy of Windows XP Embedded allows the run-time images of Windows XP Embedded to be created and deployed to an embedded device for a limited time period of 120 days. For permanent installation of a runtime image, a run-time license must be purchased.

Install the Windows XP Embedded toolset on a development machine running Windows XP Pro.

Note: Be sure to install the updates on disks 3 and 4 of the Windows XP Embedded toolset to bring the toolset into compliance with SP2.

The toolset includes:

- Target Analyzer - Analyzes target system, used to detect hardware components
- Component Designer - Create custom components
- Target Designer - Build custom operating system images
- Component Database Manager - Manage component database and repositories

Several file types will be created during the Windows XP Embedded image creation process. Here is an explanation of these file types and extensions.

- .pmq file extension - Exact definition of “pmq” is unknown. Windows XP Embedded file created by Target Analyzer and imported to Component Designer. File lists Windows XP Embedded components the developer can use to build a Windows XP Embedded image supporting the target hardware. See the following link for more information and search for “pmq”:
- .sld file extension - Exact definition of “sld” is system level definition. Windows XP Embedded System Level Definition file created by Component Design and imported to Component Database Manager. The file contains component-level definition information. For more information see the following link:
- .slx file extension - Exact definition of “slx” is unknown. Windows XP Embedded configuration file created by Target Design to be used to compile a Windows XP Embedded image.

A trial version of the Microsoft Windows XP Embedded operating system can be downloaded from Microsoft at the following location:

http://www.microsoft.com/windows/embedded/eval/trial.mspx

Using Windows XP Embedded tools and developing Windows XP Embedded images can seem a complicated task. This document provides instructional steps to creating a Windows XP Embedded image for use on the development board. However, multiple books are available for reference from online book outlets such as Amazon.com, or reference support can be found on the Microsoft Software Developer Network. Two books available from Amazon.com are:

- Windows XP Embedded Step by Step - author James Beau Cseri
- Windows XP Embedded Advanced - author Sean D. Liming

§ §
4.0 Configuring and Building an XP Embedded Image

The creation of a Microsoft® Windows® XP Embedded image requires a development machine with the Windows XP Embedded OS tools installed (as described in Chapter 3.0). Only one tool is used to analyze the target device hardware, that is the Target Analyzer (see Section 4.2).

The configuration and building of the target device Windows XP Embedded image is completed on the development machine with the Windows XP Embedded OS tools described earlier. The steps to configure and build a Windows XP Embedded are also described in the subsections following.

4.1 Unpacking the EP80579 Software Package

The Intel® EP80579 Software Drivers for Embedded Applications package comes in the form of a .zip. See Section 1.2.1, “Where to Find Current Software and Documentation” on page 6 for the software location. The package can be unpacked at any location on the development system, but for the purposes of this Getting Started Guide, a recommendation is provided.

On the development system, create a directory in the C:\ directory called “EP80579_release” with either Windows Explorer methods or with the following commands in a command prompt window:

```
cd c:\
mkdir EP80579_release
cd EP80579_release
```

Place the XP Embedded package .zip file in the EP80579_release directory. Extract the contents of the .zip file using a program called Winzip. The following are suggested steps to unzipping the EP80579 software drivers for embedded applications package with Winzip:

1. Double-click the Windows XP Embedded .zip file to open in Winzip
2. Go to options->configuration menu
3. Click on the Miscellaneous tab
4. Make sure “TAR file smart CR/LF conversion” is unchecked
5. Click OK

4.2 Target Analyzer - Analyzing Target Device

A utility called Target Analyzer or tap.exe is included with the Windows XP Embedded toolset. This tool runs within the Windows XP Pro operating system on the target machine, in this case the development board.

Note: When installing Windows XP Pro on the target hardware, the BIOS must be set to Legacy mode, not AHCI mode. During boot of the Windows XP Embedded image, set the BIOS to AHCI mode. Refer to Section 6.1.5 for directions on toggling between Legacy and AHCI mode.
1. Install Windows XP Pro on the target hardware, creating an additional, second partition while installing Windows XP Pro. The second partition will be used later during deployment of the Windows XP Embedded image that shall be created on the development machine.

2. Transfer the tap.exe executable to the target platform. Typical installation is at the location: \Program Files\Windows Embedded\utilities. It may be transferred in any way, USB Memory Stick, CDROM, or Network transfer.

For instructions on installing Windows XP Pro to a new hard disk and instructions on creating partitions, see the Microsoft documentation at the following links:

http://support.microsoft.com/kb/316941
http://support.microsoft.com/kb/313348

3. Boot the Target System to Windows XP Pro. Open a command shell window and change directory to the location of tap.exe.

4. Execute TAP, which creates a hardware profile of the Target System, defining a filename for the output, with the following command:

\tap /o EP80579_tap.pmq > EP80579_tap.log

The analysis of the target device’s hardware is captured in the file specified on the command line, in this case EP80579_tap.pmq.

5. Exit the command shell, and transfer a copy of the EP80579_tap.pmq file to the development machine running the Windows XP Embedded toolset. The EP80579_tap.pmq file will be imported into Component Designer on the development machine.

4.3 Import a Target Device .pmq File

Microsoft Component Designer is the Windows XP Embedded tool used to componentize drivers. It is also used to import .pmq files, Target Analyzer files of the target hardware. The output of Component Designer are .sld file types.

The instructions that follow essentially componentize the analysis of the development board, the EP80579_tap.pmq file, converting it to the EP80579_tap.sld format.

1. Save the EP80579_tap.pmq file to a USB memory stick on the development board.
2. Take the USB memory stick to the development machine that is running the Windows XP Embedded tools.
3. Open Component Designer on the development machine.
5. Browse for and import the EP80579_tap.pmq file (from the location to which it was copied on the development machine or possibly from the USB memory stick).
6. Click the Start button on the Import Dialog (see Figure 6). Optionally, you can enter a Tap Import Log file to be generated.

Note: It is recommended to create a log file of the import process should it be necessary to debug errors. The import process creates a new file called EP80579_tap.sld. An .sld file is an encapsulation of XP Embedded components, and in this case the EP80579_tap.sld file is an encapsulation of the XP Embedded components that match those that were analyzed on the target device.

7. Click the Close button on the Import Dialog.
8. Choose File -> Save and save the EP80579_tap.sld file to a convenient location where it can be found easily later.
4.4 Importing the Embedded Software Release .sld File

The EP80579 software drivers for embedded applications package includes the embedded_drivers_XPe_components.sld file. This .sld file is an encapsulation of the drivers developed for the EP80579 integrated processor and componentized for Windows XP Embedded. Intel has consolidated the EP80579 integrated processor drivers into one SLD file for direct installation.

The Component Database Manager stores all components distributed with the Windows XP Embedded OS and is also used to store custom-created components. Import both the embedded_drivers_XPe_components.sld and the EP80579_tap.sld file into the Component Database Manager as follows:

1. Open the Component Database Manager on the development machine.
2. Click Import... on the Database tab (see Figure 7).
3. Navigate to the embedded_drivers_XPe_components.sld file (typically, C:\EP80579_release\Embedded\EP80579_tap.pmq) and click Open.
4. Click Import to import the EP80579 integrated processor components into the repository.
5. Click the import SLD dialog box and the process is complete.
6. Repeat steps 2 to 5 with the EP80579_tap.sld file.
7. Click the Close button on the Import SLD Dialog box.
8. Close the Component Database Manager.
4.5 Inspecting the Intel® EP80579 Software Drivers for Embedded Applications Release .sld File

Open and inspect the embedded_drivers_XPe_component.sld file to become familiar with all the componentized software that Intel is providing for the EP80579 integrated processor. These steps are not part of the image creation process, but only serve to help give a high-level explanation of the EP80579 software drivers for embedded applications release .sld file.

1. Open the Component Designer.
2. Choose File -> Open.
3. Browse to and select embedded_drivers_XPe_component.sld.
4. Within Component Designer, click on embedded_drivers_XPe_component.sld in the left pane (see Figure 8).

Notice that the .sld file has several components, repositories, and a package. Most of these components constitute drivers specifically for the EP80579 integrated processor that have been componentized into this .sld file. The components, repositories, and package structure is a structure that Windows XP Embedded functions use to build in new features.
The purpose of this section is to familiarize the user with the content of the .sdl file that Intel distributes and the support provided in it.

**Figure 8. Inspecting an .sld File**

![Image of Inspecting an .sld File]

### 4.6 Creating a Target Device Image

Use the Target Designer utility to create a target device image.

1. Open Target Designer.
2. Choose File -> New.
3. Label the Configuration Name as prompted to something significant, such as the name of the device with the date. For the purposes of this Getting Started Guide, the Configuration Name label is “TDP”.
4. Click OK.
5. Choose File -> Save.

   Upon saving, an .slx file is created using the label selected for the Configuration Name. An .slx file is the runtime configuration for an embedded device, in this case, the TDP embedded device.

Add the component configuration imported previously from running TAP.exe on the target device, EP80579_tap.pqm.
6. Double click the **EP80579_tap** component in the upper-left most component window pane in the Target Designer.

7. Double-click each of the components listed below to add the component from the import process of the embedded_drivers_XPe_components.sld. These components are found in the upper-left most component window pane in the Target Designer under the Hardware, System devices, IDE ATA/ATAPI controllers, Universal Serial Bus controllers, and Network adapters directories respectively. The act of double-clicking each component adds the component to the TDP.slx image to be built.

The components listed below can be found by searching using a partial name in the Search box in the upper left of Target Design window. The components are found by expanding Hardware, Devices, then either System Devices, IDE ATA/ATAPI controllers, Universal Serial Bus controllers, and Network Adapters.

**Tip:** An easy way to find all the components listed above, except the EP80579_tap component created from the tap process, is to create a filter for the package name. This can be done by:

a. Click on the button with the funnel symbol.

b. Click New.

c. Name the filter (i.e. EP80579).

d. Scroll down the list and select (double-click) on “Component belongs to the following package: [package]”. Selecting this adds this choice to the bottom pane.

e. Double-click it from the bottom pane.

f. Select “Intel(R) EP80579 Embedded Drivers” in the pop-up window.

g. Click the OK button.

h. Apply filter.

- **System devices:**
  - Controller Area Network for Intel(R) EP80579 Integrated Processor Product Line
  - Global Configuration Unit for Intel(R) EP80579 Integrated Processor Product Line - 503e
  - GPIO for Intel(R) EP80579 Integrated Processor Product Line
  - IEEE 1588* Time Sync for Intel(R) EP80579 Integrated Processor Product Line - 503c
  - SMBus Controller for Intel(R) EP80579 Integrated Processor Product Line - 5032
  - System Devices for Intel(R) EP80579 Integrated Processor Product Line
  - WatchDog Timer for Intel(R) EP80579 Integrated Processor Product Line
  - Windows Driver Foundation (WDF) library

- **Network adapters**
  - Gigabit Ethernet for Intel(R) EP80579 Integrated Processor Product Line

- **IDE ATA/ATAPI controllers**
  - SATA AHCI for Intel(R) EP80579 Integrated Processor Product Line - 5029
  - SATA IDE for Intel(R) EP80579 Integrated Processor Product Line - 5028

- **Universal Serial Bus controllers**
  - USB Host Controller for Intel(R) EP80579 Integrated Processor Product Line
Add the following components for convenience.

8. Search for the following components by name in the Search box in the upper left corner of Target Designer window. Select “All Components” from the Filter drop-down list. Type each component named in the bullet list below and press the “component search” button (that is, the magnifying glass over the cube at the end of the search string). Then, add each component to the image to be built using the same process used to add the EP80579 integrated processor components described above.

The act of double-clicking each of the components adds them to the TDP.slx image to be built.

- User Interface Core
- TCP/IP Utilities
- Windows Accessories
- Accessories/Communications Programs
- Accessories/System Tools
- Task Manager
- System Control Panel
• Device Manager
• Explorer Application
• Legacy Shell Application Support
• Network Setup Wizard
• Registry Editor
• Safely Remove Hardware Program
• Users Control Panel
• Add Hardware Control Panel
• Add/Remove Programs Control Panel
• Display Control Panel
• PS/2 Compatible Mouse
• Standard 101/102-key or Microsoft Natural PS/2 Keyboard
• Map Network Drives/Network Places Wizard

After adding the User Interface Core component, modify its settings.

9. Select the User Interface Core component in the center pane.
10. Expand the settings for this component.
11. Under the settings for this component, select programs to appear on the Start Menu (for example, All Programs, Control Panel, Run).
12. Select all desired programs.

Dual boot will be used to deploy the Windows XP Embedded image, so configure the boot settings within the master settings of the TDP.slx file.

13. In the center pane, expand the TDP.slx file if not already expanded.
14. Click on “settings”.
15. In the top-right pane, click “show” under the heading Target Device Settings to expand the Target Device Settings subsection.
16. Change the following settings:
   • Boot drive - D:
   • Windows folder - D:\Windows
   • Program Files folder - D:\Program Files
   • Documents and Settings folder - D:\Documents and Settings
   • Boot ARC path - multi(0)disk(0)rdisk(0)partition(2)
   • Boot partition size (MB) - 1024
   • Partition cluster size (bytes) - 4096

Save this configuration, labeled TDP.slx.
17. Choose File -> Save.
18. Choose Configuration in the upper toolbar.
19. Choose Check Dependencies in the pull-down menu. The dependency check takes a few minutes and most dependencies are auto-resolved, but some need some additional selections.

The resulting output is likely to produce some dependencies for some additional components that need to be selected. The output is located in the bottom pane.
20. Resolve by double-clicking on each one and choosing all selections if possible, or just choose one selection where constrained to choose one selection and click the Add button.

Common dependencies and their selections are listed below:

- **Component: “ACPI Uniprocessor PC...”**
  - EWF NTLD...  
  - NT Loader...(the best selection for development board image creation)
- **Component: “ACPI Uniprocessor PC...”**
  - CDFS...
  - FAT...
  - NTFS...
  - UDFS...
- **Component: “Compression and Expansion Tools...”**
  - CDFS...
  - FAT...
  - NTFS...
  - UDFS...
- **Component: “Regional and Language Options...”**
  - English Language Support...
- **Component: “Session Manager...”**
  - Windows Logon (Standard)...
- **Component: ”User Interface Core...”**
  - FAT Format...
  - NTFS Format...

21. Repeat Check Dependencies until there are no additional dependencies to resolve.
22. Save the configuration.

Execute a build of the image.
23. Choose Configuration in the upper tool bar again.
24. Choose Build Target Image.... in the pull-down menu.
25. If prompted to check dependencies again, do so.
26. Click the Build button.

The build is saved to a directory called: %Windows Embedded Images%.

### 4.7 Transferring Windows XP Embedded Image to Target Device

There are a variety of ways to deploy a Windows XP Embedded Image to a target device:

- Dual Boot
- CD-ROM
- Compact Flash
• Remote Network Booting
• USB 2.0 Boot (Feature Pack 2007)

For the purposes of this Getting Started Guide, the “dual boot” deployment method is described.

4.7.1 Dual Boot Windows XP Embedded Image Deployment

In the Section 4.2, “Target Analyzer - Analyzing Target Device” on page 18, instructions are given to install Windows XP Pro on the target device including a second partition on the hard drive during Windows XP Pro installation. The second partition on the hard drive is used in the following instructions to deploy the Windows XP Embedded target device image.

The image files (folders and files) created during the image creation process were deposited into the \Windows Embedded Images\ folder on the development machine.

1. Copy all file and folders within the \Windows Embedded Images\ (not including the \Windows Embedded Images\ folder itself) to the target device, the development board into the root of the second partition created during the installation of Windows XP Pro on the development board.

Note: It is recommended to make a backup of the Boot.ini file before making any of the changes described below. The Boot.ini file may be a hidden file in C:\. When in the folder, click Tools, then Folder Options..., then View. Ensure that these configuration are enabled or disabled as instructed:
• Show hidden file and folder - enabled
• Hide extensions for known file types - unchecked
• Hide protected operating system files - unchecked

Modify the Boot.ini file to enable dual-boot. Open the Boot.ini file found at the root directory C:\ on the development board. Modify the Boot.ini file found for the install of Windows XP Pro on the development board because it is the primary operating system install.

Note: Do not modify the Boot.ini file of the image that will be created.

Modify the Boot.ini file to look like the following:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(2)\WINDOWS

[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Professional" /noexecute=allwaysoff /fastdetect
multi(0)disk(0)rdisk(0)partition(2)\WINDOWS="Microsoft Windows XP Embedded" /noexecute=allwaysoff /fastdetect
```

2. Right-click on the Boot.ini file in the root directory C:\ and choose Properties.
3. Deselect read-only if the file is in read-only mode.
4. Open the Boot.ini file with Notepad.
5. Make the additions described above.
6. Save and close the Boot.ini file.
7. Upon reboot, change the BIOS to AHCI under the IDE settings of the BIOS Setup Menu.
8. When boot options are displayed upon reboot, select Microsoft Windows XP Embedded.

The Windows XP Embedded image always needs the BIOS to be booted in AHCI mode since the AHCI component is built into the image. No existing lines were modified and only the last line was added to the Boot.ini file. System Boot.ini file may contain "redirect=usebiossettings" listed with each operating system listed in the file. This setting is used to enable Emergency Management Services (EMC) console redirection on a Windows server.

Note: If a previous Windows XP Embedded Image was deployed to this partition, perform a quick format to remove files prior to new Windows XP Embedded Image deployment.

Figure 10. boot.ini File Location
5.0  Runtime Configuration

This chapter describes how to compile and install EP80579 drivers individually.

5.1  Controller Area Network Driver

5.1.1  Windows XP Embedded Instructions

No run-time configuration is necessary for the Controller Area Network (CAN) driver. Once the component is imported and built into the target designed image, the driver will be running in the Microsoft* Windows* XP Embedded image. The driver waits for instruction from a client-designed application through use of the document APIs.

A client-designed application is not supplied by Intel. However, the Intel® EP80579 Software Drivers for Embedded Applications Programmer’s Guide and API Reference Manual provides details about the APIs so that a client application can be developed.

5.2  Enhanced Direct Memory Access Driver

5.2.1  Windows XP Embedded Instructions

There is no Enhanced Direct Memory Access (EDMA) driver available for Windows XP Embedded.

5.3  WDT - Watchdog Timer

5.3.1  Windows XP Embedded Instructions

No run-time configuration is necessary for the Watchdog Timer driver. Once the component is imported and built into the target designed image, the driver will be running in the Windows XP Embedded image. The driver waits for instruction from a client-designed application through use of the document APIs.

A client-designed application is not supplied by Intel. However, the Intel® EP80579 Software Drivers for Embedded Applications Programmer’s Guide and API Reference Manual provides details about the APIs so that a client application can be developed.

5.4  GPIO

5.4.1  Windows XP Embedded Instructions

No run-time configuration is necessary for the General Purpose I/O (GPIO) driver. Once the component is imported and built into the target designed image, the driver will be running in the Windows XP Embedded image. The driver waits for instruction from a client-designed application through use of the document APIs.
A client-designed application is not supplied by Intel. However, the Intel® EP80579 Software Drivers for Embedded Applications Programmer’s Guide and API Reference Manual provides details about the APIs so that a client application can be developed.

5.5 IEEE 1588 Hardware Assist

5.5.1 Windows XP Embedded Instructions

No run-time configuration is necessary for the IEEE 1588 Hardware Assist driver. Once the component is imported and built into the target designed image, the driver will be running in the Windows XP Embedded image. The driver waits for instruction from a client-designed application through use of the document APIs.

A client-designed application is not supplied by Intel, however the Intel® EP80579 Software Drivers for Embedded Applications Programmer’s Guide and API Reference Manual provides details about the APIs so that a client application can be developed.

5.6 Global Configuration Unit and Gigabit Ethernet Driver

Two drivers complete the software features set for network connectivity on the EP80579 integrated Gigabit Ethernet controllers; the Global Configuration Unit (GCU) driver and the Gigabit Ethernet (GbE) driver. The GCU driver controls the MAC and administrative activities. The GbE driver controls the network connectivity. The GbE driver is dependent on the GCU driver.

Note: The Global Configuration Unit driver must be installed prior to installation of the Gigabit Ethernet driver.

5.6.1 Windows XP Embedded Instructions

Run-time configuration for the Gigabit Ethernet driver can be accomplished through Windows Control Panel Device Manager similar to configuration of any Windows device driver.

Note: For information on porting the GbE driver to other PHYs, see the Software for Intel® EP80579 Integrated Processor Product Line PHY Porting Guide.

5.7 SMBus

5.7.1 Windows XP Embedded Instructions

No run-time configuration is necessary for the SMBus driver. Once the component is imported and built into the target designed image, the driver will be running in the Windows XP Embedded image. The driver waits for instructions from a client-designed application through use of the document APIs.

A client-designed application is not supplied by Intel. However, the Intel® EP80579 Software Drivers for Embedded Applications Programmer’s Guide and API Reference Manual provides details about the APIs so that a client application can be developed.
6.0 Pre-boot (BIOS) Firmware

The pre-boot firmware is executed when the system is powered up or reset. It initializes and configures system memory, devices and buses/interfaces.

The pre-boot firmware is based on the AMI Aptio* 4.5 core and compliant to EFI v1.1. The firmware is stored in the Firmware Hub (FWH) or SPI (Serial Peripheral Interface) Flash; the FWH or SPI Flash can be updated using a flash utility tool that is provided by Intel or by using a floppy drive connected to the floppy header.

The pre-boot firmware setup menu can be used to view and modify the system settings for the development board. The setup menu is accessed by pressing the <Del> key during pre-boot firmware boot up (before the operating system boot begins). The setup menu bar is shown in Table 2.

6.1 Pre-boot Firmware Setup Menu

Table 2 shows the pre-boot firmware setup main menu and provides a brief description of each menu option. Table 3 shows the function keys that can be used when navigating and selecting options from pre-boot firmware menus.

Table 2. Pre-boot Firmware Setup Main Menu

<table>
<thead>
<tr>
<th>Main</th>
<th>Advanced</th>
<th>Chipset</th>
<th>Security</th>
<th>Boot</th>
<th>Exit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Displays processor and memory configuration Setup for CMOS system date and time</td>
<td>Configures advanced features and settings</td>
<td>Configures different major components</td>
<td>Setup passwords and security features</td>
<td>Selects boot options and configurations</td>
<td>Saves or discards changes to setup program options</td>
</tr>
</tbody>
</table>

Table 3. Pre-boot Firmware Setup Program Function Keys

<table>
<thead>
<tr>
<th>Function Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; or &gt;</td>
<td>Moves cursor left or right in the main menu</td>
</tr>
<tr>
<td>^ or v</td>
<td>Moves cursor up or down to select sub-menu items</td>
</tr>
<tr>
<td>Enter</td>
<td>Executes command or selects the submenu</td>
</tr>
<tr>
<td>F7</td>
<td>Discard changes</td>
</tr>
<tr>
<td>F8</td>
<td>Load the fail-safe default</td>
</tr>
<tr>
<td>F9</td>
<td>Load the optimal default configuration for the current menu</td>
</tr>
<tr>
<td>F10</td>
<td>Save the current configuration and exit the setup menu</td>
</tr>
<tr>
<td>ESC</td>
<td>Exit the setup menu</td>
</tr>
</tbody>
</table>
6.1.1 Serial Console Redirection

The pre-boot firmware supports redirection of both video and keyboard via a serial port. When console redirection is enabled, the remote console terminal sends keystrokes to the Intel® EP80579 Development Board pre-boot firmware and the pre-boot firmware redirects the video to the console terminal.

As an option, the Intel® EP80579 Development Board can be operated without keyboard or video and can run entirely via the remote serial console. This includes accessing the pre-boot firmware setup menu.

Console redirection ends when operating system boot up begins. After boot up begins, the operating system is responsible for continuing the redirection.

Note: pre-boot firmware console redirection is text only. Graphical data, such as logos, are not redirected.

Table 4 shows the default settings of the serial console redirection.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port Number</td>
<td>COM 1</td>
</tr>
<tr>
<td>Baud Rate</td>
<td>115200</td>
</tr>
<tr>
<td>Data Bits</td>
<td>8</td>
</tr>
<tr>
<td>Parity</td>
<td>None</td>
</tr>
<tr>
<td>Stop bits</td>
<td>1</td>
</tr>
<tr>
<td>Flow Control</td>
<td>None</td>
</tr>
</tbody>
</table>

6.1.2 Changing the Boot Device

Use the following procedure to change the boot device:

1. Press the <Del> key during POST to enter the pre-boot firmware setup menu.
2. Use the arrow keys to navigate to the <BOOT> menu.
3. Move the cursor to <Boot Device Priority>.
4. Select the desired booting sequence list.

Note: Follow the instructions on the right side of the pre-boot firmware screen to navigate and change pre-boot firmware settings.

6.1.3 Maximum Memory Speed Setup

The maximum memory speed supported on the development board can be selected using the Maximum Memory Speed Setup option available in the BIOS Setup Menu on the Chipset tab.

Enter the BIOS Setup Menu and select the Chipset tab. Select North bridge, and navigate down to the bottom option, titled Max Memory Speed Support. Select this option using the Enter button. A selection box appears providing the following options:

- 400 MHz
- 533 MHz
- 667 MHz
- 800 MHz
The default setting in the BIOS is 400 MHz. If a higher speed memory DIMM is inserted into the development board, the corresponding memory speed must be selected in the BIOS Setup Menu to support the intended speed. Otherwise, the memory is reduced to the default of 400 MHz.

6.1.4 Coherent and Non-Coherent Memory Allocation

The development board supports allocation of memory regions for coherent and non-coherent use. Coherent and non-coherent memory use features are for development boards that use the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology. Intel® EP80579 Integrated Processors (without Intel® QuickAssist Technology) do not make use of the memory set aside for these features. Therefore, it is recommended that the settings for coherent and non-coherent usage be set to zero so that no memory is allocated to these features.

Enter the BIOS Setup Menu and select the Chipset tab. Select North bridge, and navigate down towards the bottom to the Coherent Mem Size option and press Enter to select this option. A dialog box is displayed prompting the user to enter a value. Type the numerical value zero, “0”, and press Enter. Navigate to the next option, Non-Coherent Mem Size, and press Enter to select this option. A dialog box is displayed prompting the user to enter a value. Again, type the numerical value zero, “0”, and press enter. These selections override the BIOS default settings and allocate no memory regions for these two features.

Note: This software package requires the pre-boot firmware (BIOS) for your hardware to allocate the values for each region called out in Table 5. For more information on these regions, refer to the Intel® EP80579 Integrated Processor Product Line Datasheet, Section 3.0.

Table 5. Memory Allocation Settings

<table>
<thead>
<tr>
<th>Datasheet name</th>
<th>Software name</th>
<th>Region Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA/ASU Shared (Coherent)</td>
<td>CDRAM</td>
<td>0</td>
</tr>
<tr>
<td>IA/ASU Shared (AIOC-Direct)</td>
<td>NCDRAM</td>
<td>0</td>
</tr>
</tbody>
</table>

6.1.5 Legacy and AHCI SATA Mode

The development board supports hard drives in legacy SATA mode and in Advanced Host Controller Interface (AHCI) mode. AHCI mode provides advanced capabilities and improved performance, provided the hard drive supports the following features:

- Hot Plug
- Native Command Queuing
- Speeds up to 3 Gb/s

Refer to the Serial ATA Organization web site for more information:

http://www.serialata.org/

The development board pre-boot firmware (BIOS) can be configured in either Legacy or AHCI mode as desired. The BIOS defaults to Legacy mode because not all hard drives support AHCI. To toggle the BIOS to either Legacy or AHCI mode, proceed as follows:

1. Press the <Del> key during POST to enter the pre-boot firmware setup menu.
2. Use the arrow keys to navigate to the Advanced menu.
3. Use the arrow keys to navigate to the IDE Configuration option.
4. Select the IDE Configuration option.
5. Use the arrow keys to navigate to the SATA Mode option.
6. Press the Enter key. A SATA Mode popup window appears.
7. Select either Legacy or AHCI as desired. Do **not** use Native as a selection.
8. Press F4 to save.
9. Choose Yes. The system continues the boot process.

### 6.2 Pre-boot Firmware Image Reflashing Instructions

One method is available for updating the pre-boot firmware flash images located on the development board firmware hub:

- **AFUEFI Flash Recovery**

It is possible that updated pre-boot firmware images will become available from Intel for the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Board. The latest pre-boot firmware image is available from Intel's public web site, http://www.intel.com/go/soc located with all other collateral related to the EP80579 integrated processor.

If the pre-boot firmware image should become corrupted on the board, also utilize these instructions to reflash the image to the firmware hub.

#### 6.2.1 Aptio Flash Update Utility (AFUEFI)

Use the following instructions to update the development board pre-boot firmware image using a USB memory stick and the Aptio Flash Update Utility from AMI.

**Necessary Hardware:**

- development board
- Socketed Firmware Hub
- USB Memory Stick

**Necessary Software:**

- development board pre-boot firmware Image
- Aptio Flash Update Utility - AFUEFI

**Steps to Reflash Image:**

1. Load the AFUEFI utility onto the USB memory stick.
2. Load the pre-boot firmware image onto the USB memory stick.
3. Boot the development board to the EFI shell. Change the boot setting in the BIOS Setup Selection to boot from the EFI shell if needed.
4. Insert the USB memory stick into the USB port.
5. Once the USB memory stick is recognized on the system (activity seen on the USB memory stick), several commands are available as follows:
   - Type “map -r” to list all devices available.
   - Type “fs0:” to enter USB device.
   - Type “ls” to list all files.
6. Once the “fs0:” command has been initiated, execute the AFUEFI utility.
   — Type “AFUEFI <pre-boot firmware image name> /X /P /B /N”
     (the <pre-boot firmware image name> will be similar to TRXTG055.ROM)
7. Reboot the development board once reflashing has completed.
8. Confirm the image has been updated to the reflashed image by looking in BIOS Setup.
7.0 Uninstalling the Software

Microsoft* Windows* XP Embedded images can be built to look like a Windows XP Pro image with any number of convenient components built in, and an Windows XP Embedded image can be built in a truly embedded spirit, with no access to change the image configuration during run-time. Device Manager capability is required in a Windows XP Embedded image to uninstall an Windows XP Embedded driver. Find the system feature in Device Manager, double-click on the feature, select the Driver tab, and select uninstall.

7.1 Windows XP Embedded Module/Driver Dependencies

Table 6 lists the dependencies for the driver modules or patch within the Intel® EP80579 Software Drivers for Embedded Applications package on Windows XP Embedded. OS installation is assumed.

<table>
<thead>
<tr>
<th>Module/Driver</th>
<th>Dependency 1</th>
<th>Dependency 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel Chipset Device Software</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>Controller Area Network (CAN)</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>Intel Matrix Storage Manager</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>IEEE 1588 Hardware Assist (1588)</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>General Purpose IO (GPIO)</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>Gigabit Ethernet (GbE)</td>
<td>GCU</td>
<td>None</td>
</tr>
<tr>
<td>Global Configuration Unit (GCU)</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>SMBus (SMB)</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Watchdog Timer (WDT)</td>
<td>None</td>
<td>-</td>
</tr>
</tbody>
</table>
8.0 Troubleshooting

- It is advised to plug the Matrox Millennium G550 PCIe* graphics card into the PCI Express* slot closest to memory. Lack of video has been exhibited in the PCI Express x4 slot.

- When using the Matrox Millennium G550 PCIe graphics card, utilize the bottom video port on this dual port video card. The top port will not output any video to the display.

- Some LCD monitors will not function properly with the Intel® EP80579 Integrated Processor with Intel® QuickAssist Technology Development Board. In one case, an Acer monitor, model AL1916 C reported "Input Not Supported" upon boot complete. Try an alternative LCD monitor if no video support is displayed.
## 9.0 Glossary

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHCI</td>
<td>Advanced Host Controller Interface</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller Area Network</td>
</tr>
<tr>
<td>EDMA</td>
<td>Enhanced Direct Memory Access</td>
</tr>
<tr>
<td>GbE</td>
<td>Gigabit Ethernet</td>
</tr>
<tr>
<td>GCU</td>
<td>Global Configuration Unit</td>
</tr>
<tr>
<td>GPIO</td>
<td>General Purpose Input Output</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IHS</td>
<td>Integrated Heat Spreader</td>
</tr>
<tr>
<td>SMbus</td>
<td>System Management Bus</td>
</tr>
<tr>
<td>TIM</td>
<td>Thermal Interface Material</td>
</tr>
<tr>
<td>WDT</td>
<td>Watchdog Timer</td>
</tr>
</tbody>
</table>

§ §