

GENE-1425

Intel® Xscale IXP422/425 CPU

RISC CPU Module

Lanx4 / WANx2

USB2.0 Host x2 / USB1.1 Client x1

Support Mini-PCI socket x2

GENE-1425 Rev. A Manual 1st Ed.

May. 2005

Copyright Notice

This document is copyrighted, 2005. All rights are reserved. The original manufacturer reserves the right to make improvements to the products described in this manual at any time without notice.

No part of this manual may be reproduced, copied, translated, or transmitted in any form or by any means without the prior written permission of the original manufacturer. Information provided in this manual is intended to be accurate and reliable. However, the original manufacturer assumes no responsibility for its use, or for any infringements upon the rights of third parties that may result from its use.

The material in this document is for product information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, AAEON assumes no liabilities resulting from errors or omissions in this document, or from the use of the information contained herein.

AAEON reserves the right to make changes in the product design without notice to its users.

Acknowledgments

All other products' name or trademarks are properties of their respective owners.

- Award is a trademark of Award Software International, Inc.
- CompactFlash™ is a trademark of the Compact Flash Association.
- Intel®, Pentium® M, and Celeron® M are trademarks of Intel® Corporation.
- Microsoft Windows® is a registered trademark of Microsoft Corp.
- ITE is a trademark of Integrated Technology Express, Inc.
- IBM, PC/AT, PS/2, and VGA are trademarks of International Business Machines Corporation.
- SoundBlaster is a trademark of Creative Labs, Inc.

All other product names or trademarks are properties of their respective owners.

Packing List

Before you begin installing your card, please make sure that the following materials have been shipped:

- 1 GENE-1425 CPU Board
- 2 RS-232 Cable
- 1 USB Cable
- 1 Null Modem Cable
- 1 Crossover Ethernet Cable
- 1 +12V DC Power Adapter
- 1 Power Cord
- 1 Quick Installation Guide
- 1 CD-ROM for manual (in PDF format) and
Embedded Linux image files

If any of these items should be missing or damaged, please contact your distributor or sales representative immediately.

Contents

Chapter 1 General Information

1.1 Introduction	1-2
1.2 Features.....	1-3
1.3 Specifications.....	1-4

Chapter 2 Quick Installation Guide

2.1 Safety Precautions.....	2-2
2.2 Location of Connectors and Jumpers.....	2-3
2.3 Mechanical Drawing	2-4
2.4 List of Connectors.....	2-5
2.5 IXP42x JTAG Header (CN1).....	2-6
2.6 Digital I/O Connector (CN2).....	2-6
2.7 USB 2.0 Connector (CN3)	2-7
2.8 Expansion Bus Pin Header (CN4)	2-7
2.9 Serial Port COM 2 (CN5).....	2-9
2.10 Serial Port COM 1 (CN6).....	2-9
2.11 Power DC-Jack (CN7)	2-10
2.12 1st Ethernet WAN RJ-45 Phone Jack (CN8).....	2-10
2.13 2nd Ethernet WAN RJ-45 Phone Jack (CN9).....	2-11
2.14 LAN x 4 RJ-45 Phone Jack (CN10).....	2-11
2.15 Client USB 1.1 Connector (CN11).....	2-12

Chapter 3 Booting the System

3.1 Overview.....	3-2
3.2 Connection setup.....	3-2
3.3 Booting Process.....	3-4

Chapter 4 RedBoot

4.1 Overview.....	4-2
4.2 RedBoot Setting.....	4-3
4.3 RedBoot Development.....	4-5

Chapter 5 Embedded Linux System

5.1 Overview.....	5-2
5.2 File System.....	5-2
5.3 Setting up Network.....	5-4
5.4 Mounting external devices.....	5-5
5.5 Updating Images.....	5-7
5.6 References.....	5-10

Appendix A Programming The Watchdog Timer

A.1 Programming.....	A-2
A.2 Watchdog Timer Demo Program.....	A-2

Chapter

1

**General
Information**

1.1 Introduction

The GENE-1425 is a single board developed for the applications of Home Gateway、Firewall、Router... etc. The GENE-1425 features fanless and low power consumption by adopting the Intel® IXP422/425 CPU which has RISC architecture focused on networking applications,

The GENE-1425 has 2 WAN ports and 4 LAN ports, and two Mini-PCI sockets for Wireless LAN Card and expansion for customization. The GENE-1425 can be connected to Mobile Disk or Web camera devices through a USB2.0 port. In addition, RS-232 port provides the function of modem connection backup. The form factor of the GENE-1425 is the same as a SubCompact board, and the onboard OS is the Linux platform.

1.2 Features

- Intel® XScale IXP422/425 Network Processor
- MontaVista Linux Pre-installed in Onboard Flash Memory. No License Fee Required
- 2 WAN ports / 4 LAN ports
- 2 COM / 2 USB2.0 Host
- Supports 2 Type II Mini PCI Slots
- JTAG Port for Debug Tools
- +9V to +24V Input

1.3 Specifications

System

- CPU Onboard Intel® Xscale
IXP420/422/425 266/533MHz
Network Processor
- Memory Onboard 32/64MB SDRAM
- Ethernet Realtek RTL8305SC for one
WAN port and four LAN ports,
Intel® 82551QM for an additional
WAN port. 10/100Base-TX
RJ-45 connector x 6
- Boot Loader RedBoot
- Watchdog Timer Generate a Time-out System
Reset, setting via software
- SSD Intel® StraFlash 16MB, Type II
CompactFlash™ slot x 1
- Expansion Interface Type II Mini PCI Socket x 2,
Expansion bus
- Power Requirement +9V to +24V
- Board Size 6.14"(L) x 4"(W) (156mm x
101.6mm)
- Gross Weight 0.88lb (0.4kg)
- Operating Temperature 32°F~140°F (0°C~60°C)

I/O

- MIO RS-232 x 2 (COM1:full modem control signals, COM2: Tx, Rx, CTS, RTS only)
- USB One 5 x 2 Pin header supports 2 USB2.0 Host ports, One USB Type B connector supports USB1.1 Client port
- Digital I/O Supports 4 in and 4 out

Chapter

2

Quick Installation Guide



Notice:

The Quick Installation Guide is derived from Chapter 2 of user manual. For other chapters and further installation instructions, please refer to the user manual CD-ROM that came with the product.

2.1 Safety Precautions

Warning!

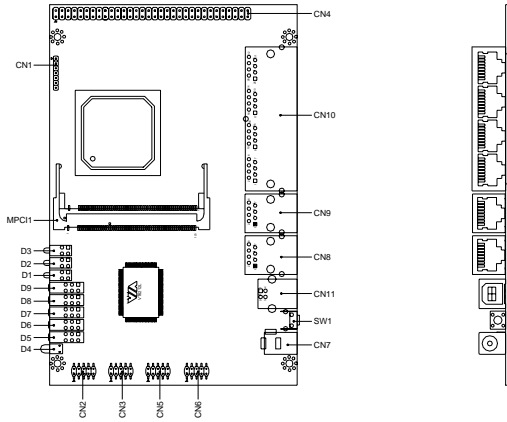
Always completely disconnect the power cord from your board whenever you are working on it. Do not make connections while the power is on, because a sudden rush of power can damage sensitive electronic components.

Caution!

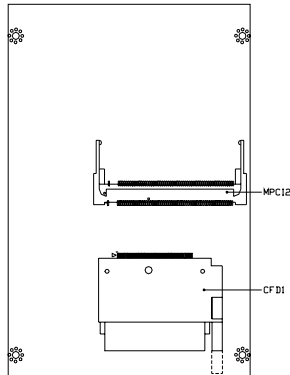
Always ground yourself to remove any static charge before touching the board. Modern electronic devices are very sensitive to static electric charges. Use a grounding wrist strap at all times. Place all electronic components on a static-dissipative surface or in a static-shielded bag when they are not in the chassis

2.2 Location of Connectors and Jumpers

Component Side



Solder Side



2.4 List of Connectors

The board has a number of connectors that allow you to configure your system to suit your application. The table below shows the function of each board's connectors:

Connectors

Connectors	
Label	Function
CN1	JTAG
CN2	Digital I/O (4 IN / 4 OUT)
CN3	Host USB 2.0 x 2
CN4	Expansion Bus Pin Header (For Test Only)
CN5	COM2
CN6	COM1
CN7	Power Adapter
CN8	WAN Port 2
CN9	WAN Port 1
CN10	LAN Port x 4
CN11	Client USB 1.1
MPC1	Expansion Mini-PCI Slot 1
MPC2	Expansion Mini-PCI Slot 2
CFD1	Compact Flash Disk Connector
SW1	Hardware RESET

2.5 IXP42x JTAG Header (CN1)

JTAG	
Pin	Signal
1	+3.3 Volt.
2	TCK
3	TDI
4	TDO
5	TMS
6	TRST#
7	Ground

2.6 Digital I/O Connector (CN2)

Digital I/O 4-in / 4-out or manual setting

Pin	Signal	Pin	Signal
1	DIO_IN0	2	DIO_IN1
3	DIO_IN2	4	DIO_IN3
5	DIO_OUT0	6	DIO_OUT1
7	DIO_OUT2	8	DIO_OUT3
9	+5 Volt.	10	Ground

2.7 USB 2.0 Connector (CN3)

USB Connector

Pin	Signal	Pin	Signal
1	+5 Volt.	2	Ground
3	D1-	4	Ground
5	D1+	6	D2+
7	Ground	8	D2-
9	Ground	10	+5 Volt.

2.8 Expansion Bus Pin Header (CN4)

Expansion Bus

Pin	Signal	Pin	Signal
1	+5 Volt.	2	+3.3 Volt.
3	+5 Volt.	4	+3.3 Volt.
5	EXB_Data 1	6	EXB_Data 0
7	EXB_Data 3	8	EXB_Data 2
9	Reserved	10	N/C
11	EXB_Data 5	12	EXB_Data 4
13	EXB_Data 7	14	EXB_Data 6
15	Ground	16	Ground
17	EXB_Data 9	18	EXB_Data 8
19	EXB_Data 11	20	EXB_Data 10

21	Chip Select 1	22	Chip Select 0
23	EXB_Data 13	24	EXB_Data 12
25	EXB_Data 15	26	EXB_Data 14
27	Ground	28	Ground
29	EXB_Address 1	30	EXB_Address 0
31	EXB_Address 3	32	EXB_Address 2
33	EXB_Address 5	34	EXB_Address 4
35	EXB_Address 7	36	EXB_Address 6
37	Ground	38	Ground
39	EXB_Address 9	40	EXB_Address 8
41	EXB_Address 11	42	EXB_Address 10
43	EXB_Address 13	44	EXB_Address 12
45	EXB_Address 15	46	EXB_Address 14
47	Ground	48	Ground
49	EXB_Address 17	50	EXB_Address 16
51	EXB_Address 19	52	EXB_Address 18
53	EXB_Address 21	54	EXB_Address 20
55	EXB_Address 23	56	EXB_Address 22
57	Ground	58	Ground
59	EXT_CLK	60	RD#
61	IOWAIT#	62	WR#
63	ALE	64	RDY

2.9 Serial Port COM 2 (CN5)

COM 2 / Console or Debug mode

Pin	Signal	Pin	Signal
1	Reserved	2	RX
3	TX	4	Reserved
5	Ground	6	Reserved
7	RTS	8	CTS
9	N/C	10	N/C

2.10 Serial Port COM 1 (CN6)

COM 1 / Full Function mode

Pin	Signal	Pin	Signal
1	DCD	2	RX
3	TX	4	DTR
5	Ground	6	DSR
7	RTS	8	CTS
9	RI	10	N/C

2.11 Power DC-Jack (CN7)

DC-Jack

Pin	Signal
1	Power IN
2	Ground
3	Ground

2.12 1st Ethernet WAN RJ-45 Phone Jack (CN8)

WAN 1

Pin	Signal	Pin	Signal
1	TX+	2	TX-
3	RX+	4	N/C
5	N/C	6	RX-
7	N/C	8	N/C
9	N/C	10	N/C

2.13 2nd Ethernet WAN RJ-45 Phone Jack (CN9)

WAN 2

Pin	Signal	Pin	Signal
1	RX+	2	RX-
3	TX+	4	N/C
5	N/C	6	TX-
7	N/C	8	N/C
9	N/C	10	N/C

2.14 LAN x 4 RJ-45 Phone Jack (CN10)

LAN x 4

Pin	Signal	Pin	Signal
1	RXD0+	2	RXD0-
3	TXD0+	4	N/C
5	N/C	6	TXD0-
7	N/C	8	N/C
9	RXD1+	10	RXD1-
11	TXD1+	12	N/C
13	N/C	14	TXD1-
15	N/C	16	N/C
17	RXD2+	18	RXD2-

19	TXD2+	20	N/C
21	N/C	22	TXD2-
23	N/C	24	N/C
25	RXD3+	26	RXD3-
27	TXD3+	28	N/C
29	N/C	30	TXD3-
31	N/C	32	N/C
33	Ground	34	Ground
35	Ground	36	Ground
37	Ground		

2.15 Client USB 1.1 Connector (CN11)

Client USB

Pin	Signal
1	+5 Volt.
2	D-
3	D+
4	Ground

Chapter

3

Booting the System

3.1 Overview

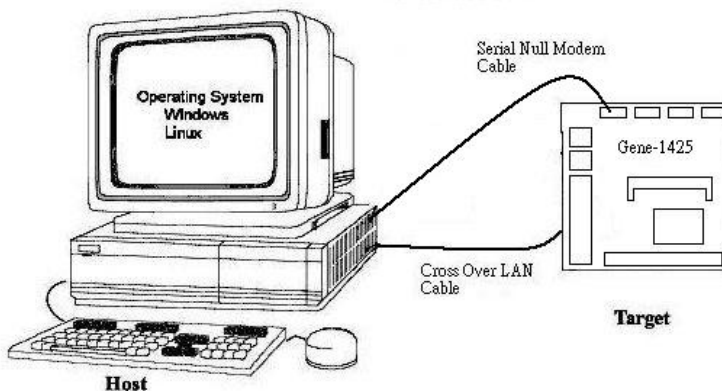
Gene-1425 includes the following software in onboard flash:

1. Bootloader: RedBoot (RedBoot for Intel XScale evaluation boards March 30, 2004)
2. Kernel: MontaVista Embedded Linux 3.1 (2.4.20 Kernel)

The software included with Gene-1425 is provided as it is for evaluation purposes.

3.2 Connection setup

Gene-1425 is a headless system design suitable for network computing applications. A host PC connected to Gene-1425 remotely by LAN or Serial connection is used for different application purposes:



Connection Setup:

There are 6 RJ-45 connectors on Gene-1425. CN8, CN9 is a single

connector act as WAN port, interfacing external network connections. CN10 is a switch connector with four RJ-45 connectors act as LAN port, interfacing internal network connections.

To demonstrate the usage of Gene-1425, AAEON provides two kinds of connection setup; Download mode connection and Router mode connection.

Download mode connection:

Network Interface Setup:

In this connection, Gene-1425 is used in download/debug mode. Connect a Cross-Over RJ-45 Ethernet cable from the host PC to additional WAN port (eth0 in figure 1, socket CN8) on Gene-1425. On the host PC, configure the network TCP/IP connection to static IP setting:

IP: 192.168.100.1

Mask: 255.255.255.0

In this configuration, host PC is act as download server servicing Gene-1425 download request through TFTP or NFS protocol.

Serial Port Interface Setup:

Connect a null modem cable from COM1 of the host PC to UART1 (CN6) on Gene-1425. On the host PC, start HyperTerminal program (if MS window), or Mincom (if Linux). Setup the connection to the following parameters:

- Connection: COM1
- Speed: 115200
- Data-Bit: 8
- Parity Check: None
- Stop-Bit: 1
- Stop-Bit: 1
- Flow Control: None

Router mode connection:

In this connection, Gene-1425 is working as a router between your Internet Service Provider and your host PC. Connect a normal RJ-45 Ethernet cable from the host PC to one of the four LAN ports on Gene-1425. Connect another normal RJ-45 Ethernet cable from WAN port (socket CN9) to your ISP provider LAN connection. On the host PC configure the network TCP/IP connection to one of the following choice:

1. Enable DHCP.
2. Or use static IP:
IP: 192.168.1.3
Mask: 255.255.255.0
Default Gateway: 192.168.1.254

3.3 Booting Process

Set up Gene-1425 with Download Mode connection. Start a HyperTerminal in the host system. Once the system connected to the power, Gene-1425 will automatically boot up. In the

HyperTerminal program window, after a '+' boot indication, RedBoot boot up screen will show up.

```

+No EEPROM found for i82559. Default MAC address loaded
Ethernet eth0: MAC address 00:07:32:04:25:03
IP: 192.168.100.3/255.255.255.0, Gateway: 192.168.100.1
Default server: 192.168.100.1, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Red Hat certified release, version 1.94 - built 16:13:44, Nov 29 2004

Platform: IXDP425 Development Platform (XScale) BE
Copyright (C) 2000, 2001, 2002, 2003, 2004 Red Hat, Inc.

RAM: 0x00000000-0x02000000, 0x0001f5b0-0x01fd1000 available
FLASH: 0x50000000 - 0x51000000, 128 blocks of 0x00020000 bytes each.
== Executing boot script in 2.000 seconds - enter ^C to abort

```

Press Control+C if one wishes to change the configuration of RedBoot. If no key is pressed during the first screen of RedBoot, the system will automatically enter Kernel initialization.

Kernel initialization includes the following process:

1. Uncompress kernel image.

The Embedded Linux Kernel is uncompressed in the following messages after “exec” command in RedBoot:

```
Using base address 0x00100000 and length 0x00100000
```

```
Uncompressing
```

```
Linux.....done, booting the
kernel.
```

```
Linux version 2.4.20_mvl31-ixdp4xx (root@victorlai) (gcc
version 3.3.1 (MontaVista 3.3.1-3.0.10.0300532 2003-12-24))
```

#14 Tue Dec 14 09:44:02 CST 2004

2. Initialization of kernel drivers.

During this stage, the Embedded Linux Kernel is initializing the kernel drivers including CPU, onboard Flash, USB, and LAN
...etc.

3. Mount ramdisk image for Linux Filesystem.

After Embedded Linux Kernel initialization, the Linux will try to find a ramdisk image on flash. System will uncompress the ramdisk image and mount the file as Filesystem for Embedded Linux.

4. Initial boot scripts under /etc/rc.d.

Customized initial scripts are being executed at this time. This includes Networking setup scripts and custom driver modules.

5. Command prompt

User application is able to execute after this stage.

6. Access through the Gene-1425 server web page:

On the host computer, start an Internet browser. Visit the following address: "192.168.1.254/index.html". A Gene-1425 Web Access page will appear. This page provides the simple implementation of Server System Access with remote information acquisition and Router setup control.

Chapter

4

RedBoot

4.1 Overview

RedBoot is a complete bootstrap environment for embedded systems. Based on the eCos Hardware Abstraction Layer, RedBoot inherits the eCos qualities of reliability, compactness, configurability, and portability.

RedBoot allows download and execution of embedded applications via serial or Ethernet, including embedded Linux and eCos applications. It can be used for both product development (debug support) and in deployed products in the field (flash update and network booting).

Ethernet download and debug support is included, allowing RedBoot to retrieve its IP parameters via BOOTP or DHCP, and program images to be downloaded using TFTP. Images can also be downloaded over serial, using X- or Y-modem.

RedBoot can be used to communicate with GDB (the GNU Debugger) to debug applications via serial or Ethernet, including the ability to interrupt a running application started by GDB.

An interactive command-line interface is provided to allow management of the Flash images, image download, RedBoot configuration, etc., accessible via serial or Ethernet. For unattended or automated startup, boot scripts can be stored in Flash allowing for example loading of images from Flash or a TFTP server.

4.2 RedBoot Setting

Flash Image System List:

Name	FLASH addr	Mem addr	Length	Entry point
RedBoot	0x50000000	0x50000000	0x00040000	0x00000000
RedBoot config	0x50FC0000	0x50FC0000	0x00001000	0x00000000
FIS directory	0x50FE0000	0x50FE0000	0x00020000	0x00000000
RedBoot-RAM	0x50040000	0x00020000	0x00040000	0x00020040
Kernel_NFS	0x50080000	0x00100000	0x00100000	0x00100000
Kernel_Ramdisk	0x50180000	0x00100000	0x00100000	0x00100000
Ramdisk	0x50280000	0x01000000	0x00300000	0x01000000
MAC_Addr	0x50580000	0x00700000	0x00020000	0x00700000
jffs2disk	0x505A0000	0x00800000	0x00100000	0x00800000

Redboot: RedBoot image

RedBoot config: Config file for RedBoot

FIS directory: FIS directory list

RedBoot-RAM: RAM version for RedBoot

Kernel_NFS: Linux Kernel supports boot from NFS filesystem

Kernel_Ramdisk: Linux Kernel supports boot to Ramdisk filesystem (Default)

Ramdisk: Ramdisk filesystem image

MAC_Addr: MAC addresses for IXP42X LAN controller (ixp0, ixp1) and onboard Intel LAN 82551.

Jffs2disk: Journaling Flash File System image. This flash image is mounted in /mnt/mtd after boot to Kernel.

Virtual memory map:

X	C	B	Description
-	-	-	-----
0	0	0	Uncached/Unbuffered
0	0	1	Uncached/Buffered
0	1	0	Cached/Buffered Write Through, Read Allocate
0	1	1	Cached/Buffered Write Back, Read Allocate
1	0	0	Invalid -- not used
1	0	1	Uncached/Buffered No write buffer coalescing
1	1	0	Mini DCache - Policy set by Aux Ctl Register
1	1	1	Cached/Buffered Write Back, Read/Write Allocate

Virtual Address	Physical Address	XCB	Size (MB)	Description
-----	-----	-----	-----	-----
0x00000000	0x00000000	010	32	SDRAM (cached)
0x10000000	0x00000000	010	32	SDRAM (alias)
0x20000000	0x00000000	000	32	SDRAM (uncached)
0x48000000	0x48000000	000	64	PCI Data
0x50000000	0x50000000	010	16	Flash (CS0)
0x51000000	0x51000000	000	112	CS1 - CS7
0x60000000	0x60000000	000	64	Queue Manager
0xC0000000	0xC0000000	000	1	PCI Controller
0xC4000000	0xC4000000	000	1	Exp. Bus Config
0xC8000000	0xC8000000	000	1	Misc IXP425 IO
0xCC000000	0xCC000000	000	1	SDRAM Config

RedBoot Default Boot Script

The default boot script will boot the system to Ramdisk Image. To change it, use “fconfig” command.

```
>> fis load Ramdisk
>> fis load Kernel_Ramdisk
>> exec 0x00100000
```

RedBoot Network Setup (To change it, use “fconfig” command)

Use BOOTP for network configuration: false

Gateway IP address: 192.168.100.1

Local IP address: 192.168.100.3

Local IP address mask: 255.255.255.0

Default server IP address: 192.168.100.1 (this is needed if one wish to download file from a TFTP server.)

Console baud rate: 115200

4.3 RedBoot Development

The source code for RedBoot is included in the Development CD of Gene-1425. The RedBoot development process is clearly described in the Intel and RedHat released documents in the following link:

1. Intel[®] IXP400 Software: RedBoot* v1.94 Software Release Notes

http://www.intel.com/design/network/products/npfamily/ixp400_current.htm (Registration with Intel is required for download. Look for Intel[®] IXP400 Software 1.5 -> Linux Boot Loader.)

2. Intel[®] IXP4XX Product Line of Network Processors and IXC1100 Control Plane Processor: Customizing RedBoot*

<http://www.intel.com/design/network/applnotts/254308.htm>

3. RedBoot™ User's Guide

<http://sources.redhat.com/ecos/docs-latest/redboot/redboot-guide.html>

Chapter

5

Embedded Linux System

5.1 Overview

The Gene-1425 uses MontaVista Embedded Linux system to provide robust and industrial trusted Operating System. This industry-leading comprehensive embedded operating system and cross development environment provides a common source and binary platform across a broad range of processor architectures. MontaVista Software provides Open Source platform allowing system designers to innovate across a wide range of inter-connected intelligent devices and communications infrastructure. It also enables software developers with a commercial-grade Linux-based operating system. The Professional Edition includes a modern OS featuring real-time functionality, multi-process and multi-threaded with extensive bundled software components including rich networking.

5.2 File System

Gene-1425 runs using text console with an 8M ramdisk using MontaVista embedded Linux system. The default Gene-1425 system image includes the following:

1. Web server (thttpd), FTP server (wu-ftpd), DHCP server (dhcpd on LAN interface).
2. Router configuration between WAN and LAN Ethernet ports.
3. Software executables for common Linux text command under “/bin, /sbin, /usr/bin, /usr/sbin”.

4. Custom evaluation software module build by AAEON is put under “/usr/local/bin”:

/usr/local/bin/dio: This program uses command interface used to control onboard Digital IO interface.

- Usage: dio option data
 - option:
 - -r (Read DIO port: 0->low, 1->high)
 - -w (Write to DIO port: 0->low, 1->high)
 - -d (Set DIO port Input/Output Direction: 0->output, 1->input)
 - -m (Set DIO port Output Style: 0->level, 1->pulse)
 - -s (Read DIO port Status)
 - data:
 - Byte Hex data. (ex: 0x0f, 0x30)

/usr/local/bin/watchdog: This program provides a menu interface for the control of onboard watchdog function.

5. **Journal Flash File System (jffs2) mounted on /mnt/mtd:** Gene-1425 reserves a partition on the onboard flash device. The Linux system automatically mounts the partition to /mnt/mtd with jffs2 (journal flash file system version 2) during boot up process. User can store data to this partition for permanent data storage

up to 10M on this partition. System network setting is stored in this partition as network.conf. **Do not remove /mnt/mtd/network.conf if user wishes to have router application work properly.**

5.3 Setting up Network

Gene-1425 has three network controller; named eth0 (cn8), ixp0 (cn9), and ixp1 (cn10) under Gene-1425 Embedded Linux system. By default, the network configuration on Gene-1425 is initialized as follows:

Eth0: In Gene-1425 Linux system image, eth0 is used to transmit images and applications between host and Gene-1425. Host and Gene-1425 is communicated on network of IP 192.168.100.X. This is by default used to link to TFTP server with IP 192.168.100.1.

IXP0: This Ethernet port is act as WAN port for router setup. IXP0 is setup as a dhcp client. If no dhcp server is presented to service dhcp request on this port, a default static IP will be assigned. To change to static IP initialization, one needs to modify /mnt/mtd/network.conf with “WAN_STATICIP = y”, “WAN_IPADDR = network_static_ip”, “WAN_NETMASK=network_mask”, and “WAN_ROUTE = network_gateway”.

IXP1: This port is act as LAN ports for router setup. A dhcp server (thttpd) is running on this port to service any external network

device connected to it. The IP for this interface is defaulted to 192.168.1.254. If a device connected to this port has no dhcp client ability, one can manually setup the network interface on the device to 192.168.1.X, where X =2~253. Initial setting for this port can also be modified in /mnt/mtd/network.conf.

To view the detail information on each network controller, issue "ifconfig" command in MontaVista Embedded Linux.

5.4 Mounting external devices

To evaluated custom applications, User can store their application file in the following media and using the corresponding way to attach their storage device on to Gene-1425 file system.

Mount an USB storage device:

When an USB storage device is plugged into the USB connector, a device file (sda) with major number 8 is needed to control the device. To use the USB storage device, use the following commands to enable the usage (assume the device has one partition with vfat filesystem): "mknod /dev/sda1 b 8 1", "mount -t vfat /dev/sda1 /mnt/usb_cf".

Script /usr/local/bin/mountusbdevice.sh is a default script for mounting USB devices. For other USB devices, a specific driver may need to build separately and load before using it.

Mount network file system (NFS):

Another way to control Gene-1425 is to use Network File System.

Gene-1425 can mount a shared folder on the remote NFS server (ex: host machine). To mount a network file system folder "/root" on the host system to local /mnt/nfs, use the following command in the HyperTerminal window: "mount -t ext2 192.168.100.1:/root /mnt/nfs" (assuming host ip:192.168.100.1). Make sure the shared folder "/root" entry is listed on the /etc/exports file on host system.

For NFS references:

<http://ibiblio.org/mdw/HOWTO/NFS-HOWTO/index.html>

Download an application through TFTP server:

1. Install a TFTP server on the host (ex: SolarWinds TFTP Server for MS windows or any other TFTP server available on the Operating System of the host PC). SolarWinds TFTP Server can be downloaded from: <http://solarwinds.net>.

2. Put the images or files to download in the TFTP server root directory. Make sure the property of TFTP server allows Transmit.

Example: Run SolarWinds TFTP Server. Select File -> Configure to set configuration. Set "TFTP Root Directory". (Default directory is "TFTP-Root"). Put the image files into the "TFTP Root Directory". In the "Security" setting, select "Transmit and Receive files".

3. Setup the connection to "Download mode Connection" according to section 3.1. Prepare for download.

4. In the HyperTerminal Window, issue the following command for Gene-1425:
 - A. "tftp HostIP" (ex: "tftp 192.168.100.1")
 - B. "get --downloadfile--" (where --downloadfile-- is the file to download to put under TFTP server Root Directory.)

5.5 Updating Images

RedBoot, Kernel and Ramdisk images can be downloaded from your host PC to Gene-1425 through TFTP. Try the following procedure for image updates.

1. On the host PC, install SolarWinds TFTP Server (or any other TFTP server available). It can be downloaded for free from <http://solarwinds.net>.
2. Put the update images in the TFTP server root directory. Make sure the property of TFTP server allows Transmit. *Example: Run SolarWinds TFTP Server. Select File -> Configure to set configuration. Set "TFTP Root Directory". (Default directory is "TFTP-Root"). Put the image files into the "TFTP Root Directory". In the "Security" setting, select "Transmit and Receive files".*
3. Setup the connection to "Download mode Connection" . according to section 3.1. Prepare for download.

4. Press reset button of GENE-1425. In the HyperTerminal window, instantly press “Ctrl-C” in the host PC to interrupt boot procedure and enter RedBoot command prompt.
5. Follow the instruction for Kernel, Ramdisk or RedBoot download accordingly:

Download Kernel Image

1. First enter command “**load -r -v -b 0x00100000 --zimagefile--**”. (Where --zimagefile-- is the updated Kernel image file.)
2. After the file is loaded, type “**fis create Kernel_Ramdisk -b 0x00100000**”. (0x00100000 is default address AAeon defined for kernel image.)

Download Ramdisk Image

1. First enter command “**load -r -v -b 0x01000000 --ramdiskfile--**”. (Where --ramdiskfile-- is the updated ramdisk image file.)
2. After the file is loaded, type “**fis create Ramdisk -b 0x01000000**”. (0x01000000 is default address AAeon defined for ramdisk file system.)

Download RedBoot Image

Note: Wrongful update of RedBoot Image will cause system fail to boot. Sending the system back for RMA is required if such problem occurs. Only update this if you know what you are doing. Also reference section 5.15 on the Intel document for detail explanations:

Intel® IXP4XX Product Line of Network Processors and IXC1100
Control Plane Processor: Customizing RedBoot*

<http://www.intel.com/design/network/applnots/254308.htm>

1. First enter command "**load -v --redbootRAMImageFile--**". (Where --redbootRAMImageFile-- is the RAM version of the updated RedBoot image file.)
2. After the file is loaded, type "**go**". This will make the system reboot to the RAM version of RedBoot image downloaded in the procedure 1.
3. Next enter command "**load -v -b 0x01600000 --redbootROMImageFile--**". (Where --redbootRAMImageFile-- is the ROM version of the updated RedBoot image file. This file is the actual file to write to flash.)
4. Next enter command "**fis create RedBoot -b 0x01600000**". (0x01600000 is default address AA EON defined for RedBoot image.)
5. Press reset button to reset the system.

5.6 References

RedBoot:

<http://sources.redhat.com/redboot/>

<http://sources.redhat.com/ecos/docs-latest/redboot/redboot-guide.html>

Intel:

<http://www.intel.com/design/network/products/npfamily/ixp4xx.htm>

<http://www.intel.com/design/network/products/npfamily/ixp425swr1.htm>

http://www.intel.com/design/network/products/npfamily/download_ixp400.htm

<http://www.intel.com/design/network/products/npfamily/docs/ixp4xx.htm#applnot>

MontaVista Linux:

<http://www.mvista.com/>

Appendix

A

**Programming the
Watchdog Timer**

A.1 Programming

GENE-1425 utilizes IXP-42X CPU internal watchdog timer.

The watchdog linux driver is provided as binary format under /device/WDTimerModule.

AAEON initial watchdog timer program is provided based on which you can develop customized program to fit your application. The following demo program is used to set the watchdog timer to time out after 60 seconds with a system reset.

A.2 Watchdog Timer Demo Program

```
#include <stdio.h>
#include <fcntl.h>

//Start timer with timer count
#define    WDTimer_startTimer    1
//Stop timer
#define    WDTimer_stopTimer    2
//Set timer mode :
#define    WDTimer_setTimerMode    3
//Reload timer count
#define    WDTimer_reloadTimerCount    4
//Read register ost_wdog (Timer Count)
```

```
#define    WDTimer_readCounter    5
//Read register ost_wdog_enab (Enable)
#define    WDTimer_readEnab      6
//Read register ost_sts (Status)
#define    WDTimer_readOSTStatus  7
//Read register ost_wdog_key
#define WDTimer_readKey          8

int main(void)
{
    unsigned long timercount,timersec;;
    int wddriver;

    wddriver = open("/dev/WDTimerModule",O_RDWR);
    if(!wddriver )
    {
        printf("Open failed.\n");
        exit(0);
    }
    timersec=60;
    timercount=timersec*66666666;
    ioctl(wddriver, WDTimer_stopTimer, NULL);
    ioctl(wddriver,WDTimer_reloadTimerCount,    timercount);
        //prevent count = 0 initially
    ioctl(wddriver, WDTimer_setTimerMode, 1);
```

```
//By Interrupt: type=0
//By Reset: type=1
//Do Nothing: type=2;
ioctl(wddriver,WDTimer_startTimer, timercount);

close(wddriver);
}
```