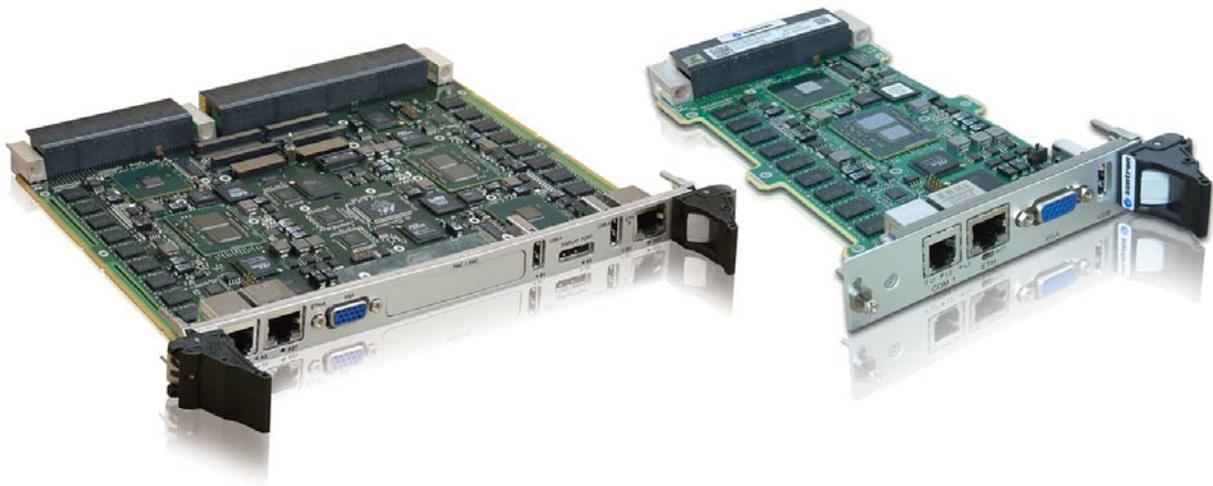


» VX6060 & VX3030 «



Release Note Fedora 12 on VX6060 & VX3030 Version 2.2 - ID 11257

SD.DT.F72-3e - November 2011

Revision History

Publication Title:		Release Notes Fedora 12 on VX6060 & VX3030
Doc. ID:		SD.DT.F72-3e
Rev.	Brief Description of Changes	Date of Issue
3e	Support of 64 bit (x86_64) Fedora 12 distribution	11-2011
2e	Version 2.0 - ID 11097 Application to VX3030 added	04-2011
1e	Version 1.3 - ID 11068	03-2011
0e	Initial Version	12-2010

Copyright © 2011 Kontron AG. All rights reserved. All data is for information purposes only and not guaranteed for legal purposes. Information has been carefully checked and is believed to be accurate; however, no responsibility is assumed for inaccuracies. Kontron and the Kontron logo and all other trademarks or registered trademarks are the property of their respective owners and are recognized. Specifications are subject to change without notice.

Proprietary Note

This document contains information proprietary to Kontron. It may not be copied or transmitted by any means, disclosed to others, or stored in any retrieval system or media without the prior written consent of Kontron or one of its authorized agents.

The information contained in this document is, to the best of our knowledge, entirely correct. However, Kontron cannot accept liability for any inaccuracies or the consequences thereof, or for any liability arising from the use or application of any circuit, product, or example shown in this document.

Kontron reserves the right to change, modify, or improve this document or the product described herein, as seen fit by Kontron without further notice.

Trademarks

This document may include names, company logos and trademarks, which are registered trademarks and, therefore, proprietary to their respective owners.

Environmental Protection Statement

This product has been manufactured to satisfy environmental protection requirements where possible. Many of the components used (structural parts, printed circuit boards, connectors, batteries, etc.) are capable of being recycled.

Final disposition of this product after its service life must be accomplished in accordance with applicable country, state, or local laws or regulations.



Environmental protection is a high priority with Kontron.

Kontron follows the DEEE/WEEE directive.

You are encouraged to return our products for proper disposal.

The Waste Electrical and Electronic Equipment (WEEE) Directive aims to:

- reduce waste arising from electrical and electronic equipment (EEE)
- make producers of EEE responsible for the environmental impact of their products, especially when they become waste
- encourage separate collection and subsequent treatment, reuse, recovery, recycling and sound environmental disposal of EEE
- improve the environmental performance of all those involved during the lifecycle of EEE

Conventions

This guide uses several types of notice: Note, Caution, ESD.



Note: this notice calls attention to important features or instructions.



Caution: this notice alert you to system damage, loss of data, or risk of personal injury.



ESD: This banner indicates an Electrostatic Sensitive Device.

All numbers are expressed in decimal, except addresses and memory or register data, which are expressed in hexadecimal. The prefix `0x` shows a hexadecimal number, following the `C` programming language convention.

The multipliers `k`, `M` and `G` have their conventional scientific and engineering meanings of $*10^3$, $*10^6$ and $*10^9$ respectively. The only exception to this is in the description of the size of memory areas, when `K`, `M` and `G` mean $*2^{10}$, $*2^{20}$ and $*2^{30}$ respectively.



When describing transfer rates, `k`, `M` and `G` mean $*10^3$, $*10^6$ and $*10^9$ *not* $*2^{10}$, $*2^{20}$ and $*2^{30}$.

In PowerPC terminology, multiple bit fields are numbered from 0 to n, where 0 is the MSB and n is the LSB. PCI and CompactPCI terminology follows the more familiar convention that bit 0 is the LSB and n is the MSB.

Signal names ending with an asterisk (*) or a hash (#) denote active low signals; all other signals are active high.

Signal names follow the PICMG 2.0 R3.0 CompactPCI Specification and the PCI Local Bus 2.3 Specification.

For Your Safety

Your new Kontron product was developed and tested carefully to provide all features necessary to ensure its compliance with electrical safety requirements. It was also designed for a long fault-free life. However, the life expectancy of your product can be drastically reduced by improper treatment during unpacking and installation. Therefore, in the interest of your own safety and of the correct operation of your new Kontron product, you are requested to conform with the following guidelines.

High Voltage Safety Instructions



Warning!

All operations on this device must be carried out by sufficiently skilled personnel only.



Caution, Electric Shock!

Before installing a not hot-swappable Kontron product into a system always ensure that your mains power is switched off. This applies also to the installation of piggybacks. Serious electrical shock hazards can exist during all installation, repair and maintenance operations with this product. Therefore, always unplug the power cable and any other cables which provide external voltages before performing work.

Special Handling and Unpacking Instructions



ESD Sensitive Device!

Electronic boards and their components are sensitive to static electricity. Therefore, care must be taken during all handling operations and inspections of this product, in order to ensure product integrity at all times

Do not handle this product out of its protective enclosure while it is not used for operational purposes unless it is otherwise protected.

Whenever possible, unpack or pack this product only at EOS/ESD safe work stations. Where a safe work station is not guaranteed, it is important for the user to be electrically discharged before touching the product with his/her hands or tools. This is most easily done by touching a metal part of your system housing.

It is particularly important to observe standard anti-static precautions when changing piggybacks, ROM devices, jumper settings etc. If the product contains batteries for RTC or memory backup, ensure that the board is not placed on conductive surfaces, including anti-static plastics or sponges. They can cause short circuits and damage the batteries or conductive circuits on the board.

General Instructions on Usage

In order to maintain Kontron's product warranty, this product must not be altered or modified in any way. Changes or modifications to the device, which are not explicitly approved by Kontron and described in this manual or received from Kontron's Technical Support as a special handling instruction, will void your warranty.

This device should only be installed in or connected to systems that fulfill all necessary technical and specific environmental requirements. This applies also to the operational temperature range of the specific board version, which must not be exceeded. If batteries are present, their temperature restrictions must be taken into account.

In performing all necessary installation and application operations, please follow only the instructions supplied by the present manual.

Keep all the original packaging material for future storage or warranty shipments. If it is necessary to store or ship the board, please re-pack it as nearly as possible in the manner in which it was delivered.

Special care is necessary when handling or unpacking the product. Please consult the special handling and unpacking instruction on the previous page of this manual.

Figure 1: VX6060 Overview (Non contractual photography)

2

Table Of Contents

Chapter 1 - Overview	1
Chapter 2 - Release Content	3
Chapter 3 - Associated Documentation	4
Chapter 4 - Required Configuration	5
4.1 Hardware Requirements	5
4.1.1 Hardware Requirements for VX6060 Board	5
4.1.2 Hardware Requirements for VX3030 Board	5
4.2 Firmware Requirements	6
4.2.1 VX6060 Board	6
4.2.2 VX3030 Board	6
4.3 Software Requirements	6
4.4 DVD-ROM Installation Example	7
4.4.1 DVD-ROM Installation Example for VX6060	7
4.4.2 DVD-ROM Installation Example for VX3030	8
Chapter 5 - Installation	9
5.1 Disk Installation	9
5.1.1 Fedora 12 Installation on CPUA of VX6060 and on VX3030	9
5.1.2 BSP Installation on CPUA of VX6060 and on VX3030	10
5.1.3 VX6060 - Fedora 12 Installation on CPUB	10
5.1.4 VX6060 - BSP Installation on CPUB	11
5.2 VX6060 - CPUB Diskless Installation	11
Chapter 6 - Fedora System Configuration	14
6.1 Network Manager	14
6.2 Network Interfaces	14
6.3 cpuspeed	17
Chapter 7 - BSP Specific Features	18
7.1 Sensors	18
7.2 Watchdog	20
7.3 VPD Tool	20
7.4 LEDs	22
7.5 Allocator	23
7.6 VXFabric	23
7.6.1 The vxfabric Command	24

7.6.2	Low Level access	26
7.6.3	IP over VXFabric	26
7.7	Multinodes Diskless	30
7.8	IGB Module	31
7.9	Sysvartool	32
7.10	GPIOs	33
7.11	cpldtool	34
7.12	I2C Busses	36
7.13	BIOS Update	36
7.14	FMRAM Example	37
7.15	VX6060: Setup Serial Console of CPUB on CPUA	38
Chapter 8 -	Additional Information	39
8.1	Known Limitations	39
8.2	Fixed Bug in the Current Release	39

Chapter 1 - Overview



Functional changes that differ from previous version of the document are identified by a vertical bar in the margin.

Linux, the open source Operating System is now taking a significant share of the OS market in Defense and Aerospace, after having taken ground initially in the enterprise server sector.



The goal of this document is to help you through the installation process of the Fedora 12 BSP distribution on the Kontron 6U VPX VX6060 board or 3U VPX VX3030 board.

In this document, the term VX6060 or VX3030 is used both for the VX6060 or VX3030 boards in standard or rugged conduction-cooled version:

»» VX6060 Single-slot 6U VPX board

- > VX6060-SA Standard Commercial version
- > VX6060-RC Rugged Conduction-Cooled version

»» VX3030 Single-slot 3U VPX board

- > VX3030-SA Standard Commercial version
- > VX3030-RC Rugged Conduction-Cooled version

In this document, the term VX6060-RTM or VX3030-RTM is associated to the VX6060 or VX3030 Rear Transition Module (RTM):

»» VX6060-RTM Rear Transition Module for the single-slot 6U VPX board

- > PB-VX3-002

»» VX3030-RTM Rear Transition Module for the single-slot 3U VPX board

- > PB-VX3-002

» Specific VX6060 Terminology

The VX6060 board is implemented as two similar CPU subsystems separated by a central PMC/XMC slot. This implies a common description of CPU subsystems on the board (ex: both parts have same CPU / SATA / USB / DDR3 /... interfaces).

For convenience, the CPU subsystem that faces P0/P1/P2 is called subsystem A and the second CPU subsystem on the other side is called subsystem B. In this documentation:

- CPU in subsystem A is named CPUA,
- CPU in subsystem B is named CPUB,
- USB in subsystem A is named USBA,
- USB in subsystem B is named USBB ,
- and so on.

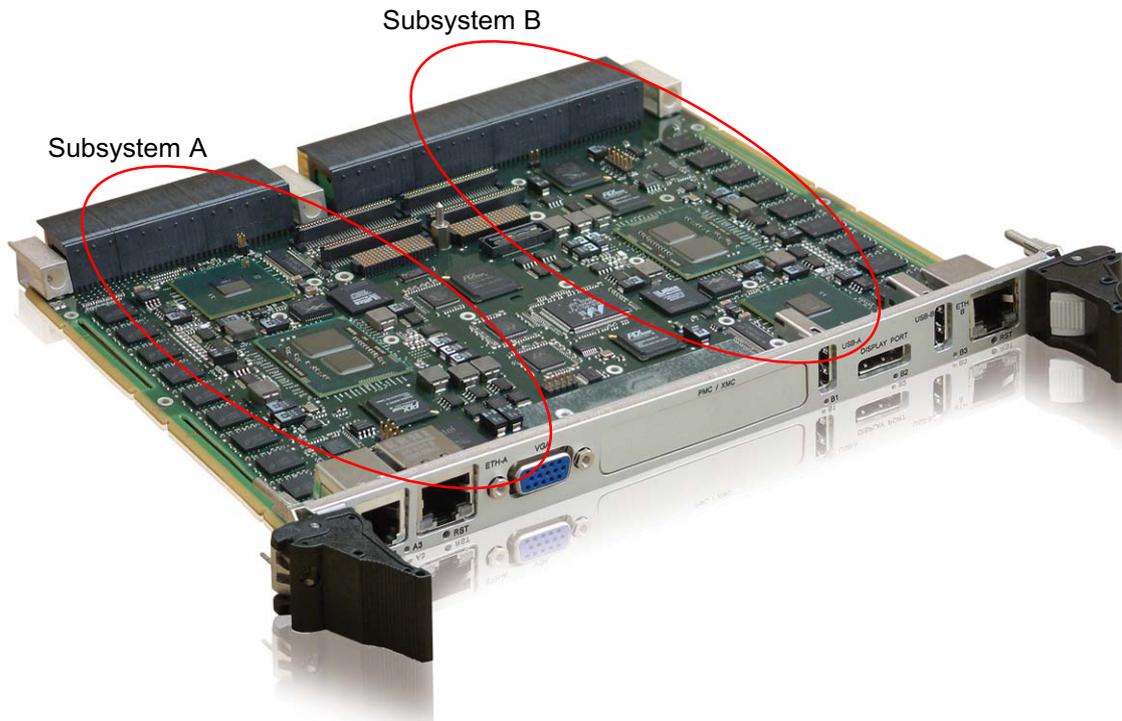


Figure 1: VX6060 Overview (Non contractual photography)

The VX6060 computing node is a VPX computing blade for parallel data and signal processing application.

With two independently implemented Intel® Core™ i7 processing nodes linked to a powerful Ethernet and PCIe infrastructure, the VX6060 is the ideal building block for intensive parallel computing workloads where a cluster of VX6060s is used in distributed or centralized OpenVPX environments. Target applications include radar, sonar, imaging systems, airborne fighters, and unmanned aerial vehicle (UAV) radar, as well as rugged multi-display consoles.

Chapter 2 - Release Content

The release is made of:

- one 32-bit DVD-ROM, reference: **Fedora 12 for i386**
or
- one 64 bit DVD-ROM: reference: **Fedora 12 for x86_64**

This DVD-ROM contains a standard Fedora 12 distribution with the addition of the installation menu in text mode (serial line configuration)

- one 32-bit DVD-ROM, on VX6060 & VX3030 Version 2.2 ID 11257 for i386
or
- one 64-bit DVD-ROM, on VX6060 & VX3030 Version 2.2 ID 11257 for x86_64

This DVD-ROM contains only the BSP specific packages as well as the related source packages.

It is required to install the related BSP DVD on the same Fedora flavor: 32-bit BSP on Fedora 12 for i686 and 64-bit BSP on Fedora 12 for x86_64.

The Board Support Package (BSP) provides support for some specific features of the board:

- Kernel: Update of the kernel of the Fedora 12 to support VX6060 & VX3030 specifics features and to fix issues.
Note that this kernel is required by the following features.
- Sensors: CPU Cores and Board temperatures and voltages.
- Vital Product Data (VPD) Tool: Get board's serial number, order code, E.C. Level, ...
- LEDs: Four Front Panel Tri-color LEDs
- GPIO: Driver to support the 5 GPIOs of the VX6060 & VX3030 (CPUA)
- Watchdog: Drivers to setup the Watchdogs of the board.
- BIOS Update tool: A command and script to update the BIOS of the board.
- Allocator: Driver and library to reserved contiguous memory area.
- VXFabric: Tool and driver to communicate over the PCIe (VPX Backplane and between CPUA and CPUB)
- CPLD register Tool (cpldtool): Tool to deal with hardware registers of the onboard CPLD
- FRAM support: Driver and special API file to read/write from/to the FRAM
- Diskless: Tool to configure, boot and manage diskless boards.
- SMI: swsmi tool is used to access the on-board ethernet switch.
- igb: igb module is an update of kernel module to support the 82580 ethernet chipset.
- PBIT report: sysvartool gives the report of the PBIT.
- I2C buses drivers: i2c bus drivers for the local i2c bus and the two backplane i2c busses.

More information on VX6060 & VX3030 BSP in Chapter 7 "BSP Specific Features" page 18.

Chapter 3 - Associated Documentation

» Kontron Documentation

> Hardware

- ▶ VX6060 6U VPX SBC User's Guide CA.DT.A76
- ▶ VX6060 Hardware Release Notes CA.DT.A77

- ▶ VX3030 3U VPX SBC User's Guide CA.DT.A87
- ▶ VX3030 Hardware Release Notes CA.DT.A88

> Firmware

- ▶ VX6060 BIOS User Manual SD.DT.F69

- ▶ VX3030 BIOS User Manual SD.DT.F81

» Fedora 12 Documentation

- > Documentation available at <http://fedoraproject.org/>

Chapter 4 - Required Configuration

4.1 Hardware Requirements

4.1.1 Hardware Requirements for VX6060 Board

- A Kontron VX6060 board.
- The Fedora 12 release may be installed on one of the following bootable disks:
 - ▶ a SATA disk connected to the SATA connectors available on VX6060-RTM board for the CPUA.
 - ▶ another SATA disk connected to the SATA connectors available on VX6060-RTM board for the CPUB.



The SATA disk connected on the CPUB (through the RTM) may be optional if the diskless boot configuration is used (refer to chapter 5.2 page 11)

- ▶ Optional onboard USB Flash Disk.
- A USB DVD-ROM device.
- Graphical display, USB keyboard and mouse.

The VX6060 board is implemented as two similar CPU subsystems separated by a central PMC/XMC slot. This implies a common description of CPU subsystems on the board (ex: both parts have same CPU / SATA / USB / DDR3 /... interfaces).

For convenience, the CPU subsystem that faces P0/P1/P2 is called subsystem A and the second CPU subsystem on the other side is called subsystem B.

In this documentation:

- CPU in subsystem A is named CPUA,
- CPU in subsystem B is named CPUB,
- USB in subsystem A is named USBA,
- USB in subsystem B is named USBB,
- and so on.

4.1.2 Hardware Requirements for VX3030 Board

- A Kontron VX3030 board.
- The Fedora 12 release may be installed on one of the following bootable disks:
 - ▶ a SATA disk connected to the SATA connectors available on VX3030-RTM board.
 - ▶ Optional onboard USB Flash Disk.
- A USB DVD-ROM device.
- Graphical display, USB keyboard and mouse.

4.2 Firmware Requirements

4.2.1 VX6060 Board

The version of the BIOS firmware must be at least:

➤ 11159

This version is displayed in the BIOS Setup.

4.2.2 VX3030 Board

The version of the BIOS firmware must be at least:

➤ 11186

This version is displayed in the BIOS Setup.

4.3 Software Requirements

➤ The DVD-ROM of the standard Fedora 12:

Fedora 12 for i386 (Graphic and Text installation)

or

Fedora 12 for x86_64 (Graphic and Text installation)

This one is the standard Fedora 12 but the installation menus are slightly modified for serial line console installation.

➤ The DVD-ROM associated to the BSP:

Fedora 12 BSP for VX6060 & VX3030 2.2 - ID 11257

or

Fedora 12 BSP for VX6060 & VX3030 2.2 - ID 11297



The 64-bit distribution should be privileged on VX6060 & VX3030 due to the CPU utilization and performances in 64-bit mode which are better than 32-bit in most of the benchmark cases. The 32-bit distribution still may be required if customers' or third parties' drivers are not available in 64-bit mode kernel. Note that most of the 32-bit applications may be used on a 64-bit distributions if 32-bit required libraries are installed and available. The last point to take into account is the fact that the memory and storage footprint may be a little bit more important in 64-bit than in 32-bit.



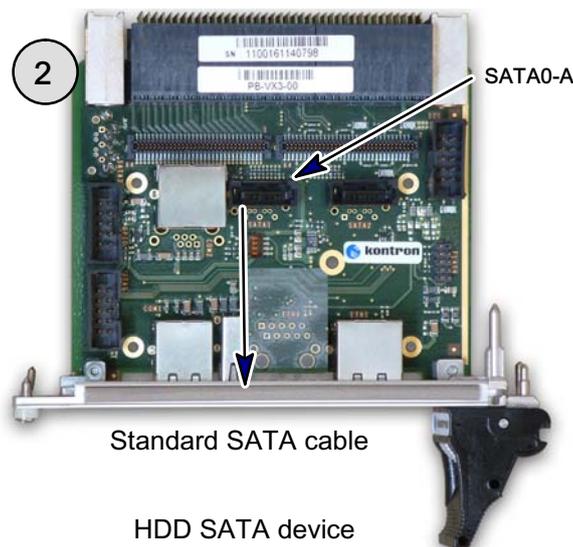
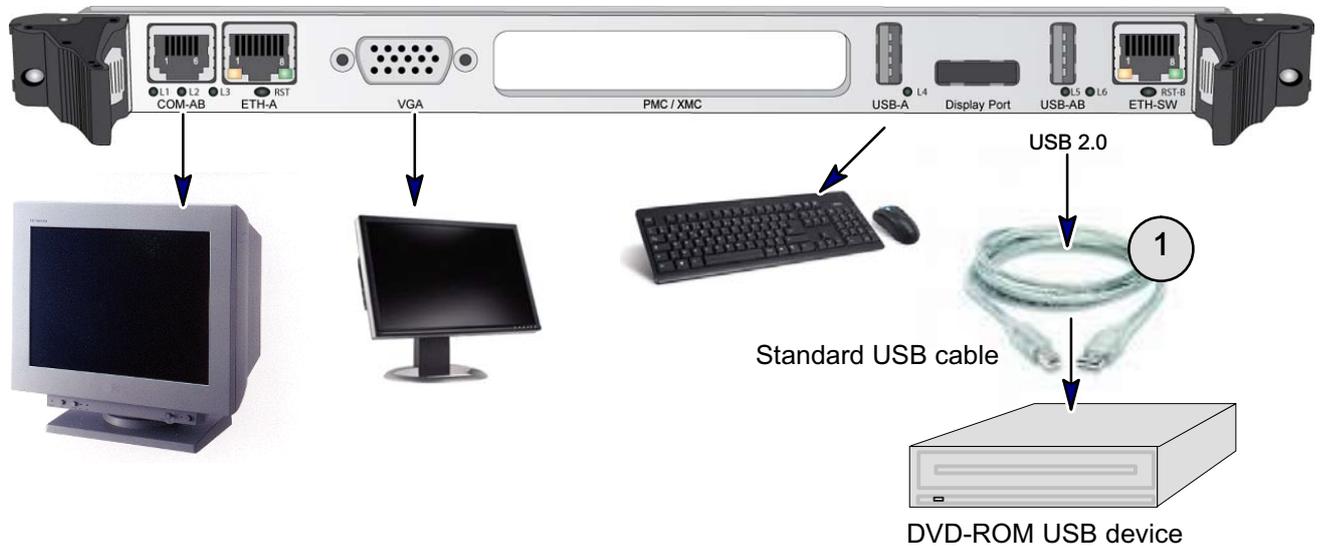
After the release is installed, the release version is saved in `/etc/board-release`:

```
# cat /etc/vx3030-release
VX3030 Board Support Package 2.0 [11097]
(Example for the VX3030 board)
```

4.4 DVD-ROM Installation Example

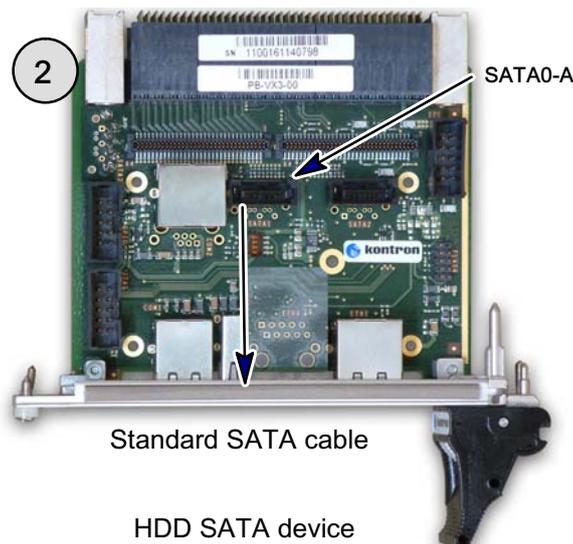
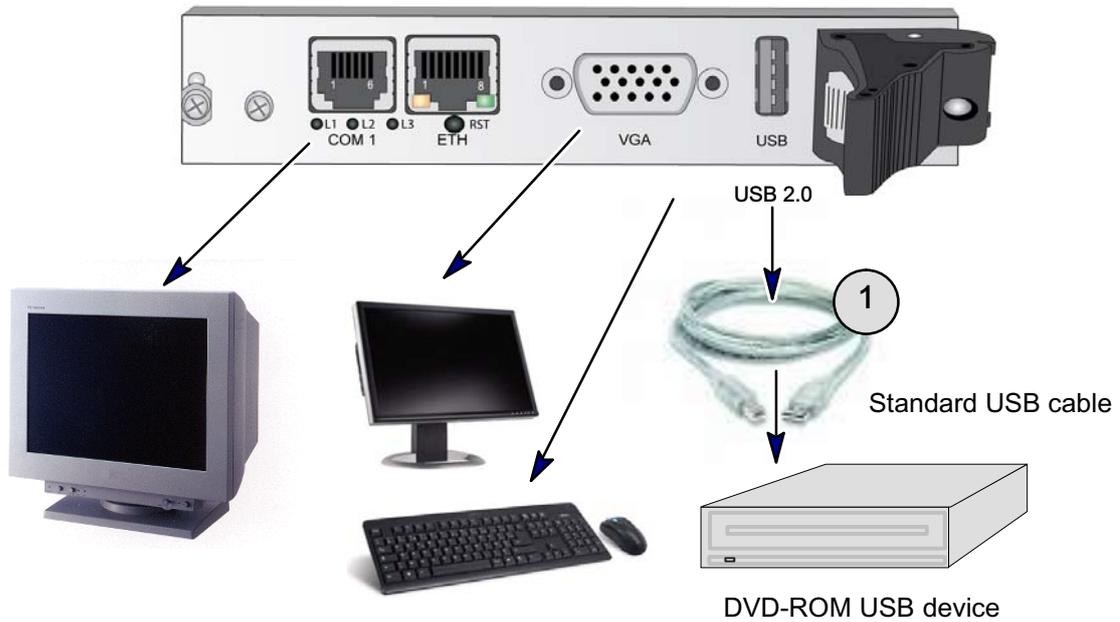
4.4.1 DVD-ROM Installation Example for VX6060

1. Plug a DVD-ROM USB device to the front panel USBA connector using a standard USB cable.
2. Plug the HDD SATA device on the SATA0-A connector of the Rear Transition Module (VX6060-RTM).



4.4.2 DVD-ROM Installation Example for VX3030

1. Plug a DVD-ROM USB device to the front panel USB connector using a standard USB cable.
2. Plug the HDD SATA device on the SATA0-A connector of the Rear Transition Module (VX3030-RTM).



Chapter 5 - Installation

This chapter describes the specific steps of the installation process of the Fedora 12 on VX6060 and VX3030 boards.

For the VX6060, and because this board is a dual subsystem board, two main installation methods are described hereafter:

➤ **A dual disk configuration (sections 5.1.1 and 5.1.3):**

A boot SATA disk on CPUA and another boot SATA disk on CPUB. With this configuration, CPUA and CPUB are quite independent. The communication between the two subsystems may be done through the local network switch or through the VXFabric (refer to section 5.1).

➤ **A CPUB diskless configuration (section 5.2):**

CPUA is configured as a disk server for CPUB. CPUB boot through the network using PXE/NFS setup and the "multinodes-diskless" package provided with the BSP (refer to section 5.2 page 11)

5.1 Disk Installation

5.1.1 Fedora 12 Installation on CPUA of VX6060 and on VX3030

This section describes the installation procedure from an USB DVD-ROM drive.

There is no major difference between the installation on the VX6060 and VX3030 and the standard Fedora 12 installation, so refer to the Fedora documentation to get more details on all the Fedora installation menus.

Nevertheless, the VX6060 or VX3030 is a graphic board, so the installation may be done in graphic mode or in text mode on the serial port 0.

1. Insert the DVD Fedora 12 in the DVD-ROM driver.
2. After a board reset, or a board power-on, Type <F7> to get the the Boot Manager Menu from BIOS and to select the DVD-ROM device as the boot device.
3. Select the menu:
 - FOR A GRAPHIC INSTALLATION: **Install system with basic video driver**
 - OR
 - ➤ FOR A SERIAL CONSOLE INSTALLATION: **Install or upgrade using SERIAL console**
4. Then the standard Fedora 12 Installer menus from anaconda should be displayed.
5. Proceed as a standard Fedora 12 installation.
6. Note that the Mezzanine USB Flash device, if present, is probed as another sata drive.
7. At the end of the installation, reboot on the installed disk drive through the Boot Manager Menu <F7>. The BIOS Setup menu should be used to set the boot devices priorities. Refer to the BIOS Manual <SD.DT.F69>.



In SERIAL CONSOLE INSTALLATION, the Fedora installer anaconda installs a sub-set of packages reduced at 200 rpms and the user menus are not providing the same level of install options than graphic 's one. Depending on the needed services and libraries, it may require some post installations.

So it is recommended to perform a network installation (with or without kickstart) in serial console configuration and to use VNC protocol to have access to the full options of the Fedora Installation through a distant network display.

The DVD menu: "Install or upgrade using VNC" may be also used if a DHCP server is available to provide IP configuration to the target.

5.1.2 BSP Installation on CPUA of VX6060 and on VX3030

The BSP CD-ROM contains only the BSP specific packages as well as the related source packages.

Follow the procedure below, to install the BSP version 1.0 onto an existing bootable media (SATA disk or USB flash) or to access to the source packages.

1. Insert the Kontron BSP CD-ROM Fedora 12 in the USB DVD-ROM drive.
2. Boot the VX6060 board on the SATA disk previously installed with the Fedora 12.
3. Login as root and run following commands:

```
[root@vx6060 ~]# mount /dev/sr0 /mnt
[root@vx6060 ~]# cd /mnt
[root@vx6060 ~]# ./install.sh -i
```



If diskless tool is not used, do
[root@vx6060 ~]# chkconfig diskless off
to avoid the service to be started at boot time.

The installation process will take few minutes.

4. Reboot the machine to take into account the new version of the BSP.

5.1.3 VX6060 - Fedora 12 Installation on CPUB

The serial console of CPUB is available through the COM-AB connector on the front panel. It requires specific cable and connector to get the serial console for CPUA and CPUB. Refer to your Kontron representative to get the information about such a cable.

1. Plug a DVD-ROM USB device to the front panel USBA/B connector using a standard USB cable.
2. Check in the BIOS setup menu the USBA/B is dedicated to the CPUB.
3. Plug the HDD SATA device on one of the SATA0-B connector of the Rear Transition Module (VX6060-RTM).

Then boot on the DVD-ROM USB and the installation process is similar to the installation steps on CPUA: section 5.1.1 page 9.

It is possible to configure the CPUB in graphical mode using the dedicated rear backplane display-port connection. Refer to the Hardware User Manual to get more information or Contact your Kontron representative.



In SERIAL CONSOLE INSTALLATION, the Fedora installer anaconda installs a sub-set of packages reduced at 200 rpms. Depending on the needed services and libraries, it may require some post installations.

So it is recommended to perform a network installation (with or without kickstart) in serial console configuration and to use VNC protocol to have access to the full options of the Fedora Installation through a distant network display.

5.1.4 VX6060 - BSP Installation on CPUB

Check in the BIOS setup menu that the USBA/B is dedicated to the CPUB.

Boot on the previously installed Fedora 12 SATA disk.

Then insert the BSP CDROM in the DVD-ROM driver and follow the step of the section 5.1.2.

5.2 VX6060 - CPUB Diskless Installation

The CPUA and CPUB of the VX6060 are connected through an Ethernet switch so a network boot (through BIOS PXE) may be setup to get a diskless linux system on the CPUB with the CPUA as a server.

The package multinodes-diskless is available to setup and configure a NFS exportable root filesystem and to get a kernel and a initrd bootable through PXE.

The following steps describe how to initialize a diskless configuration for the CPUB and how to boot it.

0. The very first step is to get a local repository on CPUA of the Fedora 12 rpm packages.

Insert the Fedora 12 DVD and mount it:

```
# mkdir -p /dist/Fedora12_VX6060/{BSP,RPMS}
# mount /dev/dvd /mnt
# cp /mnt/Packages/* /dist/Fedora12_VX6060/RPMS
# cp /usr/share/vx6060_bsp/bsp_11068/RPMS/*.rpm /dist/Fedora12_VX6060/BSP
# cd /dist/Fedora12_VX6060/
# createrepo .
```

1. Install the multinodes-diskless package.

The standard installation (refer to previous chapters) of the BSP provides the multinodes-diskless package with. Nevertheless, if it is not installed, the following command will install it:

```
# yum install --enablerepo=vx6060-bsp-11257 multinodes-diskless
```

2. In the following, the local sub-network of the diskless is 192.168.1. This may be adapted to the network constraints.

3. Configure the ETH1 on the CPUA as the server ethernet interface with the IP address 192.168.1.1

This may be done through the command system-config-network or by editing the file:

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth1

# Network Interface
DEVICE=eth1
ONBOOT=yes
NETMASK=255.255.255.0
IPADDR=192.168.1.1
BOOTPROTO=none
TYPE=Ethernet
NM_CONTROLLED=no
IPV6INIT=no
USERCTL=no
```

Then setup the interface:

```
# ifup eth1
```

Check it is properly done:

```
# route
```

4. Copy the diskless configuration files provided with the BSP:

```
# cp /usr/share/vx6060_bsp/diskless_fs.xml /etc/diskless/
# cp /usr/share/vx6060_bsp/images.xml /etc/diskless/
# chmod +w /etc/diskless/*.xml
```

5. **WARNING:** In the file `/etc/diskless/diskless_fs.xml`, remove the lines:

```
<abs>[ -f /root/.ssh/id_rsa.pub ] || ssh-keygen -y</abs>
<abs>mkdir -p %(rootdir)/root/.ssh </abs>
<abs>cp -f /root/.ssh/id_rsa.pub %(rootdir)/root/.ssh/authorized_keys
</abs>
```

If present.

6. Edit the file `/etc/diskless/image.xml` and modify the MAC addresses accordingly with the MAC address of the CPUB.

The MAC address of the CPUB is displayed by the EDI shell command:

```
VX6060-B> kmac -r
Read data from flash at 0x2000 size 0x6
MAC Address = 00:00:DE:40:36:F9
```

Set this address in the file `/etc/diskless/image.xml`:

```
...
<nodemap id="diskless-cluster_A" clients="diskless_A" default="1">
  <match criterium="macaddress">
    <value>00:00:DE:40:36:F9</value>
  </match>
  <ip>192.168.1.%(1+order.value)</ip>
  ...
```

7. Stop the iptables service and disable the selinux by running the commands :

```
# chkconfig iptables off
# service iptables stop
```

8. Set the variable `SELINUX=disabled` into the file `/etc/selinux/config`

9. Start the required service: `tftpd` and `nfs` by running the commands :

```
# chkconfig tftpd on
# chkconfig nfs on
# service nfs start
```

Those services may required to be installed first.

10.Reboot the server in order to take into account these changes.

11.To configure the exportable root file system on the server:

```
# mkdir -p /diskless/layers.  
# diskless -B
```

This will build the filesystem /diskless/layers/

This process will take few minutes.

12.Configure the dhcpd server as well as the PXE boot by :

```
# diskless -x
```

13.Export the diskless RFS by :

```
# diskless -e
```

14.On the CPUB and after a power-on, take the hand under CPUB by pressing F2.

Under BIOS Setup, select successively the menus :

Chipset->South Bridge -> Ibex Peak Option

And enable the PXE boot.

Save the changes and after reset, press again <F2>.

Select the menu "boot" and put the network device at the top of the list.

Save the change and exit.

As a result the PXE boot will happen and something like that should be displayed :

```
CLIENT MAC ADDR: 00 00 DE 40 36 F9  GUID: 00020003 0004 0005 0006  
000700080009  
DHCP. |
```

You should notice the occurrence of a login at the end of the boot step.

Chapter 6 - Fedora System Configuration

In this chapter, informations related to some specific configuration items of the Fedora 12 system are detailed.

6.1 Network Manager

With Fedora 12, the network interfaces are managed by the NetworkManager.

For an embedded system, it is hardly recommended to use the older network service instead which is easier to configure through configuration files.

For this:

- > Disable the NetworkManager service:

```
# chkconfig NetworkManager off
```

- > Enable the network service:

```
# chkconfig network on
```

- > Run `system-config-network`, then for each interface:
 - ▶ click on edit
 - ▶ unselect Controlled by NetworkManager
- > Exit from `system-config-network` saving changes
- > Reboot.

6.2 Network Interfaces

» udev Rules for VX6060 Network Interfaces

The BSP of the VX6060 comes with udev rules to define the name of the network interfaces of the VX6060.

On CPUA:

- > the `eth0` (i82577) is connected to RJ45 "ETH-A" front panel connector for SA Board and to the Geth switch for RC Board. This is configurable through BIOS setup.
- > the `eth1` and `eth2` (i82580) are connected to onboard Giga-ethernet switch.

On CPUB:

- > `eth0` (i82577), `eth1` and `eth2` (i82580) are connected to the onboard Geth switch.

One port of the Geth switch is available on the front panel connector of SA board: RJ45 "ETH-SW" connector. Three SERDES 1000BASE-BX ports from the Giga-ethernet switch are available on the backplane through P1 connector.

One 1000BASE-T port from the gia-ethernet switch is also available from P1 and routed to the rear module.

Refer to the User Manual of this board to get some more information about network interfaces.

» udev Rules for VX3030 Network Interfaces

The BSP of the VX3030 comes with udev rules to define the name of the network interfaces of the VX3030:

- the **eth0** (i82577) is connected to RJ45 "ETH-A" front panel connector for SA Board and to the Geth switch for RC Board. This is configurable through BIOS setup.
- the **eth1** and **eth2** (i82580) are routed to the backplane through P1 connector.

Refer to the User's Guide of the Board to get more information about the Ethernet connection.

» MAC Address

By default, the MAC address is stored with the configuration parameters of each interface. If the MAC address of a device is found different from the one expected (board changed for example), the interface is not brought up. This is not suitable for an embedded system when boards must be changed for maintenance without requiring additional configuration.

To workaround this behavior, do not bind an Ethernet interface to a MAC address:

- Run `system-config-network`
- For each interface:
 - ▶ click on Edit
 - ▶ click on Hardware Device tab
 - ▶ unselect Bind to MAC address
- Exit from `system-config-network` saving changes

This can be done also by editing the `/etc/sysconfig/network-scripts/ifcfg-eth*` files and removing the `HWADDR` lines but only if the service network (and not NetworkManager) is used.

» Renaming Interface

Additionally udev binds the Ethernet device number to the MAC address: each time an interface with a new MAC address is detected (MAC address not previously used with this installation of Fedora), it is assigned a new device number `<n>`, so a new device `eth<n>`.

This means that:

- After the Fedora installation, onboard Ethernet devices are named `eth0`, `eth1`, `eth2` and `eth3`.
- If the board is changed, onboard Ethernet devices become `eth4`, `eth5`, ... and so on.
- The network has to be reconfigured each time a new board is used, because the network parameters for the new interfaces must be set to have a functional network.
- Until the network is reconfigured, the only way to connect to the board is through its console (graphics or serial line).
- An operator is required to reconfigure the network.

More precisely, when the Ethernet driver is loaded, it creates the Ethernet devices as they are probed. If the only Ethernet devices present are the onboard ones, they are always named `eth0` and `eth1` regardless of their MAC address. These interfaces are then renamed by udev to have a specific device number according to their MAC address. Some information about that can be found in the output of the command `dmesg`:

```
udev: renamed network interface eth0 to eth2
udev: renamed network interface eth1 to eth3
```

This behavior is not suitable for an embedded system where boards must be changed for maintenance without requiring additional configuration.

To disable this renaming behavior, do as follows:

1. Edit `/lib/udev/rules.d/75-persistent-net-generator.rules` to remove `eth*` from the device white list.

This is done by changing the following line:

```
KERNEL!="eth*|ath*|wlan*[0-9]|msh*|ra*|sta*|ctc*|lcs*|hsi*",
GOTO="persistent_net_generator_end"
```

to

```
KERNEL!="ath*|wlan*[0-9]|msh*|ra*|sta*|ctc*|lcs*|hsi*",
GOTO="persistent_net_generator_end"
```

2. Remove `/etc/udev/rules.d/70-persistent-net.rules` if existing:

```
# rm -f /etc/udev/rules.d/70-persistent-net.rules
#
```

If a new Ethernet device is added to the system, the device numbers may vary (depending on the order they are probed by the driver).

» Firewall

If the firewall must be disabled but has been enabled during the installation:

- > Run `system-config-firewall`
- > Click on `disable` and `exit`
- > Make sure to disable the `iptables` service by running:

```
# chkconfig iptables off
# chkconfig ip6tables off
#
```

- > Reboot

» SELinux

SELinux stands for Security-Enhanced Linux. The Security-Enhanced Linux kernel enforces mandatory access control policies that confine user programs and system servers to the minimum amount of privilege they require to do their jobs.

If you experience some trouble running some services or have some permission issues, try to set the System Default Enforcing Policy to Permissive instead of Enforcing by running the `system-config-selinux` tool.

6.3 cpuspeed

cpuspeed is a program that monitors the system's idle percentage and reduces or raises the CPU's clock speeds and voltages accordingly to minimize power consumption when idle, and maximize performances when needed. This is mainly useful on laptop and may not be suitable for embedded systems.

For embedded system, the "performance" policy (maximum performances) should be set in most of the case.

To set up the mode, edit the file `/etc/sysconfig/cpuspeed` and modify the line:

```
GOVERNOR=
```

by

```
GOVERNOR=performance
```

Then the service should be restarted:

```
# service cpuspeed restart
```

Chapter 7 - BSP Specific Features

7.1 Sensors

The BSP contains an RPM named `sensors_addons` that configures the standard `lm_sensor` software for the VX6060 & VX3030.

To display sensors information:

» For VX6060

```
[root@node1 ~]# sensors
acpitz-virtual-0
Adapter: Virtual device
temp1:      +38.0 °C (crit = +100.0 °C)
temp2:      +38.0 °C (crit = +100.0 °C)

lm73-i2c-6-48
Adapter: I2C CPLD adapter
LM73 sensor Temperature: +38.0 °C (low = -30.0 °C, high = +70.0 °C)

lm73-i2c-6-49
Adapter: I2C CPLD adapter
LM73 sensor Temperature: +43.0 °C (low = -30.0 °C, high = +70.0 °C)

lm73-i2c-6-4a
Adapter: I2C CPLD adapter
LM73 sensor Temperature: +42.0 °C (low = -30.0 °C, high = +70.0 °C)

ads7830-i2c-5-4b
Adapter: SMBus I801 adapter at e000
12V VPX:    +11.66 V
5V VPX:     +4.95 V
3V3:        +3.30 V
2V5 6U:     +2.54 V
3V3 VPX SB: +3.24 V
1V05S 3U:   +1.06 V
1V05 3U IBEX: +1.04 V
1V 3U:      +1.00 V
```



A BIOS \geq 11067 is required to get the ACPI temp1 and temp2.

» For VX3030

```
[root@lnx4 ~]# sensors
acpitz-virtual-0
Adapter: Virtual device
temp1:      +36.0°C (crit = +100.0°C)
temp2:      +36.0°C (crit = +100.0°C)

lm73-i2c-6-48
Adapter: I2C CPLD adapter
  LM73 sensor Temperature: +42.0°C (low = -30.0°C, high = +70.0°C)

ads7830-i2c-5-4b
Adapter: SMBus I801 adapter at e000
12V VPX:    +12.13 V
5V VPX:     +5.06 V
3V3:        +3.30 V
2V5 6U:     +2.48 V
3V3 VPX SB: +3.28 V
1V05S 3U:   +1.05 V
1V05 3U IBEX: +1.05 V
1V 3U:      +1.00 V
```

7.2 Watchdog

The BSP 2.2 provides a driver for the watchdog of the CPLD.

To check that the driver is loaded:

```
[root@lnx4 ~]# lsmod | grep cpld_wdt
cpld_wdt          3138  0
...
```

This driver can be loaded with two options: timeout of the watchdog (timeout) and the mode (trigger_mode).

By default, timeout is 21s and the trigger_mode is reset mode which means that if the watchdog falls, the board is reseted.

To change the timeout, reload the driver with the required timeout:

```
# rmmod cpld_wdt
# modprobe cpld_wdt timeout=11
```

NOTE: only odd timeout value are available.

To change the mode to get a interruption instead of a reset:

```
# rmmod cpld_wdt
# modprobe cpld_wdt timeout=11 trigger_mode=2
```

To wait on the interruption, the read system call can be used:

```
# echo Y > /dev/watchdog ; time cat /dev/watchdog
```



If no one is waiting on the interruption, when the watchdog falls, a emergency reboot system call is run.

7.3 VPD Tool

vpdtool command is useful to get information related to the VX6060 & VX3030 boards. The following examples are given for the VX6060 board:

```
# vpdtool
VX6060 detected
Board type   : VX6060-2SA24-0000
EC Level    : 01000
Serial Number: 1110271030014
Variant     : 0010043801100001
Keylist                                           :
/PCB_A/SACCLASS/MEZZUSB/BHDUAL/ARLDV2G/IBOMON/ETHSTD6U/1GB_DDR3_1067/DDR3-1067/IRTC/XDPON/STDC
LK/CK505REFOSC/STD_EARTH/BATON/HDDON/PXMC/2G5TS/P1PWRMAGOFF/PWRMAGOFF/FP4220FF/TMLVDSOFF/I2CS
TD/I2CAB/1SLOT/STDVPXKEY/P800FF/VS20FF/NOSYNACAB/ITIN/BC/
```

To get the features list:

```
# vpdtool --human
VX6060 detected
Board type   : VX6060-2SA24-0000
EC Level    : 01000
Serial Number: 1110271030014
Variant     : 0010043801100001
Keylist
/PCB_A/SACCLASS/MEZZUSB/BHDUAL/ARDLV2G/IBOMON/ETHSTD6U/1GB_DDR3_1067/DDR3-1067/IRTC/XDPON/STDC
LK/CK505REFOSC/STD_EARTH/BATON/HDDON/PXMC/2G5TS/P1PWRMAGOFF/PWRMAGOFF/FP4220FF/TMLVDSOFF/I2CS
TD/I2CAB/1SLOT/STDVPXKEY/P800FF/VS20FF/NOSYN CAB/ITIN/BC/

Features   :
PCB A
SA Class
USB mezzanine slot configuration
Dual link NH82580
Arrandale ECC LV 2GHz
Force extended temperature range BOM generation
6U basic Geth switch configuration, 1000Base-T link on P1
1GB DDR3-1067 device
DDR3-1067 dram speed
PCH internal RTC configuration
XDP port available
Standard clock option
CK505 oscillator source configuration
Standard EARTH connection
Battery present
on-board SATA HDD option ON
PMC P64s, XMC X8d+X12d IOs available on VPX P3/P4 connectors
Internal PCIe link speed set to 2.5GT/s
P1 magnetic power option off
Internal magnetic power option off
```

```
Front panel RS-422 option off
MLVDS terminations off
CPLD I2C equipment: one F-RAM, one EEPROM
CPLD A I2C linked to CPLD B I2C
One VPX slot version for SA
VPX Keying set to universal 6U VPX (315 degrees keying)
P80 debug option not available on debug connector
VS2 PWR pins not used
No Sync A/B option
PCB plating: Immersion Tin
Use of Backward compatible lead free component
```

For the VX3030 board:

```
[root@lnx4 ~]# vpdtool
VX3030 detected
Board type   : VX3030-SA22-01000
EC Level    : 01000
Serial Number: 1110431010002
Variant     : 0010042010020001
Keylist
/PCB_A/SACCLASS/MEZZUSB/BHDUAL/ARLDV2G/IBOMON/ETHSTD6U/1GB_DDR3_1067/DDR3-1067/IRTC/XDPON/STDC
LK/CK505REFOSC/STD_GND/BATON/HDDOFF/NOMEZZIO/2G5TS/P1PWRMAGOFF/PWRMAGOFF/FP4220FF/TMLVDS0N/I2
CSTD/I2CA/1SLOT/STDVXPKEY/P800N/VS20FF/NOSYN CAB/ITIN/BC/
```

7.4 LEDs

The driver `leds_cpld` allows to deal with the front panel LEDs for a user mode.

The driver `leds_cpld` creates a list of special files and classes in `/sys`:

```
# cd /sys/class/leds/
[root@lnx8 leds]# ls
led1:amber led1:OFF led2:amber led2:OFF led3:amber led3:OFF
led1:green led1:red led2:green led2:red led3:green led3:red
```

For each LED, there are three different colors available: green, red, amber that can be set by addressing the related file.

For each color (which are exclusive), there are four different modes:

- ▶ ON (echo 0, see following example)
- ▶ low blinking (echo 1, see following example)
- ▶ fast blinking (echo 2, see following example)
- ▶ OFF (echo 1, see following example)

Example, to set those different modes on the LED 2 in amber:

```
# echo 0 > led2:amber/brightness The LED 2 is ON in AMBER
# echo 1 > led2:amber/brightness The LED 2 blinks low in AMBER
# echo 2 > led2:amber/brightness The LED 2 blinks fast in AMBER
# echo 1 > led2:OFF/brightness The LED 2 is OFF
```

- VX6060: On the front panel, the name of the led1 on CPUA is L1, led2 is L2 and led3 is L3. On CPUB, led1 is L4, led2 is L5 and led3 is L6
- VX3030: On the front panel, the name of the led1 on VX3030 is L1, led2 is L2 and led3 is L3.



The led1 (L1 and L4) is not manageable at user level. So do not use led1 special files.

7.5 Allocator

Allocator is a module allowing the allocation of large contiguous memory areas. Allocator module may be needed to map VME bus windows or to get big bunch of memory for DMA transfers.

Refer to the man page allocator to get detailed information about this module.

Edit the file `/etc/init.d/allocator` in order to set the size of the different areas needed.

VXFabric (refer to section 7.6) used to depend on allocator but with BSP newer than 2.0 there is not more dependencies between allocator and VXFabric.

7.6 VXFabric

The VXFabric tools and kernel modules are provided to get communication over the PCIe between CPU nodes. There is two main PCIe express domains which may be dedicated to VXFabric:

- Backplane VPX PCIe ports (#0)
- Onboard VX6060 PCIe between CPUA and CPUB (#1) - Only on VX6060

Each nodes on a VXFabric domain represents a board (or a subsystem in the VX6060 case) on the PCIe bus and gets a unique Node ID used to identify it.

There is two main ways to communicate on VXFabric domains: low level protocole, and IP over VXFabric.

The following chapters give some inputs about VXFabric. There is also the different man pages available on the system to complete the information about it:

- `vx` (4) - VX fabric for DMA over PCIe on VX6060 with PLX8609
- `vxdma` (3) - DMA over PCIe fabric API
- `vxfabric` (1) - utility to perform DMA over PCIe
- `vxfabric_api` (3) - PCIe fabric API

On the 64 bit BSP, a VXFabric service has been added. To start the service:

```
[root@lnx8 ~]# service vxfabric start
```

and to stop it:

```
[root@lnx8 ~]# service vxfabric stop
```

To get this service configured at every boot time:

```
[root@lnx8 ~]# chkconfig --add vxfabric
```

7.6.1 The vxfabric Command

The vxfabric command may be used to get the status of the different nodes on the VXFabric domains and to exercise the low level communication (see next chapter) between nodes.

```
[root@evalpasemi ~]# vxfabric --help
vxfabric: unrecognized option: --help
Usage vxfabric -S <SrcAddr> [-s <SrcType>] -D <DstAddr> [-d <DstType>] [-B <Size>] [-L
<LoopNum>] [-P] [-R] [-C <ChanNum>]
Options are :
[-a] : Give status of the whole nodes or of a particular node (with -N)
[-S <AddressValue>] : Set the Physical SOURCE address ( PCI or MEMORY area )
[-s <AddressType> ] : Set the SOURCE address type, a value among :
    0 => Physical MEM or PCI Area ( requests -S <AddressValue>)
    1 => Allocated Virtual MEM Area
    2 => Allocated Phys MEM Area
[-D <AddressValue>] : Set the Physical DESTINATION address ( PCI or MEMORY area )
[-d <AddressType>] : Set the DESTINATION address type ( i.e SOURCE address type )
[-B <SizeValue>] : Set the Size of the transfer ( DEFAULT value = 1MB)
[-L <LoopNum>] : Set the number of loops of the transfer ( DEFAULT value = 1
[-P ] : Select the Polling Mode ( DEFAULT value = FALSE)
[-R ] : Select the Round Robin Mode ( DEFAULT value = FALSE)
[-C <ChanNum> ] : Specify the channel number to be used ( DEFAULT value = 0)
[-F <FabricBusNum> ] : Specify the Fabric bus number to be used ( DEFAULT value = 0)
[-M <DMAEngineNum> ] : Specify the DMA engine number to be used ( DEFAULT value = 0)
```

```
[-c ] : Enable the control of the transferred DATA
[-E ] : READ the binary content of the EEPROM into a file
[-e ] : READ the binary content of the EEPROM and convert it into an ASCII format
[-W ] : WRITE the content of a binary DATA file into the EEPROM
[-w ] : Interpret a Command FILE , convert it to binary DATA and WRITE it to the EEPROM
[-n ] : Specify the source NODE ID
[-N ] : Specify the destination NODE ID or the NODE ID containing the EEPROM ( DEFAULT = 2
)
[-f ] : Specify the FILE NAME which will contain the DATA of the EEPROM ( DEFAULT =
/tmp/EEPROMBinFile )
[-r ] : Perform a simple read to a PCIexpress address and evaluate the latency
[-dump] : args passed to dump routine. see -dump -h
[root@evalpasemi ~]#
```

The following command gives a status of the VXFabric on the VPX backplane.

```
[root@lnx4 ~]# vxfabric -a
-----
VX FABRIC 0 configuration
-----
N H S S C S P P M A ( I A (
O O L T P Y C C E R S 0 R S
D S O A U S I I M E I E I
E T T T C B D O A Z A Z
I U O U E R E E
D S N S V Y ) )

1 * 1 READY A * 0x14001000(0x3fff000) ( )
3 2 READY A 4 0 0xf0000000(0x4000000) 0xf7c00000(0x20000)
```

On the local VXFabric (for VX6060 between CPUA and CPUB):

```
[root@board25 ~]# vxfabric -a -F1
-----
VX FABRIC 1 configuration
-----
N H S S C S P P M A ( I A (
O O L T P Y C C E R S 0 R S
D S O A U S I I M E I E I
E T T T C B D O A Z A Z
I U O U E R E E
D S N S V Y ) )

1 * 1 READY A * 0x14000000(0x4000000) ( )
2 1 FIRM B 6 1 0xf0000000(0x4000000) 0xf7200000(0x20000)
```

The command VXfabric may be used to exercise and to test the link between nodes.

```
# vxfabric -n1 -N2 -F1 -S 0x0 -D 0x0 -c
Getting the SOURCE DMA buffer address (NODEID AREA) Size = 0x100000
Status of TARGET NODEID 1 = READY
PCI MEM AREA addr = 0x14000000
Initializing the SOURCE DMA buffer
NODEID 1 offset = 0x0 Getting the DESTINATION DMA buffer address (NODEID AREA) Size = 0x100000
Status of TARGET NODEID 2 = READY
PCI MEM AREA addr = 0xec000000
NODEID 2, offset = 0x0
Perform DMA transfer Size = 0x100000
Write of 1048576 bytes in 2030 usec (516 MB/s)
Comparing Data: DATA Control OK
```

Refer to the man pages of VXfabric to get some more detailed information.

7.6.2 Low Level access

Low Level protocol over VXFabric is based on standard `read()` and `write()` to specific character device files `/dev/vx_rnodeX` (`/dev/vx_CPUA` or `/dev/vx_CPUB`). `read()` and `write()` system calls will be DMAs over VXFabric.

The system call `poll()` may be used to wait for interruptions over VXFabric and generated through a specific `ioctl` (`VXFABRIC_IOCTL_INTR_SEND`).

The `mmap()` is also provided to PIO access to the VXFabric memory area of a distant node.

The `write()` should be privileged over the `read()` because it is much faster and less CPU

> Known issue:

- `read()` from CPUB to CPUA is not working.

The `lseek()` system call should be used to change the address in the reception buffer of the target for the next read or write. The reception buffer for VXFabric #0 is 64 MB on each board of the VXFabric.

There is also another 64 MB reception buffer for VXFabric #1 (between CPUA and CPUB).

With the BSP delivery, there are samples C code files which can be used and adapted to the customer application. Refer to `/usr/share/vxfabric/examples` repository.

7.6.3 IP over VXFabric

The User Programming model of VXFabric is based on the IP protocol and a socket layer API through an emulation of an Ethernet interface over PCIe like similar implementations of pseudo-Ethernet over virtualization boxes. This is the main warranty that the portability to VXFabric is straight forward for software developers.

The objective is to allow a user to move from Gbit Ethernet TCP/IP implementation to VXFabric avoiding low level complex and proprietary APIs (like most of sRIO implementation for example) with a breakthrough of performances, if compared with Gbit Ethernet.

VXFabric is available under Linux for VX6060 and VX3030 and it is designed to be portable on other architectures and other operating systems. This implementation is scalable and will not require much effort to evolve towards PCIe gen3.

Furthermore, it does not require any other infrastructure than the VPX backplane and a VPX PCIe switch to interconnect VPX boards.



IP over VXFabric when it is setup, is disabling the "low level" protocols described in the previous chapter.

» VXFabric Setup

Once the system slot1 VX6060 has booted Linux operating system, few configuration steps are required to get ready the Ethernet emulation over VXFabric.

The first step is to get some low level information about the VXFabric through the CLI command VXFabric:

```

vxfabric -a
-----
VX FABRIC 0 configuration
-----
N H S S C S P P M A ( I A (
O O L T P Y C C E R S 0 R S
D S O A U S I I M E I E I
E T T T C B D O A Z A Z
I U O U E R E E
D S N S V Y ) )

1 * 1 READY A * 0x14001000(0x3fff000) ( )
2 1 READY B 6 0 0xa0000000(0x4000000) 0xf5c00000(0x20000)
5 3 READY A 16 0 0xb8000000(0x4000000) 0xf6000000(0x20000)
6 3 READY B 15 0 0xb4000000(0x4000000) 0xf5f00000(0x20000)
11 6 READY A 7 0 0xa8000000(0x4000000) 0xf5d00000(0x20000)
12 6 READY B 8 0 0xac000000(0x4000000) 0xf5e00000(0x20000)
-----
    
```

After the boot of all the boards on the VXFabric, all the boards reach the READY status. The vxeth module, dealing with the Ethernet emulation, is not loaded by default to avoid the resources to be stolen when they are not needed. To load it:

```
# modprobe vxeth
```

This leads to the creation of two new pseudo-ethernet interfaces: vxeth0 and vxeth1 which is similar to a standard ethX ethernet interface. Those interfaces are dealing with the VXFabric #0 for the backplane and VXFabric #1 for the local communication between CPUA and CPUB (only on the VX6060).

The vxeth0 is dedicated to the backplane vxfabric and vxeth1 should be used to communicate between CPUA and CPUB of the VX6060.

```

client9# ifconfig vxeth0
vxeth0      Link encap:Ethernet  HWaddr 02:00:00:00:00:01
            BROADCAST MTU:65500  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
    
```

Those pseudo-Ethernet interfaces should be managed like the other Ethernet interfaces and a local sub-network address could be allocated to this interface. For example:

```
# ifconfig vxeth0 192.168.20.1
```

This is not different from other Ethernet interfaces configuration and, for example, on Fedora distribution, the service «network» can be used to setup this interface through the configuration file /etc/sysconfig/network-scripts/ifcfg-vxeth0.

On another node, the same previous steps should be performed with another IP address on the same local sub-network address.

```
client5# ifconfig vxeth0 192.168.20.2
client5# ifconfig vxeth0
vxeth0      Link encap:Ethernet  HWaddr 02:00:00:00:00:07
            inet addr:192.168.20.2  Bcast:192.168.20.255  Mask:255.255.255.0
            inet6 addr: fe80::ff:fe00:9/64 Scope:Link
            UP BROADCAST  MTU:65500  Metric:1
            RX packets:4803815 errors:0 dropped:0 overruns:0 frame:0
            TX packets:7190431 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:367044680 (350.0 MiB)  TX bytes:92151086 (87.8 MiB)
```

It is possible to control the status of the interfaces through the netstat command:

```
client5# netstat -i
Kernel Interface table
Iface      MTU Met  RXOK    RXERR   RXDRP   RXOVR   TXOK    TXERR   TXDRP   TXOVR Flg
bond0      1500 0    382872  0       0       0    391990  0       0     0 BMmRU
eth0       1500 0    204225  0       0       0    206587  0       0     0 BMsRU
eth1       1500 0    178647  0       0       0    185403  0       0     0 BMsRU
eth2       1500 0      0       0       0       0      0       0       0     0 BMU
vxeth0     65500 0    5372700  0       0       0   18477441  0       0     0 BMRU
lo         16436 0      36      0       0       0      36      0       0     0 LRU
```

The hosts IP addresses of the previous nodes could be added in the `/etc/hosts` file to be populated.

```
client5# more /etc/hosts
# hostname vx6060 added to /etc/hosts by anaconda
127.0.0.1          localhost          localhost.localdomain  localhost4
localhost4.localdomain4
192.168.1.1       server
192.168.20.1      node1
192.168.20.2      node2
192.168.20.3      node3
192.168.20.4      node4
192.168.20.5      node5
192.168.20.6      node6
### netconfig updates ###
192.168.1.12      client12
192.168.1.11      client11
192.168.1.10      client10
192.168.1.7      client7
192.168.1.6      client6
192.168.1.5      client5
192.168.1.4      client4
192.168.1.3      client3
192.168.1.2      client2
192.168.1.9      client9
192.168.1.8      client8
### netconfig ends ###
```

The interconnection could be tested through the ping command:

```
client5# ping node2
PING node2 (192.168.20.2) 56(84) bytes of data.
64 bytes from node2 (192.168.20.2): icmp_seq=1 ttl=64 time=0.022 ms
64 bytes from node2 (192.168.20.2): icmp_seq=2 ttl=64 time=0.029 ms
64 bytes from node2 (192.168.20.2): icmp_seq=3 ttl=64 time=0.020 ms
64 bytes from node2 (192.168.20.2): icmp_seq=4 ttl=64 time=0.019 ms
64 bytes from node2 (192.168.20.2): icmp_seq=5 ttl=64 time=0.022 ms
```

When every node on the VXFabric is configured with a valid internet address, the CLI command VXFabric output should be:

```
vxfabric -a
-----
VX FABRIC 0 configuration
-----
N H S S C S P P M A ( I A (
O O L T P Y C C E R S 0 R S
D S O A U S I I M E I E I
E T T T C B D O A Z A Z
I U O U E R E E
D S N S V Y ) )

1 * 1 CONNECT A * 0x14001000(0x3fff000) ( )
2 1 CONNECT B 6 0 0xa0000000(0x4000000) 0xf5c00000(0x20000)
5 3 CONNECT A 16 0 0xb8000000(0x4000000) 0xf6000000(0x20000)
6 3 CONNECT B 15 0 0xb4000000(0x4000000) 0xf5f00000(0x20000)
11 6 CONNECT A 7 0 0xa8000000(0x4000000) 0xf5d00000(0x20000)
12 6 CONNECT B 8 0 0xac000000(0x4000000) 0xf5e00000(0x20000)
-----
```



- ▶ The broadcast addressing is emulated by software.
- ▶ The MTU is set by default to 65500 bytes to get benefit of the maximum bandwidth.

» Standard Network Services

The standard network services are supported «de facto» by the VXFabric due to the Ethernet emulation. After the setup is done as described in the previous chapter and modulo specific configuration and setup of each service, the major network services are available: ftp, telnet, ssh, NFS, scp...

Example:

```
client5# ssh node3
Warning: Permanently added the RSA host key for IP address '192.168.20.3' to the
list of known hosts.
Last login: Thu Nov 25 11:04:51 2010 from server
client7#
client7# scp /boot/vmlinuz2.6.32.1410176.vx6060.fc12.i686.PAE node2:/tmp
Warning: Permanently added the RSA host key for IP address '192.168.20.1' to the
list of known hosts.
vmlinuz2.6.32.1410176.vx6060.fc12.i686.PAE          100% 3565KB   3.5MB/s
00:00
client7#
logout
client5# ll /tmp
total 3568
-rwxr-xr-x  1  root  root   3650176   20101125   16:04  vmlinuz2.6.32.14
10176.vx6060.fc12.i686.PAE
```

The iperf tool could be used to evaluate and measure the performances. Refer to the following chapters.

7.7 Multinodes Diskless

Multinodes Diskless is a powerfull set of python scripts to help the configuration of diskless root filesystems and to boot distant nodes through the ethernet network (PXE).

The main command is diskless:

```
# diskless --help
```

There are two xml files to configure the diskless system:

`/etc/diskless/images.xml`: describes the layers, the nodes and main configuration setup.

`/etc/diskless/diskless_fs.xml`: list the rpms used to build the root filesystem of the distant nodes;

Have a look at the chapter CPUB Diskless Installation to get a example of diskless usage.

The user manual is available at: `/usr/local/share/doc/diskless/pdf/manual.pdf`



`disklessctl` and `fsbuilder` are not properly working in this version of BSP.

7.8 IGB Module

The Barton Hill (82580) which is the central onboard dual ethernet component connected to the central ethernet switch is not supported by the standard Fedora 12 kernel delivery (2.6.32.xx).

The `igb` rpm package contains the module for this component. It is compiled for the BSP kernel delivery. The source rpm (in SRPMS) may be used to recompile it for another kernel flavour.

After the installation of the BSP, it may be required to force the installation of the `igb` rpm,

Check if the `eth1` and `eth2` are available:

```
# ifconfig -a | egrep "eth1|eth2"
eth1      Link encap:Ethernet  HWaddr 00:00:DE:40:38:00
eth2      Link encap:Ethernet  HWaddr 00:00:DE:40:38:01
```

If there is no output:

```
# rpm -ivh --force /usr/share/vx6060_bsp/bsp_11068/igb-2.1.2-2.i686.rpm
```

And then to add the new `igb` module in the `initrd` image:

```
# dracut --force
```

7.9 Sysvartool

To get the report of the PBIT (Power Built In Test), run the command

```
sysvartool:
# sysvartool -A pbit -l
VX6060 detected
area = 2, arch = 2
POSTs configured to run from command line:
    mem_data: PASSED
    mem_addr: PASSED
    mem_pattern1: PASSED
    mem_pattern2: PASSED
    mem_pattern3: PASSED
    mem_pattern4: PASSED
    ether_loop0: PASSED (FAILED ONCE)
    ether_loop1: PASSED (FAILED ONCE)
    ether_loop2: PASSED
    system: PASSED

PASSED : 10
FAILED : 0
NOT RUN : 0
TOTAL : 10

POSTs configured to run automatically from RAM:

PASSED : 0
FAILED : 0
NOT RUN : 0
TOTAL : 0

POSTs configured to run automatically from ROM:

PASSED : 0
FAILED : 0
NOT RUN : 0
TOTAL : 0
```

7.10 GPIOs

There are 6 GPIOs on the VX6060 (only from CPUA) and on the VX3030. When the GPIO driver is loaded, a collection of special sys files is created for each GPIO:

`/sys/devices/platform/cpld_gpios/gpio/gpioX/direction` should be set to in or out:

```
# echo out > /sys/devices/platform/cpld_gpios/gpio/gpio4/direction
# cat /sys/devices/platform/cpld_gpios/gpio/gpio4/direction
out
```

`/sys/devices/platform/cpld_gpios/gpio/gpioX/value` to get or set (depending on the direction) the value of the GPIO: 1 or 0

```
# echo 0 > /sys/devices/platform/cpld_gpios/gpio/gpio4/value
# cat /sys/devices/platform/cpld_gpios/gpio/gpio3/value
1
```

`/sys/devices/platform/cpld_gpios/gpio/gpioX/value_it` get or set the value. A read to this file is waiting for a interruption on the GPIOs (refer below).

`/sys/devices/platform/cpld_gpios/gpio/gpioX/interrupt` enable or disable interruption on the GPIO: on or off

```
# echo on > /sys/devices/platform/cpld_gpios/gpio/gpio3/interrupt
```

`/sys/devices/platform/cpld_gpios/gpio/gpioX/polarity` to set the polarity of the interruption: hi (high level or rising edge) or lo (low level or falling edge).

`/sys/devices/platform/cpld_gpios/gpio/gpioX/mode` edge or level interruption mode.

```
# cat ./devices/platform/cpld_gpios/gpio/gpio5/mode
edge
```

`/sys/devices/platform/cpld_gpios/gpio/gpio3/toggle` set toggle mode means the interruption is raised when the GPIOs state changes.

```
# echo on > /sys/devices/platform/cpld_gpios/gpio/gpio3/toggle
```

`/sys/devices/platform/cpld_gpios/gpio/gpio3/intr_stat` is a counter of the interruption on that GPIOs.

```
# cat ./devices/platform/cpld_gpios/gpio/gpio5/intr_stat
12
```



The GPIO2 is dual function because it is also a Masquable Reset. This is configurable through CPLD registers:

To set the GPIO2 as a standard GPIO (without issuing a local reset):

```
[root@lnx4 ~]# port 0x870
@0x870 = 0x1d
[root@lnx4 ~]# port 0x870 0xd
@0x870 <- 0x0d
[root@lnx4 ~]# port 0x870
@0x870 = 0x0d
```

7.11 cpldtool

cpldtool may be used to get some high level information about the setting of the CPLD.



NOTE: this should be used with a lot of caution.

```
[root@lnx8 ~]# cpldtool -a
Reg 0x0 - CPLD_ID = 0x0A
    CPLD_ID=0x0
    CPLD_Debug=0x0
    CPLD_Version=0xA
Reg 0x1 - PCB_ID = 0x60
Reg 0x2 - FIRM_PWON = 0x00
Reg 0x3 - PWON_STATUS = 0x00
Reg 0x4 - PWR_RST_CONFIG = 0x23
    PWRON_MODE=0x0
    PLTRST_to_PERST3U_INHIB=0x0
    Alarm_inhib=0x1
    PLTRST_to_PERST6U_INHIB=0x0
    Reset_Mode=0x0
    PERSTb_Control=0x1
    Software_Cross_Reset=0x1
Reg 0x5 - VPD_BC1 = 0x00
Reg 0x6 - VPD_BC2 = 0x00
Reg 0x9 - MEM_PROTECT = 0x00
    Boot_flash_CS_swap_DIP=0x0
    Boot_flash_CS_swap_Valid#=0x0
    Boot_both_flash=0x0
    USER_WP=0x0
    SYS_WP=0x0
    VPD_WP=0x0
    NVMRO=0x0
Reg 0xa - COM_TX = 0x00
    Semaphore=0x0
    Data=0x0
Reg 0xb - COM_RX = 0x00
Reg 0xc - BOARD_CONFIGURATION = 0x08
    Port_D_Mode=0x0
    Port_B_Mode=0x0
    ESW_LAN_CON=0x0
    HANK_LAN_CON=0x0
    USB_SW_SEL=0x1
    SERIAL1_Mode=0x0
    SERIAL2_Mode=0x0
    SERIAL2_cfg=0x0
Reg 0xd - PCI_MODE = 0xEE
    DRAM_DIMM=0x1
    DRAM_RANK=0x1
    DRAM_SIZE=0x2
    PCI_BUSMODE=0x1
    PCI_M66EN=0x1
    PCI_XCAP=0x0
    PCISEL100=0x0
```

```

Reg 0xe - DIP_SWITCH_STATE = 0xF6
  USER_DIP=0xF
  SP2=0x1
  SP1=0x1
  FACTORY_MODE=0x0
Reg 0xf - SERIRQ_CONTROL = 0xE0
  SMB_MSK=0x1
  SMB_TEMP_MSK=0x1
  TIP_MSK=0x1
  GPIOs_INT=0x0
  SMB_INT=0x0
  SMB_TEMP_INT=0x0
  TIP_INT=0x0
  WDG_INT=0x0
Reg 0x5b - ALERT_STATUS = 0x01
  CATERR=0x0
  THERMTRIP=0x0
  THERM_PROT=0x0
  PROCHOT=0x0
  TEMP_ALERT=0x0
  6U_PCIE_ERR=0x0
  3U_PCIE_ERR=0x0
  DDR_Throttling=0x1
Reg 0x6a - GEO_ADD = 0x3E
  SYSCON=0x0
  Error=0x0
  GAP=0x1
  GA=0x1E
Reg 0x70 - VPX_CONTROL = 0x1D
  GDiscrete1_Ct1=0x0
  SMB_Alert_Status=0x0
  MSKR2LOC=0x1
  SW_NT=0x1
  VPX_RST=0x1
  VPX2LOC=0x0
  LOC2VPX=0x1
Reg 0x71 - PCIE_SWITCH_VPX = 0x24
  VPX_REFCLK=0x0
  VPX_AUXCLK=0x0
  ENSW3U=0x1
  FREQ=0x0
  ENSW6U=0x0
  CFG=0x2
  CFC_clock=0x0
Reg 0x72 - I2C_BOARD_STATUS = 0x94
  Power_Status=0x1
  Reset_Source=0x0
  Reset_Status=0x1
  Boot_Status=0x4
Reg 0x73 - I2C_BOARD_CONTROL = 0x60
  Board_Id=0x6
  Check_Errors=0x0
  Reset_3UB=0x0
  Reset_3UA=0x0
  Power_OnOff=0x0

```

7.12 I2C Busses

The `cp1d_i2c` driver is supporting the local i2c bus (I2C bus number 6) which is local to the board and used to address sensors and system eeproms.

There is also two other I2C busses routed to the backplane; the bus number are 7 and 8: `/dev/i2c-7` and `/dev/i2c-8`.

Those busses may be used to access I2C devices on the enclosure or the other boards on the backplane.

7.13 BIOS Update

The `flashrom` package provides with a set of commands and scripts to update the BIOS of the boards.

To update the BIOS on one side (CPUA or CPUB) of the VX6060 or on the VX3030, use the command `ki7updbios`:

Help command:

```
[root@lnx4 sys]# ki7updbios -h
usage:
/usr/sbin/ki7updbios <options> <bios image>
Script to update the BIOS on VX6060 board using flashrom tools.\n
Options:\n
-r Read the current BIOS in bios image file.\n
-w Update the BIOS image using the given bios image file.\n
-a Update the all BIOS eeprom.\n
```

Read the current BIOS:

```
[root@lnx4 ~]# ki7updbios -r BIOS.bin
flashrom v0.9.4 on Linux 2.6.32.14-11035.vx6060.fc12.i686.PAE (i686), built with libpci 3.1.7,
GCC 4.4.2 20091027 (Red Hat 4.4.2-7), little endian
flashrom is free software, get the source code at http://www.flashrom.org

Calibrating delay loop... OK.
No coreboot table found.
Found chipset "Intel QM57", enabling flash write... OK.
This chipset supports the following protocols: FWH,SPI.
Found chip "SST SST25VF032B" (4096 KB, SPI) at physical address 0xffc00000.
Reading flash... done.
[root@lnx4 ~]# ls -ltr BIOS.bin
-rw-r--r-- 1 root root 4194304 2011-03-03 15:54 BIOS.bin
```

Write a new BIOS file:

```
[root@lnx4 ~]# ki7updbios -w VX6060_ID11257.bin
flashrom v0.9.4 on Linux 2.6.32.14-11035.vx6060.fc12.i686.PAE (i686), built with libpci 3.1.7,
GCC 4.4.2 20091027 (Red Hat 4.4.2-7), little endian
flashrom is free software, get the source code at http://www.flashrom.org

Looking for "bios"... found.
Calibrating delay loop... OK.
No coreboot table found.
Found chipset "Intel QM57", enabling flash write... OK.
This chipset supports the following protocols: FWH,SPI.
Found chip "SST SST25VF032B" (4096 KB, SPI) at physical address 0xffc00000.
Flash image seems to be a legacy BIOS. Disabling checks.
Erasing and writing flash chip... Done.
Verifying flash... VERIFIED.
```



Take care of the fact that if the system is boot from the RESCUE BIOS flash (refer to the BIOS user manual), the update of the BIOS using the `ki7updbios` command will update the RESCUE BIOS Flash which is not recommended.



The update of the BIOS through the `ki7updbios` command does not preserve the setup parameters of the BIOS. For deployment of a BIOS version with its own setup parameters, preset one board with the desired parameters, backup this using `ki7updbios -r`, and use this version of BIOS+ setup to be deployed on other similar boards.



In previous BSP, the update BIOS command was `vxupdbios` it has been changed by `ki7updbios`.

7.14 FMRAM Example

The `fmram` packages gives a example of how to access to the FMRAM device which may be used to save some customer data which need to be backedup.

```
# fmram -h
```

Usage `fmram` [options]

a tool to read or write the FerroMagnetic RAM

Options are :

```
-h           : this help
-r <value>  : read at offset <value> (default 0)
-w <value>  : write at offset <value> (default 0)
-s <value>  : data size to read or write (default 4)
-f <filename> : file name used to store (optional) or read (mandatory) data
```

Example:

```
fmram -w 0x10 -s 0x11 -f data_file :
store 17 bytes read from data_file to the ferromagnetic RAM at offset 16
```

With `src` package, the C code of this command is delivered as an example.

7.15 VX6060: Setup Serial Console of CPUB on CPUTA

It is possible to get the serial console of the CPUB available through minicom over an xterm on the graphic display of the CPUTA.

At BIOS level:

- Enable the COM1 console redirection on CPUB
- Disable the COM1 console redirection on CPUTA

Then Boot CPUTA.

- Setup the CPLD to get the console of CPUB routed to CPUTA, by setting bit0 of register 0x80c

```
# port 0x80c
@0x80c = 0x08
# port 0x80c 0x9
```

- Install and setup minicom to get the prompt of the CPUB console: on CPUTA, the CPUB console is /dev/ttyS1 in 115200 bauds rate.



It is possible to have the serial console of CPUB directly available through the front panel console connector. Refer to the Hardware User Guide. It is also possible to get a graphic display on the CPUB through the backplane interface using PIM module.

Chapter 8 - Additional Information

8.1 Known Limitations

» ACPI warnings under Linux OS - CRP 3907

During the boot process, the following messages related to ACPI are displayed (or saved in the boot log):

```
ACPI Warning: 32/64 FACS address mismatch in FADT - two FACS tables! (20090903/tbfadt-369)
ACPI Warning: 32/64X FACS address mismatch in FADT - 7B2C8F40/000000007B2CBD40, using 32
(20090903/tbfadt-486)
...
ACPI Error: No handler for Region [ECF2] (d00174d8) [EmbeddedControl1] (20090903/evregion-319)
ACPI Error: Region EmbeddedControl(3) has no handler (20090903/exfldio-295)
ACPI Error (psparse-0537): Method parse/execution failed [\_SB_.LID0._LID] (Node d00349a8),
AE_NOT_EXIST
```

Those warnings and errors do not impact the system behaviour.

» VX6060 I2C devices are not addressable from CPUA and CPUB at the same time - CRP 3911

It is recommended not to read the board temperature sensors (through the `sensors` command) neither to access the FMRAM (`fmram` command) from the CPUA and from the CPUB in the same time. The read data would be unpredictable.

8.2 Fixed Bug in the Current Release

» TSC (Timestamp Counter) - CRP 3883

The TSC (Timestamp Counter) may drift too much to be a trusted time base for the system.

Fixed in kernel \geq 2.6.32.14-11257

» VXFabric Failure During Setup - CRP 3885

Time to time, VXFabric may fail during setup due to "allocator" initialization failure. Reboot the system to get allocator and VXFabric ready.

Fixed in VXFabric \geq 1.5 and kernel \geq 2.6.32.14-11257 allocator is not used anymore with the previous packages versions.

» Linux system failure during boot

Time to time, the Linux system may fail after a power on at boot time during the udev service startup; reboot the system to recover the system.

MAILING ADDRESS

Kontron Modular Computers S.A.S.
150 rue Marcelin Berthelot - BP 244
ZI TOULON EST
83078 TOULON CEDEX - France

TELEPHONE AND E-MAIL

+33 (0) 4 98 16 34 00
sales@kontron.com
support-kom-sa@kontron.com

For further information about other Kontron products, please visit our Internet web site:
www.kontron.com.