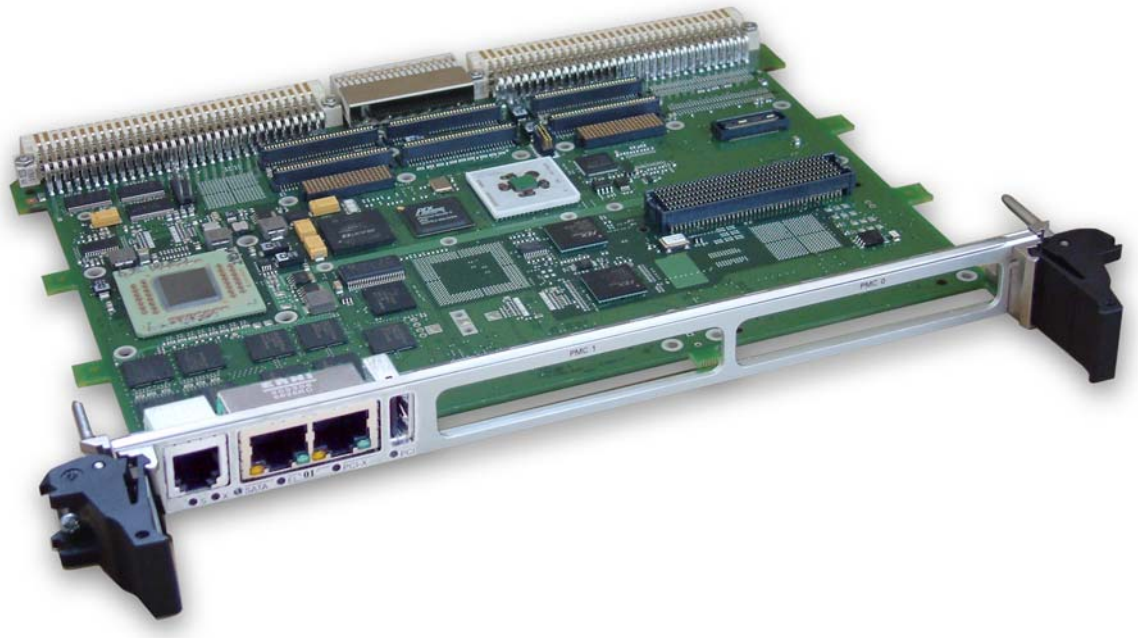
kontron

» VM6250 «



Release Notes Fedora 9
Version 2.0 - ID 09350

SD.DT.F37-3e - January 2015

Revision History

Publication Title:	Fedora 9 on VM6250	
Doc. ID:	SD.DT.F37-3e	
Rev.	Brief Description of Changes	Date of Issue
3e	Section 8.2 - Known Limitations updated	01-2015
2e	Typo on section 7.5 LEDs	12-2010
1e	Release 2.0 ID 09350	01-2010
0e	Initial Version	10-2009

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.

Table Of Contents

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.

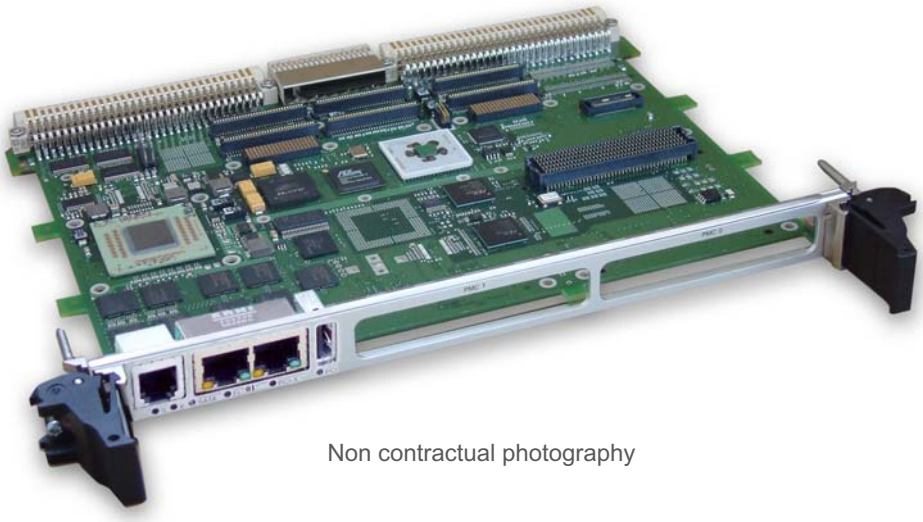
The goal of this document is to help you through the installation process of the Fedora 9 BSP distribution on the Kontron 6U VME VM6250 board.

In this document, the term VM6250 is used both for the VM6250 boards in standard or rugged conduction-cooled version:

- VM6250 Single-slot 6U VME board
  - VM6250-SA Standard Commercial version
  - VM6250-RC Rugged Conduction-Cooled version

In this document, the term VM6250-RTM is associated to the VM6250 Rear Transition Module (RTM):

- VM6250-RTM Rear Transition Module for the single-slot 6U VME board
  - PBV36-P0-PENTXM2-00



Non contractual photography

Figure 1: VM6250 Overview

Chapter 2 - Release Content

The release is made of:

- one DVD-ROM, reference:

  Fedora 9 PPC on VM6250
  Version 2.0 ID 09350

  This DVD-ROM contains a complete Fedora 9 distribution,supporting the VM6250 board (including the BSP).

- one CD-ROM, reference:

  Fedora 9 PPC on VM6250
  BSP 2.0 ID 09350

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

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

- Sensors — CPU Cores and Board temperatures
- Watchdog — Integrated into the onboard PLD
- User GPIOs — Three GPIOs
- Vital Product Data (VPD) Tool — Get board's serial number, order code, E.C. Level, ...
- LEDs — Four Front Panel Tri-color LEDs
- Allocator — Driver and library to reserved contiguous memory area.
- VME Toolkit — Tools and samples to address ALMA VME driver
- U-Boot Environment Variables Tool (sysvartool)
- CPLD register Tool (cpldtool) — Tool to deal with hardware registers of the onboard CPLD
- Upgrade U-Boot (flash_uboot) — Tool to upgrade the U-Boot firmware of the VM6250 from Linux.
- NvSRAM support — Driver and special API file to read/write from/to the NvSRAM

More information on VM6250 BSP in Chapter 7 “BSP Specific Features” page 7.

Information on Fedora 9 is available at http://fedoraproject.org/

Chapter 3 - Associated Documentation

Kontron Documentation

- Hardware
  - VM6250 6U VME SBC User's Guide CA.DT.A65
  - VM6250 Hardware Release Notes CA.DT.A66
- Firmware
  - VM6250 PBIT User's Guide SD.DT.F35
  - VM6250 U-Boot User Manual SD.DT.F36

Fedora 9 Documentation

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

Chapter 4 - Required Configuration

4.1 Hardware Requirements

- A Kontron VM6250 board.
- A Fedora 9 released installed on one of the following disks:
  - SATA disk connected to one of the two SATA connectors available on VM6250-RTM board.
  - Optional onboard USB Flash Disk.
- A DVD-ROM device.

4.2 Firmware Requirements

The version of the U-Boot firmware must be at least:

- **10019**

This version is displayed in the U-Boot Setup.

4.3 Software Requirements

- The DVD-ROM associated to these Release Notes:

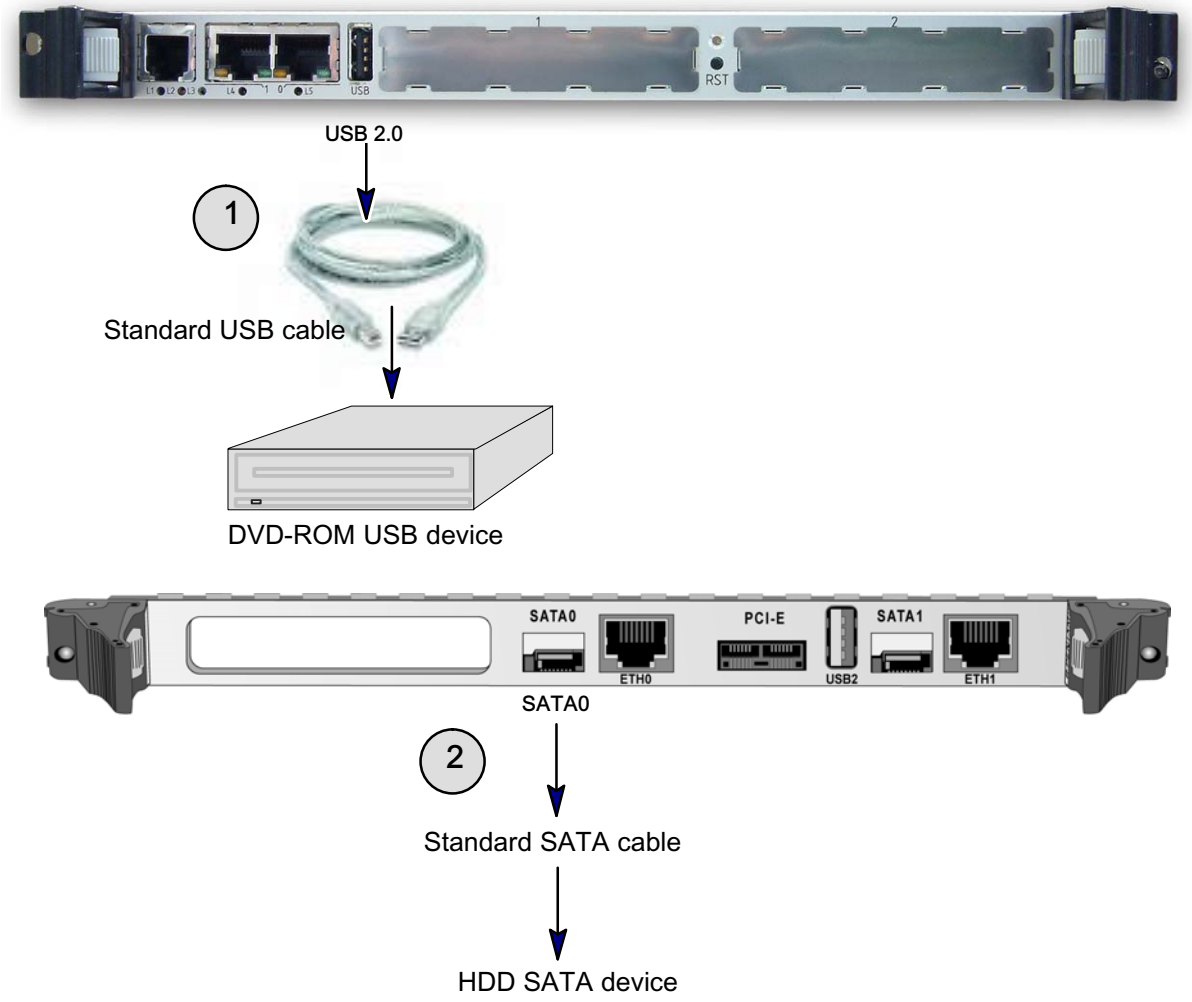
  Fedora 9 PPC on VM6250
  Version 2.0 - ID 09350

- The CD-ROM associated to these Release Notes:

  Fedora 9 PPC on VM6250
  BSP 2.0 - ID 09350

4.4 DVD-ROM Installation Example

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 one of the SATA connectors (**SATA0** or **SATA1**) of the Rear Transition Module (VM6250-RTM).



Chapter 5 - Installation

This chapter describes the specific steps of the installation process of the Fedora 9 on VM6250 boards.

Two installation methods are described hereafter:

- The DVD-ROM Installation. Refer to section 5.1 “DVD-ROM Installation” page 6.
- The CD-ROM Installation. Refer to section 5.2 “CD-ROM Installation” page 9.

  The CD-ROM contains only the BSP specific packages as well as the related source packages. Apply this procedure to install the BSP version 2.0 onto an existing bootable media (SATA disk or USB flash) or to access to the source packages.
- The Network Installation. Refer to section 5.3 “Network Installation” page 10.

For development purpose or to deal with more than one installation, it is recommended to use the Network Installation (faster than the DVD-ROM Installation).

5.1 DVD-ROM Installation

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

There is no major difference between this installation and the standard Fedora 9 installation, so refer to the Fedora documentation to get detail on all the Fedora installation menus.

Nevertheless, the VM6250 is not a graphic board (if no graphic PMC/XMC is fitted), so the installation is done in text mode on the serial port 0.

The other main point is related to the U-Boot firmware and the way to boot on the DVD-ROM and on the SATA drives.

1. After a board reset, or a board power-on, the U-Boot firmware of the VM6250 is run. Configure the console running the following commands:

```
VM6250 =>  setenv bootargs ´console=ttyS0,115200´
VM6250 => saveenv
Saving Environment to EEPROM...
VM6250  =>
```

2. Insert the kontron DVD Fedora 9 in the USB DVD-ROM drive.

3. Start the USB:

```
VM6250 => usb start
(Re)start USB...
USB:   OHCI pci controller (1131, 1561) found @(7:0:0)
OHCI regs address 0x80200000
OHCI pci controller (1131, 1561) found @(7:0:1)
OHCI regs address 0x80201000
OHCI pci controller (1131, 1561) found @(7:1:0)
OHCI regs address 0x80203000
OHCI pci controller (1131, 1561) found @(7:1:1)
OHCI regs address 0x80204000
scanning bus for devices... 6 USB Device(s) found
       scanning bus for storage devices...  Lastsect:00000015
Reading block 16
Reading block 17
 Boot Entry at: 000001A0
 part 0 found @ 1a1 size 1837
2 Storage Device(s) found
VM6250 =>
```

4. Load the Boot Image from the DVD-ROM:

```
VM6250 => usbboot 0x1000000 1:0
 roothub number = 1
  Lastsect:00000015
 Reading block 16
 Reading block 17
  Boot Entry at: 000001A0
  part 0 found @ 1a1 size 1837

 Loading from USB device 1, partition 0: Name: usbdb0
   Type: U-Boot
    Image Name:  VM6250 MultiImage
    Created:      2009-05-18  18:03:53 UTC
    Image Type:   PowerPC Linux Multi-File Image (gzip compressed)
    Data Size:    12695236 Bytes = 12.1 MB
    Load Address: 00000000
    Entry Point:  00000000
    Contents:
       Image 0: 2373914 Bytes =  2.3 MB
       Image 1: 10309016 Bytes =  9.8 MB
       Image 2: 12288 Bytes = 12 kB
  ..........................................................................
  ..........................................................................
  ..........................................................................
  ..........................................................................
 VM6250=>
```

5. Stop the USB:

```
VM6250 => usb stop
stopping USB..
VM6250 =>
```

6. Boot on the loaded image:

```
VM6250 => bootm 0x1000000

Welcome to Fedora for ppc
...
```

7. The installation is now the standard Fedora 9 installation in text mode.
8. If you are installing on the System Flash Mezzanine, it is hardly recommended not to configure swap on this device. This requires to edit the default partition table of the installer.

   The System Flash Mezzanine should be identified as:

```
                sda     3929 MB (Intel Value SSD)
```

9. At reboot time, some configuration of the U-Boot is required to setup the boot parameters for the newly installed system.

   Boot the installed **vmlinuz** and **initrd**, it is required to setup some U-Boot variables.

   The following example is related to a disk on the SATA port 1 installed with the Logical Volume manager partitioning.

```
VM6250 => setenv bootdisk 'ext2load sata 0:1 0x1000000 vmlinuz-2.6.25.ppc.smp;ext2load sata 0:1
0x2000000 initrd-2.6.25.ppc.smp.img;bootm 0x1000000 0x2000000 -'
VM6250 => setenv bootargs 'root=/dev/VolGroup00/LogVol00 ro console=ttyS0,115200'
VM6250 => saveenv
```

10. To boot the system:

```
VM6250=> run bootdisk
```

11. To boot on the USB flash disk, the setup is a bit different due to the access to the USB drive from U-Boot:

    For example:

```
VM6250 => setenv bootargs 'root=/dev/sdb2 ro console=ttyS0,115200'
VM6250 => setenv loadusb 'ext2load usb 0:1 0x1000000 vmlinuz-2.6.25.ppc.smp;ext2load usb 0:1  0x2000000
initrd-2.6.25.ppc.smp.img'
VM6250 => setenv bootusb 'usb start;run loadusb;usb stop;bootm 0x1000000 0x2000000 -'
VM6250 => saveenv
```

    And then boot:

```
VM6250 => run bootusb
```

vmlinuz-2.6.25.ppc.smp and **initrd-2.6.25.ppc.smp.img** are linked to the last installed vmlinuz and initrd respectively. So there is no need to change this variable setup for future updates.

5.2 CD-ROM Installation

The CD-ROM contains only the BSP specific packages as well as the related source packages. Follow the procedure below, to install the BSP version 2.0 onto an existing bootable media (SATA disk or USB flash) or to access to the source packages.

1. Insert the Kontron CD-ROM Fedora 9 in the USB DVD-ROM drive.
2. Boot the VM6250 board on the SATA disk.
3. Login as root and run following commands:

```
[root@vm6250 ~]# mount /dev/sr0  /mnt
[root@vm6250 ~]# cd /mnt
[root@vm6250 ~]# ./install -i
```

   The installation process will take few minutes.

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

5.3 Network Installation

This section describes the installation procedure with a server located on the local network area.

1. The first stage is to place all the Fedora DVD on a server (at least 5 GB are required):

   Insert the Fedora DVD-ROM in the drive.

   ```
   #  mount /dev/sr0 /mnt/cdrom
   #  cp -af /mnt/cdrom/ <target_dir>
   #  umont /mnt/cdrom
   ```

2. The <*target_dir*> of this server should be made accessible to FTP (anonymous or not), HTTP or NFS clients on the network.

3. Configure the BOOTP/TFTP service on the server and copy the network bootable image in the `/tftpboot` directory (or any other proper location):

   ```
   #  cp /mnt/cdrom/ppc/ppc32/vm6250.multi.img /tftpboot
   ```

   The `bootp/tftp` server should be made available on the same sub-network as the target board.

4. On the VM6250 target board, at the U-Boot prompt, run:

   ```
   VM6250 => bootp 0x1000000 vm6250.multi.img
   VM6250 => bootm 0x1000000
   ```

   The **bootp** command may be changed by the **tftp** command.

5. Then, the installation process is the same as the DVD-ROM installation process.

Chapter 6 - Fedora System Configuration

In this chapter, informations related to some specific configuration items of the Fedora 9 system are detailed.

6.1 Network Manager

With Fedora 9, the network interfaces are managed by the NetworkManager service instead of the older network service. The purpose is to make easy to configure several network connections through several types of interfaces and switch from one to another easily.

When editing the configuration of a network interface using system-config-network, make sure to select Controlled by NetworkManager and Activate device when computer starts.

This sets the following lines in the `/etc/sysconfig/network-scripts/ifcfg-ethx` script:

```
ONBOOT=yes
NM_CONTROLLED=yes
```

When the NetworkManager is enabled, the **/etc/resolv.conf** file is automatically edited according to the active network connections. The DNS IP address and domain name are stored in the `/etc/sysconfig/network-scripts/ifcfg-ethx` files at the following entries:

```
DNS1=xxx.xxx.xxx.xxx
DNS2=xxx.xxx.xxx.xxx
SEARCH=mydomain
```

If no such entry is found, no DNS server will be used. DNS2 is optional.

For an embedded system, if may not be suitable to use the NetworkManager service and use the older network service instead.

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

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.

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 VM6250 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 `/etc/udev/rules.d/75-persistent-net-generator.rules` to remove eth* from the device white list.

   This is done by the 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 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.

Chapter 7 - BSP Specific Features

7.1 Sensors

The BSP contains an RPM named `vm6250_sensors` that configures the standard `lm_sensor` software for the VM6250.

To display sensors information:

```
# sensors
lm73-i2c-1-48
Adapter: MPC adapter
 LM73 sensor Temperature: +36.0°C  (low  = -30.0°C, high = +70.0°C)

lm73-i2c-1-49
Adapter: MPC adapter
 LM73 sensor Temperature: +32.0°C  (low  = -30.0°C, high = +70.0°C)

lm73-i2c-1-4a
Adapter: MPC adapter
 LM73 sensor Temperature: +40.0°C  (low  = -30.0°C, high = +70.0°C)

lm95231-i2c-1-2b
Adapter: MPC adapter
Local LM95233 Temperature: +47.0°C
8641D Core Temperature:    +54.0°C
#
```

7.2 Watchdog

The Watchdog service, implemented into the CPLD device, is managed by a driver called **cpld_wdt**.

By default, the **cpld_wdt** driver is configured in interrupt mode.

As a consequence, when a Watchdog timeout occurs, an interrupt is issued which cause an automatic reboot step called: emergency restart.

Watchdog is a daemon that checks if the system is still alive. If programs in user space are not longer executed, it will reboot the system.

To start this service, run the following command:

```
# service watchdog start
#
```

To enable the automatic restart at each reboot, run:

```
# chkconfig --level 3 watchdog on
#
```

7.3 User GPIOs


In following, X=1,2 or 3 and is related to a GPIO number.

> To set the direction of the GPIOs:

As an input:

```
# echo 0 > /sys/class/gpio/gpioX/direction
#
```

As an output:

```
# echo 1 > /sys/class/gpio/gpioX/direction
#
```

By default, the GPIOs are set in input mode.

> To read a GPIO value when it is set as an input signal:

```
#  cat /sys/class/gpio/gpioX/value
```

> To set a GPIO when it is set as an output signal:

```
# echo 1 > /sys/class/gpio/gpioX/value
```

or

```
# echo 0 > /sys/class/gpio/gpioX/value
```

> An interrupt mode is also available. Different conditions can lead to the occurence of an interrupt:

- Rising Edge,
- Falling Edge,
- High Level,
- Low Level.

To specify the interrupt mode you want, enter the following command:

```
# echo "128" > interrupt
```

to enable the Interrupt Falling Edge mode.

```
# echo "129" > interrupt
```

to enable the Interrupt Low level mode.

```
# echo "144" > interrupt
```

to enable the Interrupt Rising Edge mode.

```
# echo "145" > interrupt
```

to enable the Interrupt High Level mode.

Finally, if you want to make the access to the GPIO conditional to the occurence of an interrupt, use the "value_it" interface instead of the "value" interface by entering:

```
# cat value_it
```

In this case, you will remain blocked until the occurence of the interrupt related to this GPIO.

To come back to the normal mode, run:

```
# echo "0" > interrupt
```

7.4 VPD Tool

`vpdtool` command is useful to get information related to the VM6250 board:

```
# vpdtool
Autodetect a VM6250 board.
Loading i2c-cpld module
Board type      : proto-nice-lotA-A
EC Level      : 11000
Serial Number: 0000000000003
Variant       : 0010046004020000
Keylist                                                                                          :
/PCB_A/SACLASS/POPCIEON/SATAON/IOFPGAOFF/IBOMON/ETHFPR/XMC/8641D/MPX4:1/CORE2.5:1/DDR2_400/DDR2_1GB/NOP
E7SERIAL/POUHM/NV128K/1RANK/2ESST/REFSD2_100/REFCPU_25/1SLOT/H8OFF/HDDOFF/IPMIOFF/BC/BATOFF/ITIN/0V95/1
G/
MAC address  : eth0: 0:0:de:40:36:3c, eth1: 0:0:de:40:36:3d, eth2: 0:0:de:40:36:3e, eth3: 0:0:de:40:36:3f
#
```

7.5 LEDs

The driver `leds-cpld` allows to deal with the front panel LEDs in user mode.

The driver `leds-cpld` creates a list of special files and classes in **/sys**:

```
# cd /sys/class/leds/
# ls
led1:amber  led4:amber  led3:green  led2:OFF  led1:red  led4:red
led2:amber  led1:green  led4:green  led3:OFF  led2:red
led3:amber  led2:green  led1:OFF    led4:OFF  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
- low blinking
- fast blinking
- OFF

Example, to set those different modes on the LED 1 in amber:

```
# echo 0 > led1:amber/brightness    The LED 1 is ON in AMBER
# echo 1 > led1:amber/brightness    The LED 1 blinks low in AMBER
# echo 2 > led1:amber/brightness    The LED 1 blinks fast in AMBER
# echo 1 > led1:OFF/brightness       The LED 1 is OFF
```

7.6 Allocator

Allocator is a module allowing the allocation of large contiguous memory areas. Allocator module is may be needed to deal with VME bus. 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.

7.7 VME Bus

ALMA2f is the VME Bridge of the VM6250 boards.

The associated driver is `almavme`. Refer to the manual page of the `VME Toolkit` to get the API of this driver.

The `almavmechan` command (installed through the addons RPM) provides some way to manage master and slave VME channels from user space (through `ioctls` of the ALMA2f driver).

» Example 1: Open a slave VME channel (VME->PCI)

Open a slave VME channel (VME->PCI) on the target decoded at 0x1000000 on the VME for all the AMs matching 0x2FF to access the local memory (8 MB size) at physical address given by `allocator`:

```
[root@sls14 ~]# cat /proc/allocator
Info on Allocator module [0x19000000-0x1cffffff]

Partition 2ESSTIN [ 0x19000000-0x19ffffff ]

Partition BDCSTIN [ 0x1a000000-0x1affffff ]

Partition 2ESSTOUT [ 0x1b000000-0x1bffffff ]

Partition BDCSTOUT [ 0x1c000000-0x1cffffff ]
```

with a `/etc/init.d/allocator` setting: `OPTIONS='linsizemb=64 allocator_part="16M(2ESSTIN):16M(BDCSTIN):16M(2ESSTOUT):16M(BDCSTOUT)"'`

```
[root@sls14 ~]# almavmechan -pcialloc TEST 0x10000000 0x19000000 0x800000 0x2FF013
[root@sls14 ~]# almavmechan -pciprint
-------------------------------------------------------------------------------
VME to PCI CHANNELS
-------------------------------------------------------------------------------
Num    Name     VMEaddr      addr     Space size(MB)     AM    Conv   WP   RH
-------------------------------------------------------------------------------
0     TEST    0x10000000  0x19000000  DRAM   0008      D-02-ff ADDR  Yes  Yes
-------------------------------------------------------------------------------
```

On the VM6250, the VME bus should be locked before issuing a single VME access concurrently with VME DMA accesses. Refer to the option `-b` of `mbm3k` command, and to the `ioctl VMEIOCTL_GET_VME` and `VMEIOCTL_FREE_VME` of the ALMA2f driver. VME DMA transfers are preferred to single VME transfers.

Example 2: Open a master VME channel (PCI->VME)

Open a master VME channel (PCI->VME) to access a VME address at 0x20000000 in A32SData mode:

```
[root@sls14 ~]# almavmechan -vmememalloc TEST2 0x20000000 0x800000 0x0
[root@sls14 ~]# almavmechan -vmememprint
-------------------------------------------------------------------------------
                          PCIMEM to VME CHANNELS
-------------------------------------------------------------------------------
  Name    CPUaddr      PCIaddr      VMEaddr      size(Mb)     AM    Conv    WP    RH
-------------------------------------------------------------------------------
TEST2     0xc0800000 0x00800000  0x20000000     0008      A32SDATA ADDR  No    No
-------------------------------------------------------------------------------
```

Example 3: `mbm3k` command

The `mbm3k` command may be used to exercise some VME DMA transfers. For example to run 100 loops of a DMA read in 2eSST mode at the VME address 0x10000000, VME block size is 128 and the global transfer size is 0x600000:

```
[root@sls14 ~] mbm3k dma2esst r 128 0x10000000 0x600000 100
```

Example 4: Open a slave 2eSST VME channel

Open a slave 2eSST VME channel at 0x10000000 on the VME bus. In that example and if the distant board is a VM6250 running this Linux distribution, the command `almavmechan` may be used to open such a slave 2eSST VME channel:

```
[root@sls14 ~]# almavmechan -pcialloc 2ESST 0x10000000 0x0 0x800000 0x06000003
```

Example 5: Send and receive VME interrupts

Refer to the source examples delivered in `/usr/share/vmetools/src/intr/ directory`.

Enter following commands to generate the `RECV_intr and GEN_intr` programs.

```
#  cd /usr/share/vmetools/src/intr
# make
```

To test the VME interrupts; for example, using two VM6250 boards:

- on one VM6250 board, enter next command to wait for the VME interrupt level 5 and vector 0xa0:

```
# RECV_intr 0x5 0xa0
```

- on the other VM6250 board, enter next command to generate the VME interrupt level 5 and vector 0xa0:

```
# GEN_intr 0x5 0xa0
```

The VME bus must be locked during the acknowledgement of the VME interrupts, this is done in the interrupt handler of the ALMA2f driver. The time to lock/unlock of the VME bus depends on the VME bus load.

7.8 U-Boot Setup and System Environment Variables

The `sysvartool` command may be used to read/edit U-Boot System Environment Variables.

> To list all the U-Boot variables:

```
# sysvartool -l
 Selected area : uboot_area , address = 0x0, size = 0x2000  Transfering 0x2000 bytes from bus /dev/i2c-1
device EEPROM 0x50
   The AREA   uboot_area contains the following variables
  ------------------------------------------------
ramboot = setenv bootargs root=/dev/ram rw console=$consoledev,$baudrate $othbootargs;tftp $ramdiskaddr
$ramdiskfile;tftp $loadaddr $bootfile;tftp $fdtaddr $fdtfile;bootm $loadaddr $ramdiskaddr $fdtaddr nfsboot
=          setenv          bootargs          root=/dev/nfs          rw          nfsroot=$nfsip:$rootpath
ip=$ipaddr:$serverip:$gatewayip:$netmask:$hostname:$netdev:off
console=$consoledev,$baudrate  $othbootargs;tftp $loadaddr $bootfile;tftp $fdtaddr $fdtfile;bootm $loadaddr
- $fdtaddr baudrate = 115200 loads_echo = 1 rootpath = /opt/nfsroot hostname = unknown bootfile = uImage
...
```

> To set a variable. In following example, to change the linux bootargs variable:

```
# sysvartool -a bootargs="console=ttyS0,115200 root=/dev/sda2 ro"
```

> To delete a variable. In following example, a variable is created and deleted:

- Create a variable:

```
# sysvartool -a foo=fee
```

- Check, the variable is saved:

```
# sysvartool -l | grep foo
foo = fee
```

- Delete the variable:

```
# sysvartool -d foo
 Selected area : uboot_area , address = 0x0, size = 0x2000
 Transfering 0x2000 bytes from bus /dev/i2c-1  device EEPROM 0x50
 The variable foo has been deleted successfully
 cp = 0x10015b5a, size written = 0xb42, size remaining 0x14be
 Transfering 0x2000 bytes to  bus /dev/i2c-1  device EEPROM 0x50
 Check it is done
# sysvartool -l | grep foo
```

> To retrieve the PBIT (Power-On Built In Test) results if it had been run during the U-Boot startup:

```
# sysvartool -A pbit -l
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
    mem_bitflip: PASSED
    mem_addrpat: PASSED
   mem_addrpat2: PASSED
           cpu0: PASSED
           cpu1: PASSED
     pcieswitch: PASSED
pciepci64switch: PASSED
pciepci32switch: PASSED
         serial: PASSED
        battery: PASSED
            rtc: PASSED
       sysflash: FAILED
           cpld: PASSED
   temp_sensors: PASSED
    temperature: PASSED
         nvsram: PASSED
    ether_loop0: PASSED
    ether_loop1: PASSED
    ether_loop2: PASSED
    ether_loop3: PASSED
     sata0_test: PASSED
     sata1_test: PASSED
     eeprom_vpd: FAILED
       watchdog: PASSED
            vme: PASSED
  pmcAxmc_check: PASSED
  pmcBxmc_check: PASSED
usb0_controller: PASSED
usb1_controller: FAILED

PASSED  : 32
FAILED  : 3
NOT RUN : 0
TOTAL   : 35

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
```

7.9 CPLD Register Tool

The onboard CPLD manages different functions of the board (resets, watchdog,...). The `cpldtool` gives a ligth interface to access the registers of the component.

Some registers of the CPLD have a direct impact on the hardware configuration of the VM6250. So, handle those registers with care. Note that some registers and some bits are read only.

To load the module:

```
# modprobe lbc-cpld

# cpldtool
Usage: cpldtool no arguments displays help
       cpldtool -a displays all registers
       cpldtool -d <register address> displays one register
       cpldtool -i <register address> displays info on one register
       cpldtool -s <register address> <hexval> sets one register to hexval
       cpldtool -f <register address> <field> <hexval> ... <field> hexval sets register field by field.
       option: -v : gets some more verbosity

# cpldtool -a

Reg 0x2 - Memory Configuration = 0x74
MEM_SP[1:0]=0x3, ECC=0x1, MEM_SZ=0x2, MEM_BK=0x0

Reg 0x4 - I2C Control = 0x40
Strobe/Busy=0x0, SetAck/GetAck=0x1, Mode=0x0

Reg 0x5 - I2C Data = 0x00

Reg 0x8 - Interface Configuration = 0xFF
SDT[2:0]=0x7, SC=0x1

Reg 0x80 - Firmware POST Code Low = 0x00

Reg 0x81 - Firmware POST Code High = 0x00

Reg 0x84 - Debug POST Code Low = 0x00

Reg 0x85 - Debug POST Code High = 0x00
...
...
...

Reg 0x297 - Geographical Addressing register = 0x9C
SYSCON=0x1, Res.=0x0, GAP=0x0, GA=0x1C

#
```

For example, to set the ETH1 port on the front panel:

```
#  cpldtool -d 0x287

Reg 0x287 - Host I/O Configuration = 0x07
WDG_LOCK=0x0, USB1_DIR=0x0, USB0_DIR=0x1, ETH1_LS=0x1, ETH0_LS=0x1
#  cpldtool -s 0x287 0x05
#  cpldtool -d 0x287

Reg 0x287 - Host I/O Configuration = 0x05
WDG_LOCK=0x0, USB1_DIR=0x0, USB0_DIR=0x1, ETH1_LS=0x0, ETH0_LS=0x1
#
```

7.10 Upgrade U-Boot

The **flash_uboot** tool may be used to upgrade and check the U-Boot of the VM6250

To read the U-Boot binary image in a file:

```
# flash_uboot -r -f /tmp/uboot.img
Flash Operation on VM6250 U-Boot 1.3.3
ID09201 for KONTRON VM6250 board
ID09201
UBoot size 0x80000
CRC stored 00001eb5
CRC calculated is the same
#
```

To write an U-Boot image:

```
# flash_uboot -w -f /tmp/uboot.img
Flash Operation on VM6250
Writing Uboot ID 09201 in flash
#
```

To check the U-Boot flash:

```
# flash_uboot -c
Flash Operation on VM6250
Information in flash
U-Boot 1.3.3 ID09201 for KONTRON VM6250 board Version ID09201
#
```

7.11 NvSRAM Option

The NvSRAM is a customizable option. So the NvSRAM is not always available onboard.

If available, there is a driver to read/write into this device:

The character special file `/dev/nvsram` may be used to address the NvSRAM.

For example:

```
# dd if=/etc/passwd of=/dev/nvsram
3+1 records in
3+1 records out
1763 bytes (1.8 kB) copied, 1.97963 s, 0.9 kB/s
# cat /dev/nvsram
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
```

Chapter 8 - Additional Information

8.1 Main Upgrades

8.1.1 Main Upgrades between Release 09138 and 09350

Fixed CRPs

- CRP3763 VM6250 Several modules not downloaded by default.
- CRP3768 VM6250 Module **sunrpc** not found during boot step.
- CRP3764 VM6250 Network configuration script does not work.
- CRP3765 VM6250 BSP packages not installed automatically.
- CRP3767 VM6250 ALMA2f VME bridge not correctly identified by the **lspci** command.
- CRP3769 VM6250 Bad interrupts caused by ethernet activity.
- CRP3772 VM6250 CPLD Watchdog interrupt mode not supported.
- CRP3773 VM6250 PCIExpressToPCI bridges not configured in an optimal way.
- CRP3775 VM6250 Make easier the boot process by using generic name for linux images and initrd.
- CRP3776 VM6250 Adding **cpldtool** and **flash_uboot** utilities.
- CRP3770 VM6250 CPLD GPIOs interrupts implementation does not conform to the specification.

8.2 Known Limitations

Issue raised at boot time when the second PCI bus is registered

```
 io scheduler cfq registered (default)
 proc_dir_entry ‘00’ already registered
 Call Trace:
 [ef82be80] [c0008cf4] show_stack+0x4c/0x180 (unreliable)  [ef82bed0] [c00ed4f8]
proc_register+0x174/0x1c4  [ef82bf00] [c00ed6ec] proc_mkdir_mode+0x44/0x70  [ef82bf20] [c01b9408]
pci_proc_attach_device+0x94/0x128  [ef82bf50] [c04d5bc0] pci_proc_init+0x74/0xa0  [ef82bf60] [c04bd3a0]
kernel_init+0x24c/0x3f4  [ef82bff0] [c0010e4c] kernel_thread+0x44/0x60  assign_interrupt_mode Found MSI
capability
```

This is due to the fact that the 2nd PCI bus start the bus numbering at 0 like the first one.

This may produce some trouble for application still using `/proc/bus/pci` to deal with pci busses instead of `/sys filesystem` (new implementation till 2.6 kernels).

Five seconds delay at boot time with some SATA hard drives

At boot time, with some SATA hard drives, the boot process may stop during SATA discovery displaying following message:

```
sata_sil24 0003:0b:00.0: version 1.1
scsi0 : sata_sil24
scsi1 : sata_sil24
ata1: SATA max UDMA/100 host m128@0xb0000000 port 0xb0004000 irq 72
ata2: SATA max UDMA/100 host m128@0xb0000000 port 0xb0006000 irq 72
ata1: SATA link up 1.5 Gbps (SStatus 113 SControl 10)
ata1.00: failed to IDENTIFY (INIT_DEV_PARAMS failed, err_mask=0x80)
ata1: failed to recover some devices, retrying in 5 secs
ata1: SATA link up 1.5 Gbps (SStatus 113 SControl 10)
ata1.00: ATA-6: ST3120026AS, 3.18, max UDMA/133
```

Boot Fedora 9 on some FDM-USB may failed

Some USB devices (and FDM-USB) may take too much time to be initialized so the init script fails to mount the root file system.

On that kind of devices the boot will fail with this kind of ouput:

```
switchroot: mount failed: No such file or directory
Booting has failed.
scsi 2:0:0:0: Direct-Access     SMART    eUSB             893A PQ: 0 ANSI: 0 CCS
sd 2:0:0:0: [sda] 7864320 512-byte hardware sectors (4027 MB)
sd 2:0:0:0: [sda] Write Protect is off
sd 2:0:0:0: [sda] Assuming drive cache: write through
sd 2:0:0:0: [sda] 7864320 512-byte hardware sectors (4027 MB)
sd 2:0:0:0: [sda] Write Protect is off
sd 2:0:0:0: [sda] Assuming drive cache: write through
sda: sda1 sda2
sd 2:0:0:0: [sda] Attached SCSI disk
```

To work around the issue, it is required to boot in rescue mode on the Fedora 9 installation DVD.

Restart the VM6250 board with the DVD device and Installation DVD-ROM (ID=09350) and boot on it following the steps:

Set the **bootargs** variables as follow:

```
VM6250 => setenv bootargs ´console=ttyS0,115200 rescue´
VM6250 => saveenv
```

Boot on the DVD:

```
VM6250 => usb start
VM6250 => usbboot 0x1000000 1:0
VM6250 => usb stop
VM6250 => bootm 0x1000000
```

Go through the menus (Language, …) and select the **Continue** option to get the prompt at the rescue console.

Then:

```
# chroot /mnt/sysimage
```

Edit the script **mkinitrd**:

```
# vi /sbin/mkinitrd
```

Add just after the line:

```
emit "stabilized /proc/bus/usb/devices"
```

the line:

```
emit "sleep 5"
```

Save the file and rebuild the **initrd** image:

```
# mkinitrd /boot/initrd_fix.img 2.6.25-09350.vm6250.fc9.ppc.smp
resolveDevice: device spec expected
# /usr/bin/mkimage -n ´uboot ext2 ramdisk rootfs´ -A ppc -O linux -T ramdisk -C gzip -d /boot/initrd_fix.img
/boot/initrd_fix.img.uboot
Image Name:   uboot ext2 ramdisk rootfs
Created:      Wed Dec 31 19:18:31 1969
Image Type:   PowerPC Linux RAMDisk Image (gzip compressed)
Data Size:    3746371 Bytes = 3658.57 kB = 3.57 MB
Load Address: 0x00000000
Entry Point:  0x00000000
```

Set the **symlink** to the new **initrd**:

```
# cd /boot
# rm initrd-2.6.25.ppc.smp.img ; ln -s initrd_fix.img.uboot initrd-2.6.25.ppc.smp.img
```

Then restart the board with the **exit** command (2 times):

```
# exit
# exit
```

Set back the **bootargs** to the standard boot value:

```
VM6250 => setenv bootargs ´root=/dev/VolGroup00/LogVol00 ro console=ttyS0,115200´
VM6250 => saveenv
```

Remove the DVD and reboot on the FDM-USB device:

```
VM6250 => run bootusb
```

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: Order-ATD-Toulon@Kontron.com
Support: GSS-ATD-Toulon@Kontron.com

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