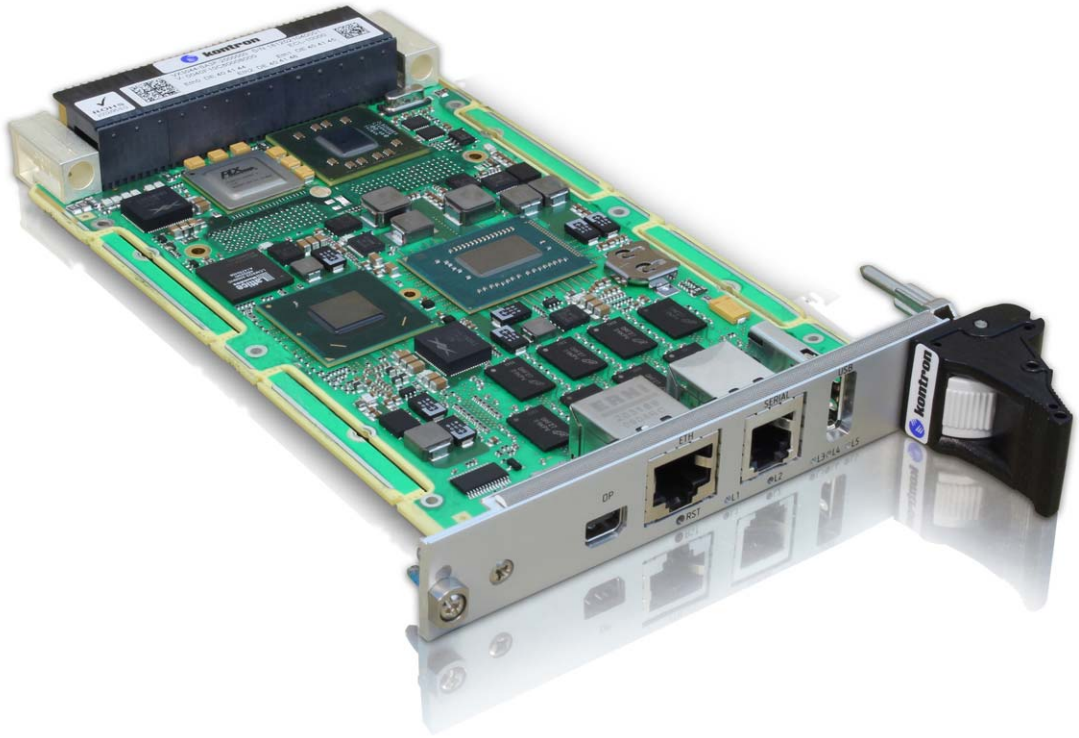
kontron

» VX3042 & VX3044 «



AMI BIOS User Reference Manual

SD.DT.F96-4e - December 2013

Revision History

Publication Title:	VX304x AMI BIOS User Manual	
Doc. ID:	SD.DT.F96-4e	
Rev.	Brief Description of Changes	Date of Issue
4e	New Release ID13346	12-2013
3e	New Release ID13287	10-2013
2e	New Release ID13205 and Update of: - Section 5.1 - CPU Configuration - New sections 6.1 & 6.2 - Section 10.1.28 - New Section 11.5	09-2013
1e	New Release ID13148 and Update of: - Chapter 2 - Accessing the Setup Menu - Chapter 3 - Main Menu - Section 4.2 - SATA Configuration - Section 4.5 - CPU PPM Configuration - Section 10.1.28: kvpx - Chapter 11 - BIOS Versions Description	06-2013
0e	Initial Version	01-2013

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.

Table Of Contents

Chapter 1 - Overview

This manual introduces the SETUP, EFI-SHELL of the AMI BIOS firmware available on Kontron VX304x boards.

The BIOS SETUP is a ROM-based configuration utility that displays the system's configuration status and provides users with a tool to set their system parameters. These parameters are stored in the non-volatile System Flash which saves this information even when the power is turned off. When the system is turned on, the system is configured with the last saved values. Using easy-to-use pull down menus, users can configure such items as:

- Date & Time
- Serial Port, Terminal Type, Console redirection
- USB keyboard layout
- Watchdog for OS boot
- LAN routing, VPX configuration
- CPU active cores
- Boot method and boot device priority
- Security password

This manual applies to the release ID13346 of the AMI BIOS *

** Enter SETUP/MAIN menu to get BIOS ID*

1.1 Structure

1.2 Related Documents

VX304x Hardware

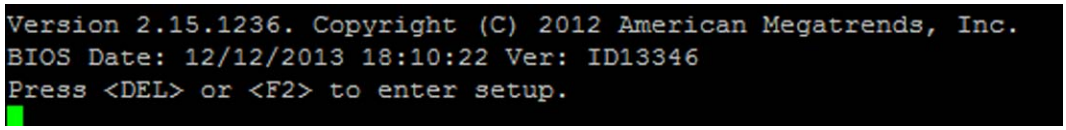
- VX304x Hardware Release Notes CA.DT.A99
- VX304x User's Guide CA.DT.A98

VX304x Software

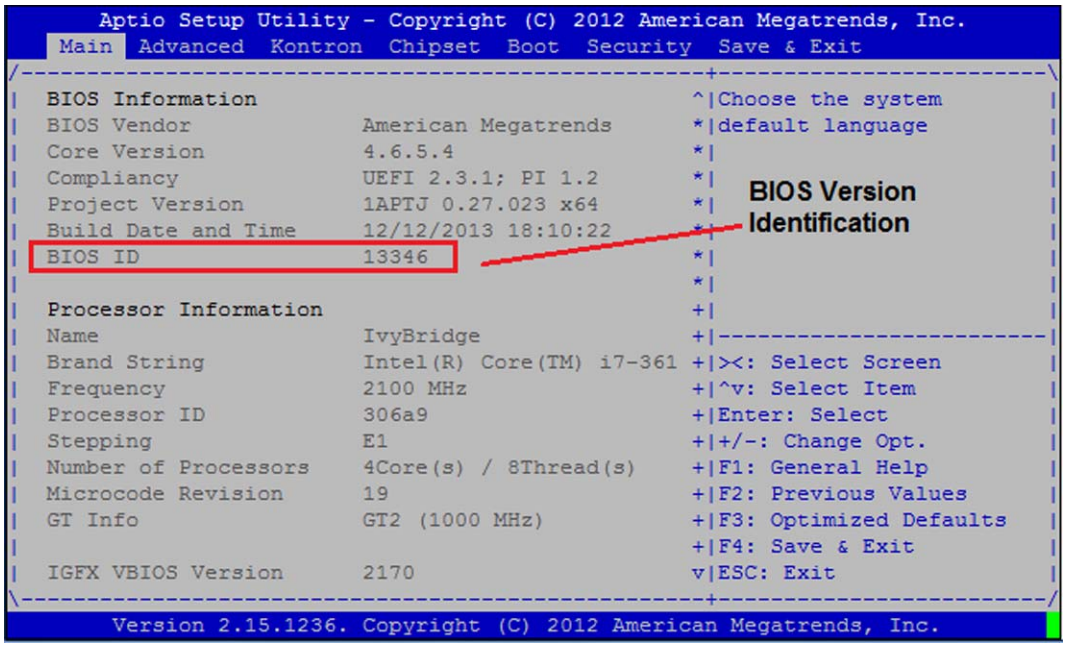
- VX304x - Release Notes for BSP Fedora 16 . . . . . SD.DT.G11

Chapter 2 - Accessing the SETUP Menu

To access the SETUP MENU, press <F2> during system boot when the message below is displayed :



A screen similar to the one shown below will appear:



The SETUP displays the system's current configuration settings. The top of the screen has a menu bar with various items (i.e., Main, Advanced, Kontron, etc.). The menu bar items are linked to submenus. Any submenu includes various items to configure the system or to perform specified tasks. For example, the Main menu contains a list of items such as setting the date and time or displaying the AMI BIOS version and ID ...

To get the SETUP menu from COM0 serial line, configure your terminal to 115200 baud. COM0 is available either via the front panel or via the backplane connector of the VX304x board.

The following chapter details the items that are available on Kontron VX304x. Some of them are for future implementation, so are marked as reserved and should not be used.

The following chapters provide a sampling of menu items:

2.1 Working with First Level Menu Items

To access the menu of your choice:

> Use the < → > or < ← > keys to select the desired item Menu
> Use the < ↑ > or < ↓ > keys to highlight the desired setting or submenu in item
> Press < Enter > key to validate your choice.

Depending on the menu item selected, one of the following occurs:

> A pop-up window prompts users to enable/disable the selected item.
> A window appears with a list of options to choose from.
> A window appears prompting the user to supply input.
> Links to the submenu.

While the menu item is highlighted, its corresponding Help text is also displayed to help explain the purpose of the item.

> Use < ESC > to get out of the current menu item and jump to its parent item

2.2 Boot Manager Menu

To access the Boot Manager menu, press < F7 > during system boot up. The Boot Manager menu is used to select the boot device.

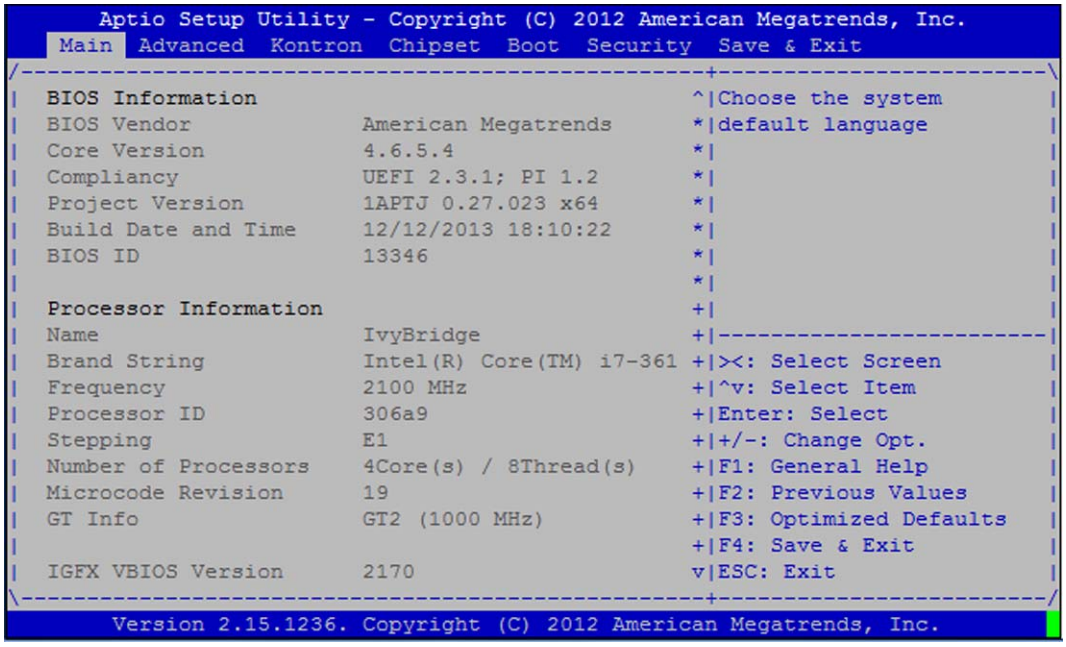
> Select a device from the list (Use the <↑> or <↓> to highlight the desired item)
> Press < ENTER > to boot the selected device or enter setup

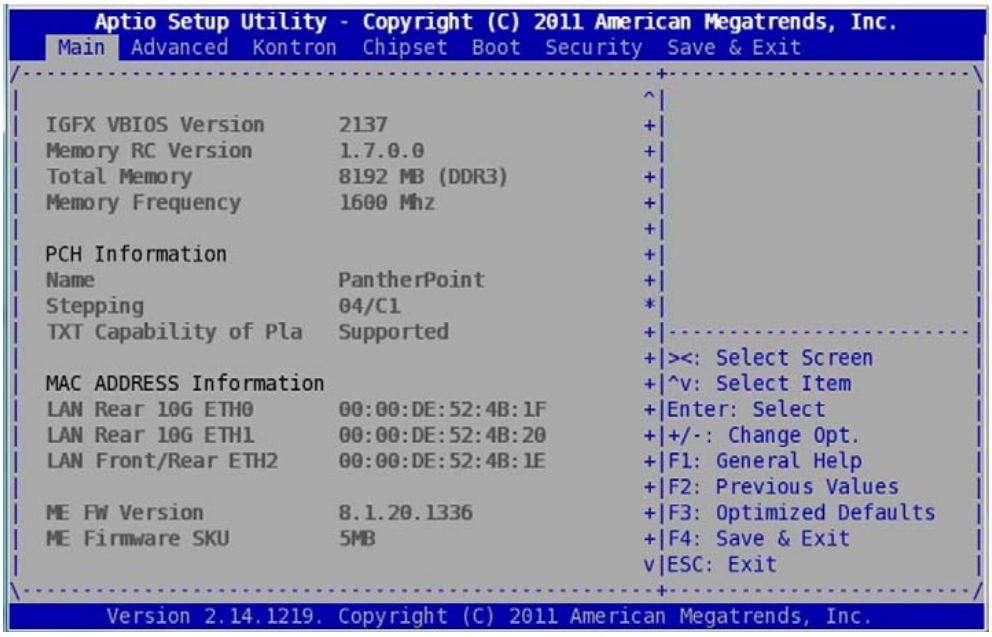
Chapter 3 - Main Menu

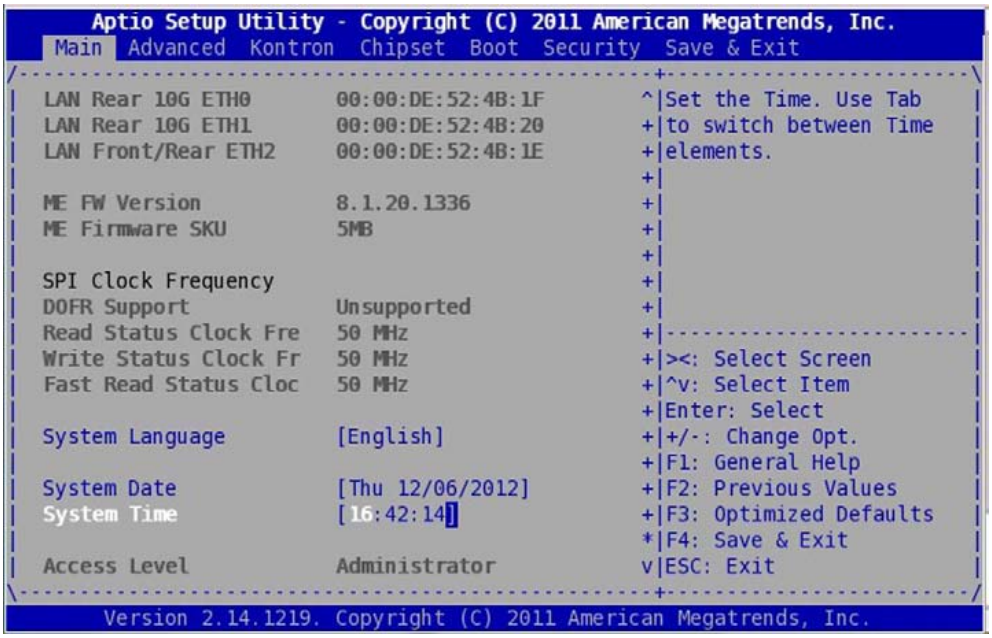
The Main Menu provides general system information and is the first accessible menu page.

Six sections are accessible from the main menu:

- BIOS Information
- Processor Information
- PCH Information
- MAC ADDRESS Information
- SPI Clock Frequency
- System Language
- System Date Time







The "Information" section displays:

- The BIOS ID and build date
- The board identity
- The processor name, frequency, stepping, number of cores and threads, graphic information, total memory size and frequency
- The PCH (Platform Controller Hub) name, stepping
- The MAC addresses of the 3 Ethernet interfaces

The entire display is accessible by scrolling down using the arrow key <↓>.

Only English is supported as System Language in this version.

The System Date and System Time fields allow the user to specify the month/day/year as well as the hour/minute/second of the system.

Time is represented in a 24-hour format.

To update the System Date, use the **<+>** or **<->** keys to select the Month (**<+>** to increase / **<->** to decrease the number of the month), and press the **<Enter>** key to validate your choice. Proceed in the same way for the day and finally for the year.

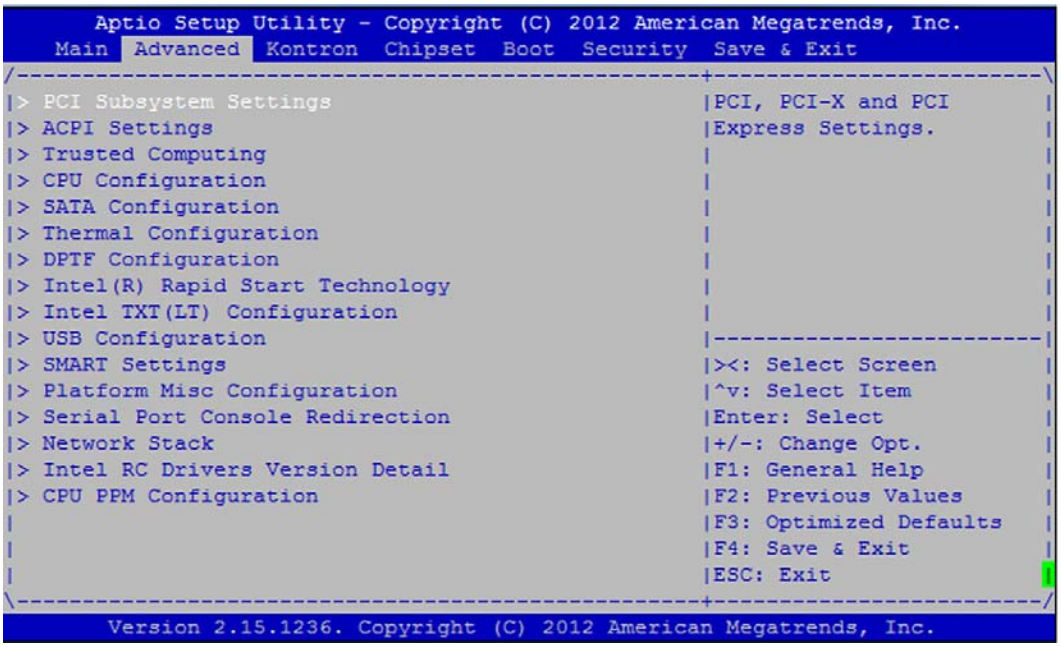
To update the Time, use the **<+>** or **<->** keys to select the Hour (**<+>** to increase / **<->** to decrease the hour), and press the **<Enter>** key to validate your choice. Proceed in the same way for the minutes and finally for the seconds.

The firmware always reads a RTC to display the date and time at each power-on. To keep the current date and time, the RTC needs to be supplied with the external battery otherwise System Date and System Time are initialized with the build date of the BIOS.

The VX304x board can operate safely without any battery fitted. In this case, the non-volatile board settings are managed this way:

- All the BIOS user settings are kept forever (in a specific area of the BIOS Flash)
- The Date/Time is lost at each Power-Down, and without battery fitted, the BIOS displays the BIOS build Date/Time instead of the current Date/Time.

Chapter 4 - Advanced Menu



The Advanced menu provides system-level controls to configure the device settings in the following submenus:

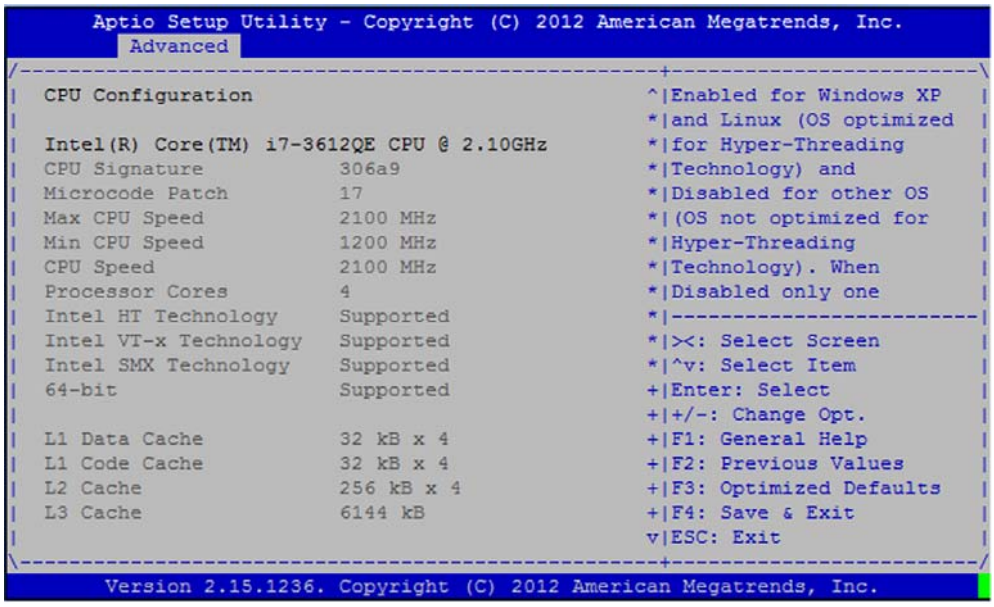
- `CPU Configuration` (hyper-threading, active cores) - Section 4.1 page 8
- `SATA Configuration` (mode selection, speed, port management) - Section 4.2 page 12
- `USB Configuration` (Legacy support) - Section 4.3 page 14
- `Serial Port Console Redirection` - Section 4.4 page 16
- `CPU PPM Configuration` (Turbo mode) - Section 4.5 page 19

The other submenus are Not intended to be changed.

4.1 CPU Configuration

This menu displays information about the CPU speed capabilities and speed setting.

» On a VX3044 board:



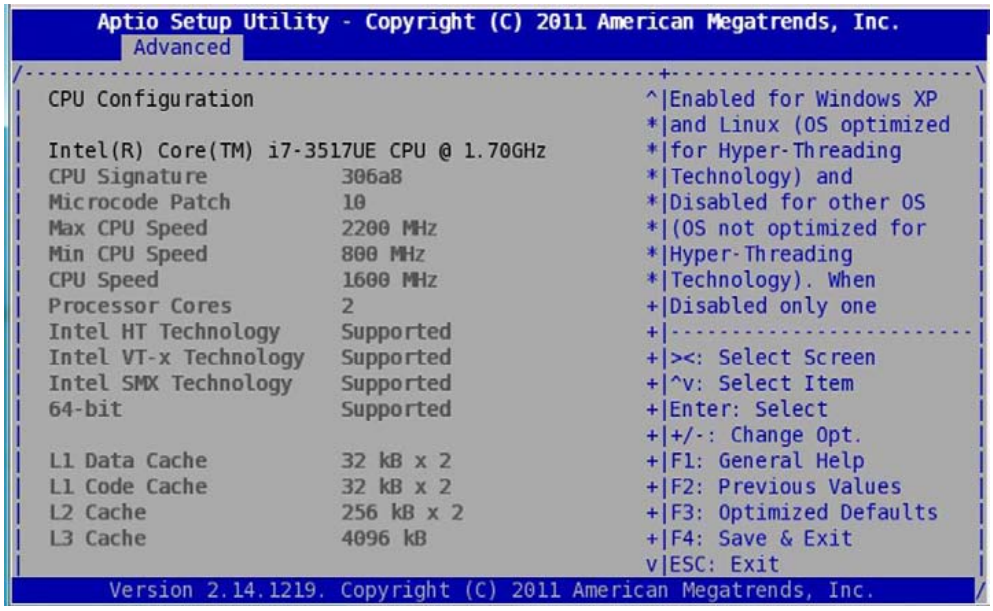
On a VX3044 board, the Thermal Design Power (TDP) is not configurable.

By default, the CPU speed corresponds to the frequency suitable for the maximum processor power 35W.

To force the CPU to its minimum power 30W/1.2 GHz, the microswitches SW3[1-2] must be set to ON.

Refer to the VX3042 and VX3044 - User's Guide - CA.DT.A98, section "Microswitch SW3 Description".

On a VX3042 board:



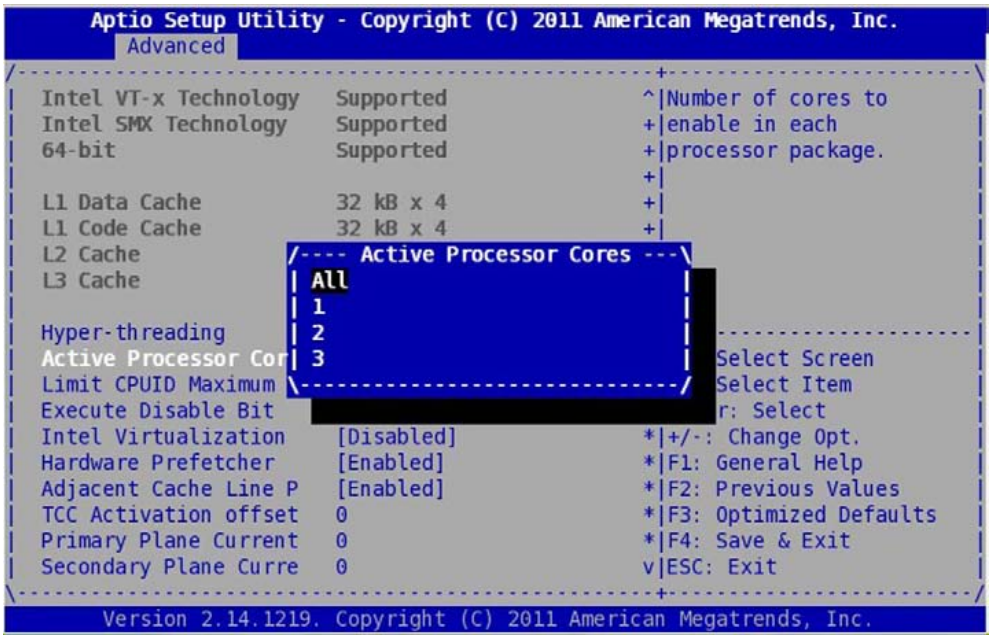
On a VX3042 board, the Thermal Design Power (TDP) is configurable by setup.

By default, the CPU speed corresponds to the frequency suitable for the nominal TDP 17W.

Also, this menu allows the user to configure the number of active cores and to enable/disable the Hyper-Threading feature.

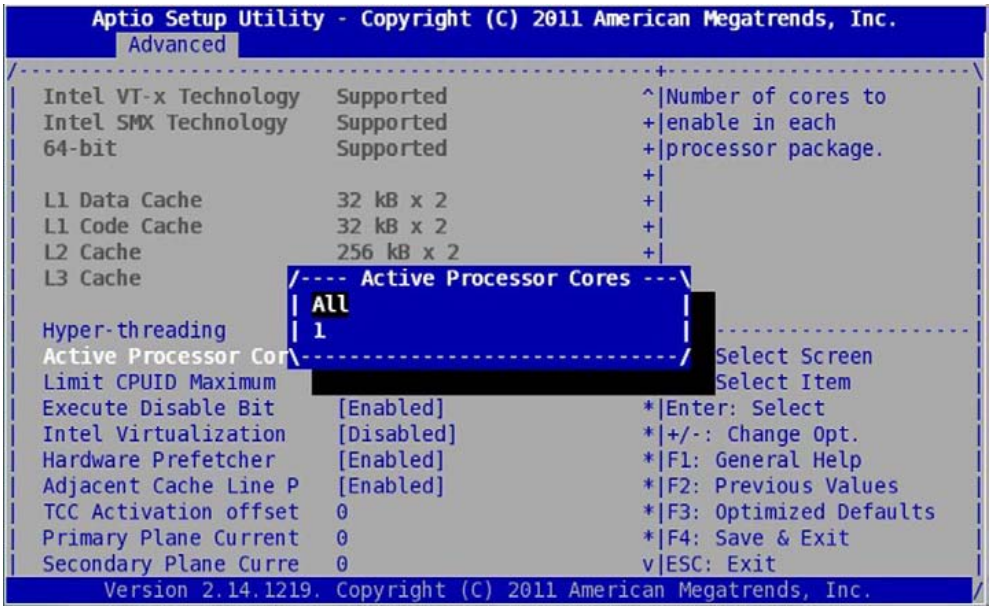
4.1.1 Active Processor Cores

» On a VX3044 board:



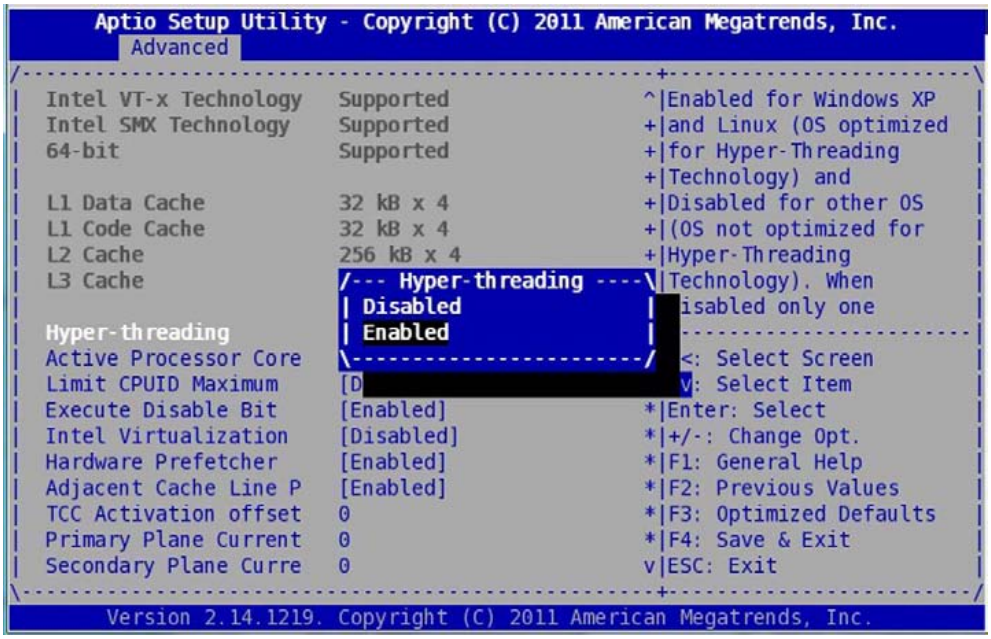
On a VX3044 board, up to 4 cores can be activated.

» On a VX3042 board:



On a VX3042 board, up to 2 cores can be activated.

4.1.2 Hyper-Threading

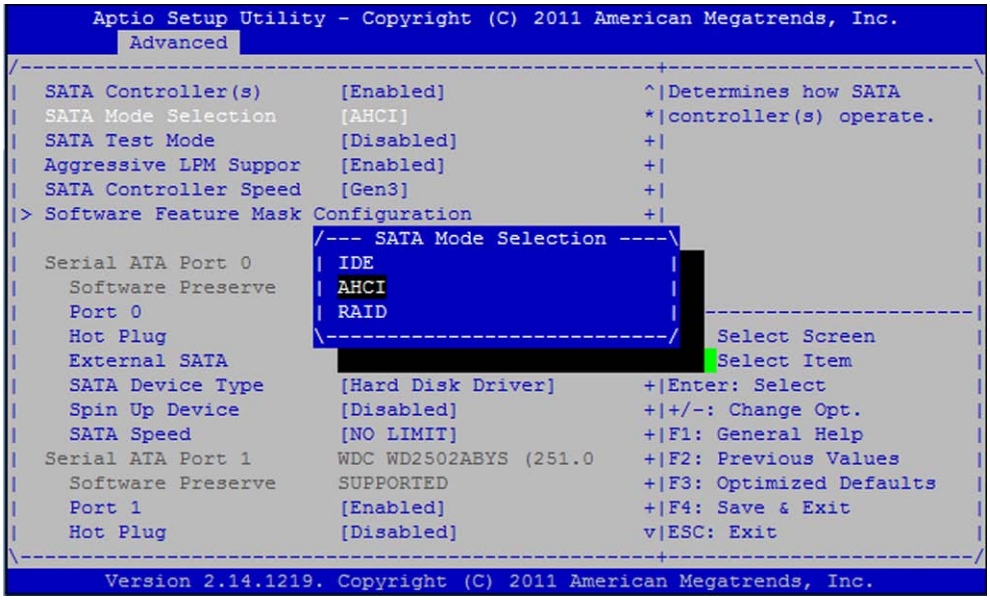


When `Hyper-Threading` is `Enabled`, 2 logical CPUs per core are present so there are up to 4 logical CPUs on a VX3042 board and up to 8 logical CPUs on a VX3044 board.

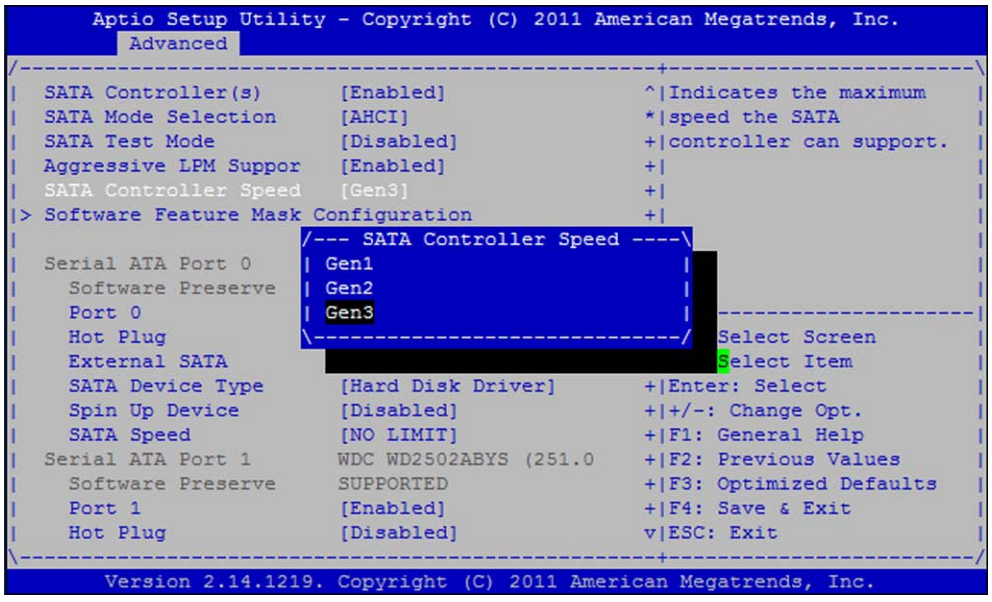
4.2 SATA Configuration

This menu can be used to :

- Select the SATA mode (AHCI or IDE)



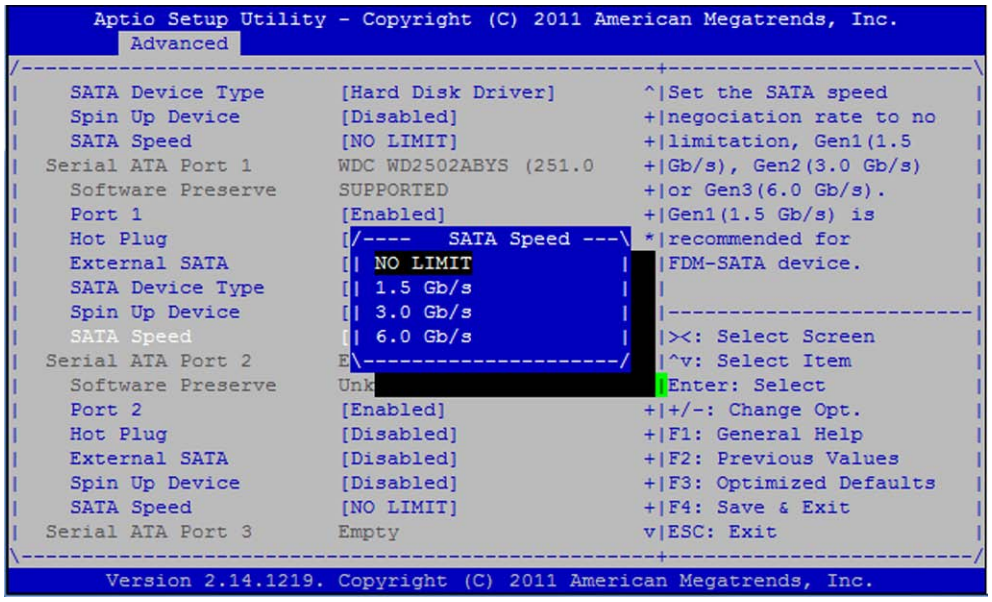
- Select the maximum speed supported by the SATA controller.



The SATA controller speed selection impacts all the SATA ports.

SATA RAID mode is supported by the BIOS but has not been tested.

- Select the maximum speed for the SATA Port:



By default, the SATA Speed for each Port is not limited (`NO LIMIT`).

However, the SATA Speed Port #4 for onboard FDM-SATA is set to GEN1 (`1.5 GT/s`) by default in setup due to the limitation of the device.

1. In AHCI mode, the SATA controller speed takes precedence over the SATA speed by port.
2. In IDE Mode, only the SATA speed by port can be set.
3. In AHCI mode, usually, the operating system renegociates the SATA speed based on the capabilities registers. It is possible to force the SATA speed using the `libata.force` option at the kernel command line to boot Linux OS.

4.3 USB Configuration

This menu can be used to:

- Enable/disable the Legacy USB Support (such as DOS legacy environment). This can be used to avoid booting on a USB device when a USB device is connected.
- Enable/disable the USB 3.0 Support. If enabled, the corresponding USB port is accessible at the rear of the VX304x board.

The other options should Not be changed.

4.3.1 Legacy USB Support

Select menu `Legacy USB Support` to change it. There are three options to choose from:

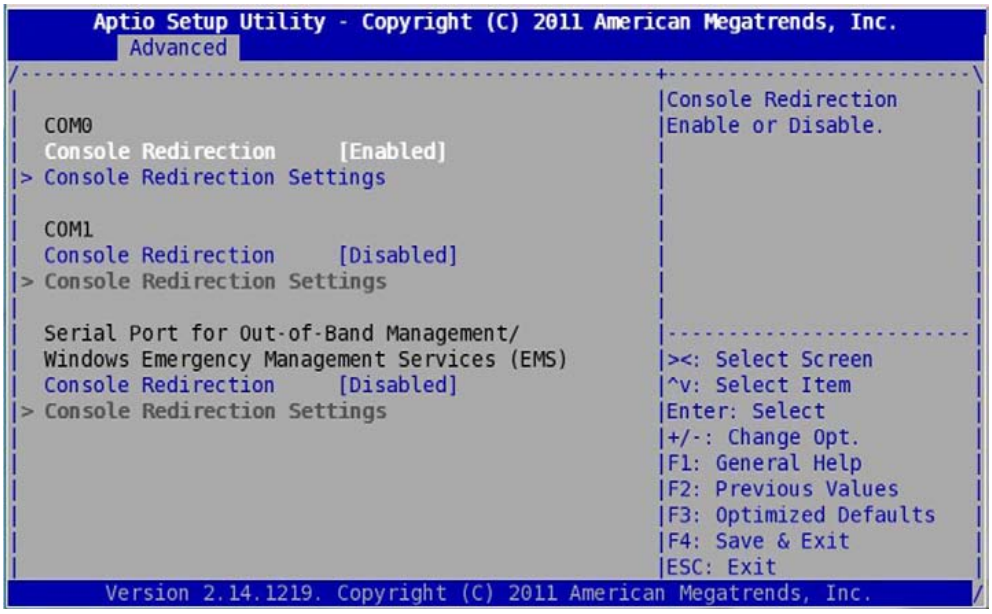
- `Enabled`
- `Disabled`
- `Auto`

`AUTO` option will disable the Legacy Support if no USB device is connected.

`Disabled` option will keep the USB device available for EFI application.

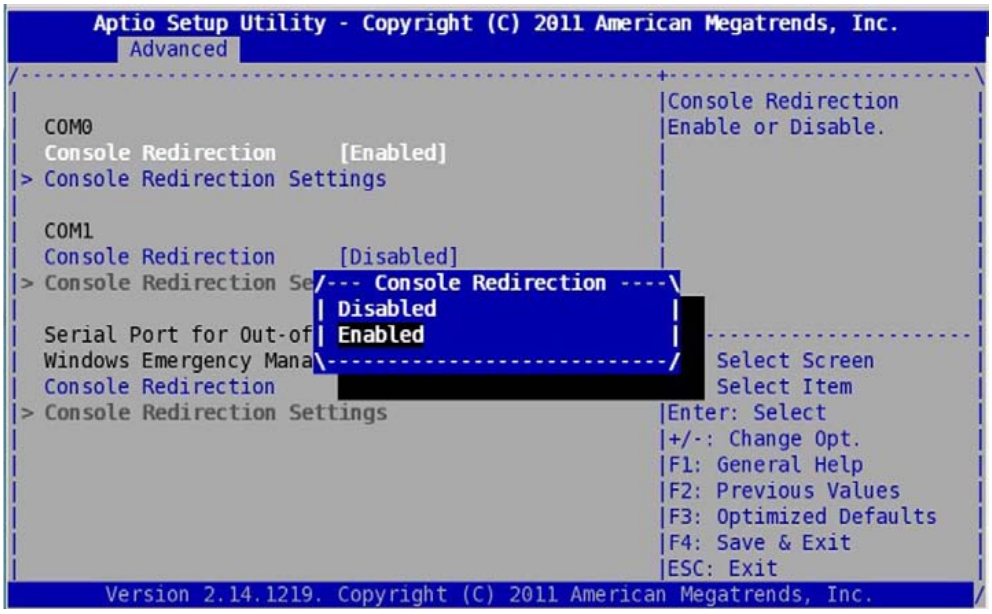
4.4 Serial Port Console Redirection

The BIOS console can be redirected to the serial COM0 and/or the serial COM1 with the Console Redirection menus. Also the characteristics of the COM0 or COM1 serial line can be modified with the Console Redirection Settings menus as described after:



4.4.1 COM0/COM1 Console Redirection

The user has the option to enable/disable the serial Console Redirection on COM0 or on COM1. COM0 is a serial line available on front panel or on rear of the VX304x and COM1 is available on the rear. To have SETUP displayed and EFI shell visible on a serial line it is necessary to enable the Console redirection on it. COM0 Console Redirection is enabled by default and COM1 is disabled by default.

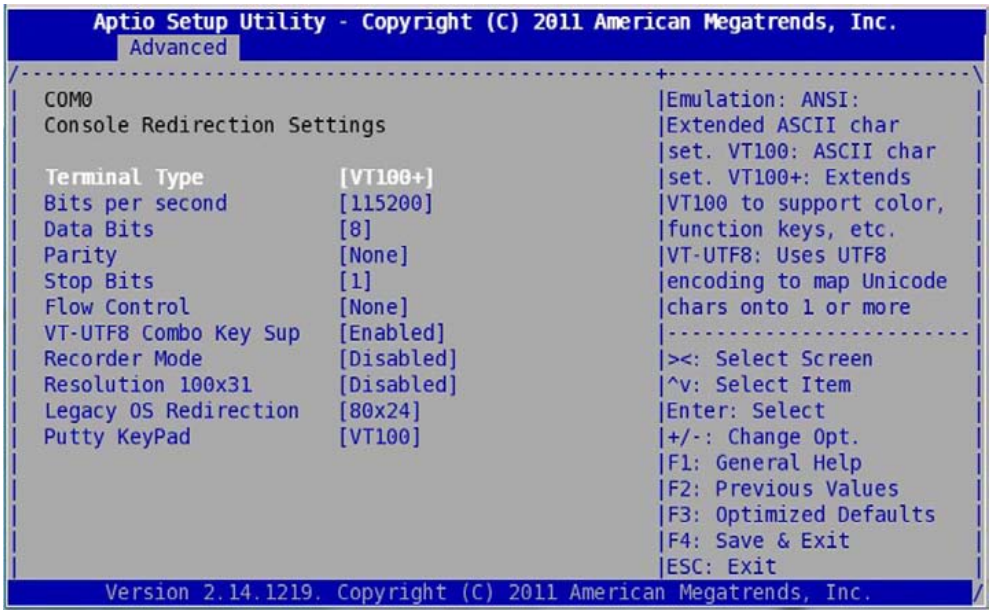


In case the user would like to display the PXE messages on serial COM1 instead of serial COM0, serial COM0 redirection must be disabled because only one serial port is selected by PXE.

4.4.2 COM0/COM1 Console Redirection Settings

This menu allows to configure several parameters for a serial line on which the console redirection has been enabled. The main configurable parameters are:

- Terminal Type
- Bits per second
- Data Bits
- Parity
- Stop Bits
- Flow Control

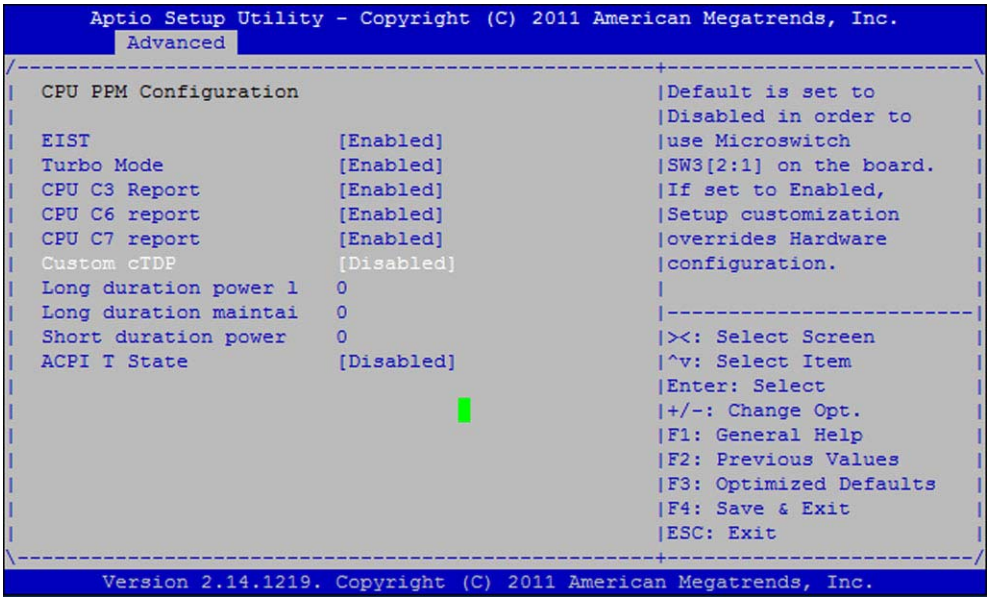


This shows the default settings.

4.5 CPU PPM Configuration

This menu can be used to:

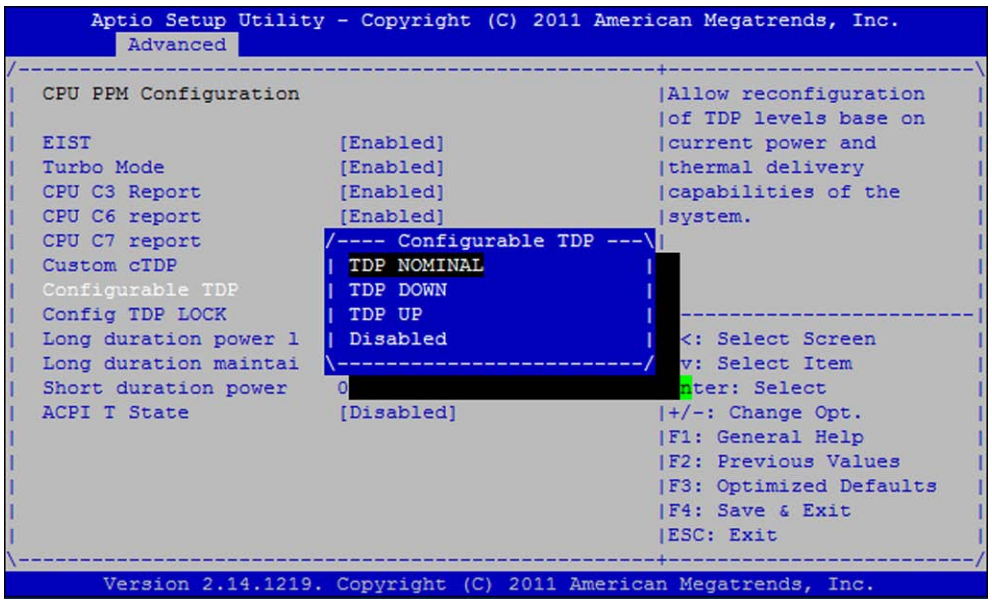
- Enable/disable the Turbo mode
- Configure the Thermal Design Power (TDP) (on VX3042 only)



Default of Custom cTDP setting is set to [Disabled] in order to use Hardware Microswitches SW3[1-2] on the board to set the CPU cTDP.

Switching this setting to [Enabled] allows the user to overtake the hardware configuration and allows manual selection for cTDP as shown in following paragraph.

» On VX3042 only:



Three TDP can be configured:

- TDP Nominal corresponding to 17W
- TPD Down corresponding to 14W
- TPD Up corresponding to 25W

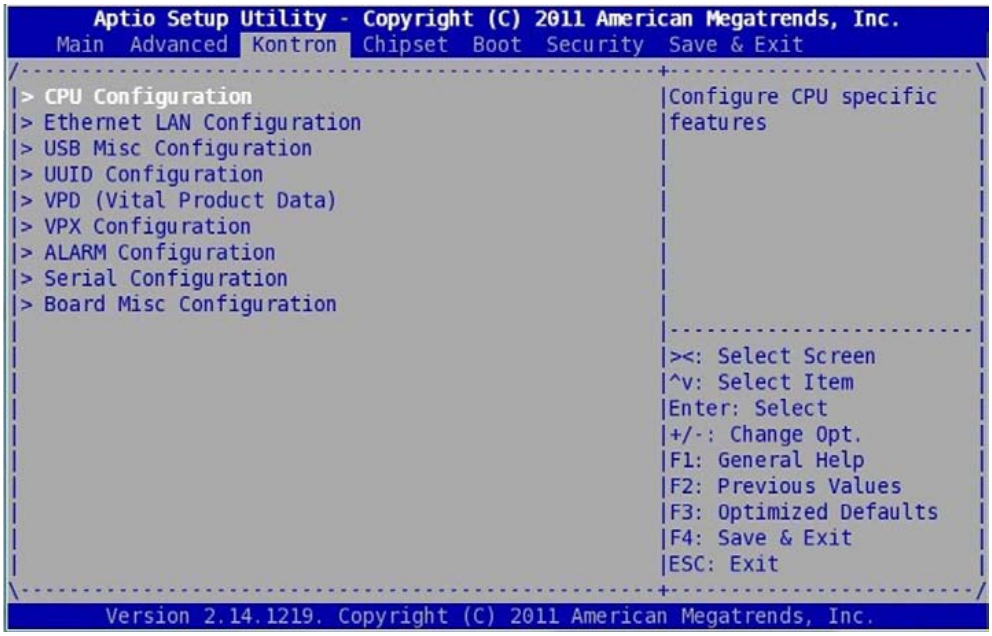
The setting “Disabled” will force the TDP to the nominal setting 17W.

Chapter 5 - Kontron Menu

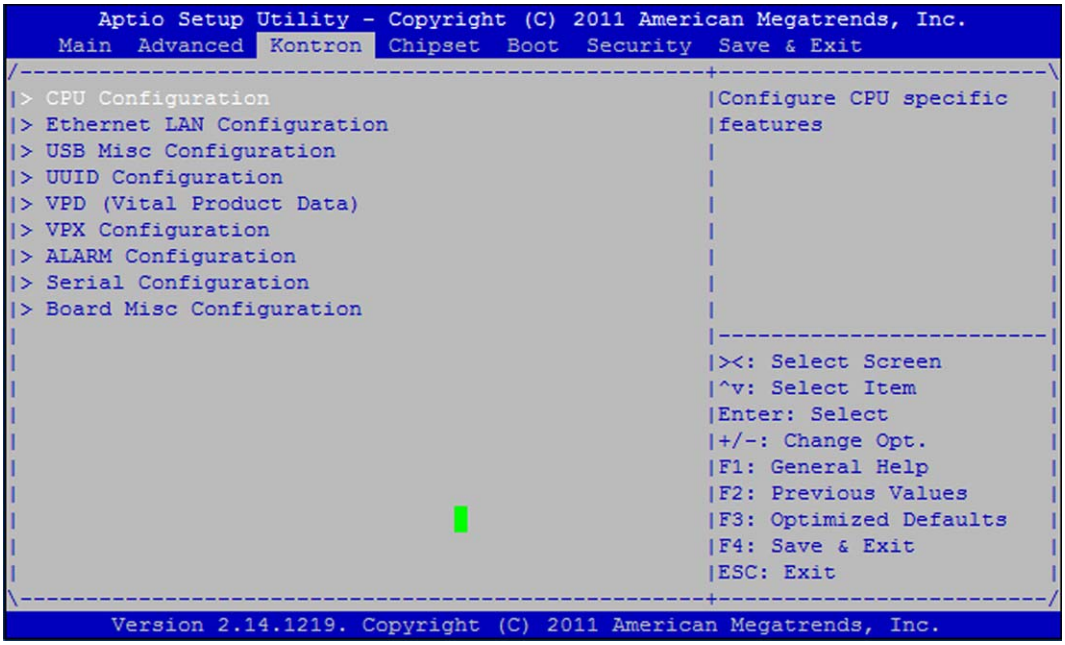
The Kontron Menu provides system-level controls to configure specific VX304x hardware design.

The different parameters are described in the following sections:

- `CPU Configuration` - Section 5.1 page 22
- `Ethernet LAN Configuration` - Section 5.2 page 24
- `USB Misc Configuration` - Section 5.3 page 25
- `UUID Configuration` - Section 5.4 page 26
- `VPD (Vital Product Data)` - Section 5.5 page 27
- `VPX Configuration` - Section 5.6 page 28
- `ALARM Configuration` - Section 5.7 page 33
- `Serial Configuration` - Section 5.8 page 34
- `Board Misc Configuration` - Section 5.9 page 35



5.1 CPU Configuration



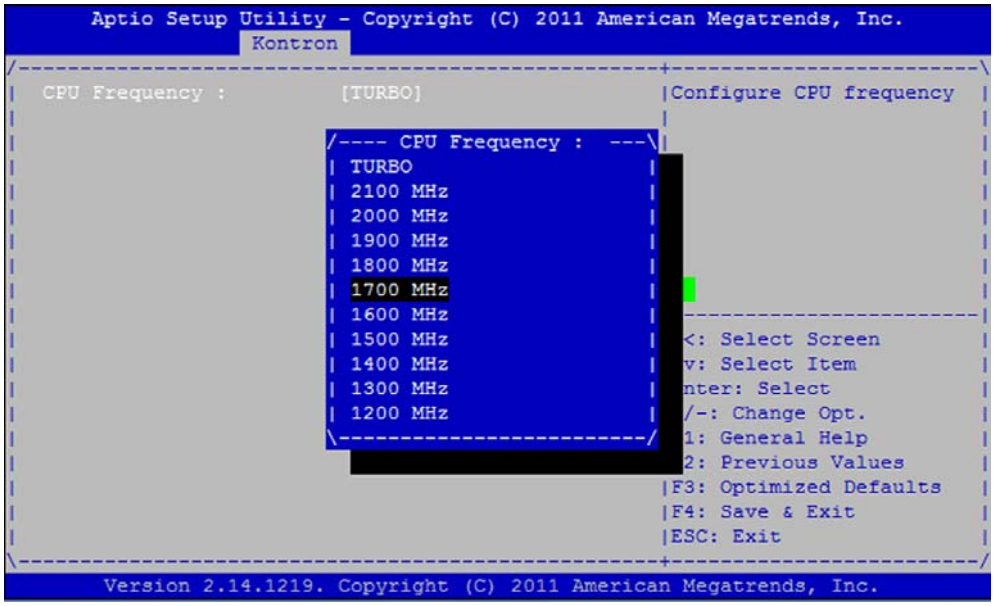
This menu is used to configure the CPU speed requested by user for VX3044 board. Only VX3044 boards are concerned by this parameter since CPU on those boards does not support Configurable TDP.

See section 4.5 page 19 for setting CPU configurable TDP for VX3042 board. By default, the CPU speed for VX3044 board is set to TURBO. In this mode, with Turbo Mode set in Advanced menu ⇨ CPU PPM Configuration ⇨ Turbo Mode [Enabled], the BIOS allows OS to obtain the maximum performance.

The CPU speed defaults to 1200 MHz if CPU is in idle state and, if the CPU load increases, the CPU speed can reach a maximum frequency of 3100 MHz. In "Turbo Mode", the CPU speed oscillates between 1200 and 3100 MHz according to the CPU load and power management (ACPI). Conversely, if user wants to set the CPU speed at a fixed frequency, the BIOS offers in this menu several CPU speed supported by the CPU on VX3044 board.

- For example:

  To set the CPU speed at a fixed CPU speed of 1700 MHz, select 1700 MHz as shown in picture below and validate by typing enter, then move to Save & Exit menu ⇨ Save Changes and Exit. In this mode "fixed frequency", the BIOS programs ACPI _PSS table to have ONLY one available CPU speed for OS, and this CPU speed is completely independent of the turbo mode either set to enabled or not as in previous configuration. In this case, the CPU speed does not oscillate under OS and will be set at the specified frequency under BIOS Setup.

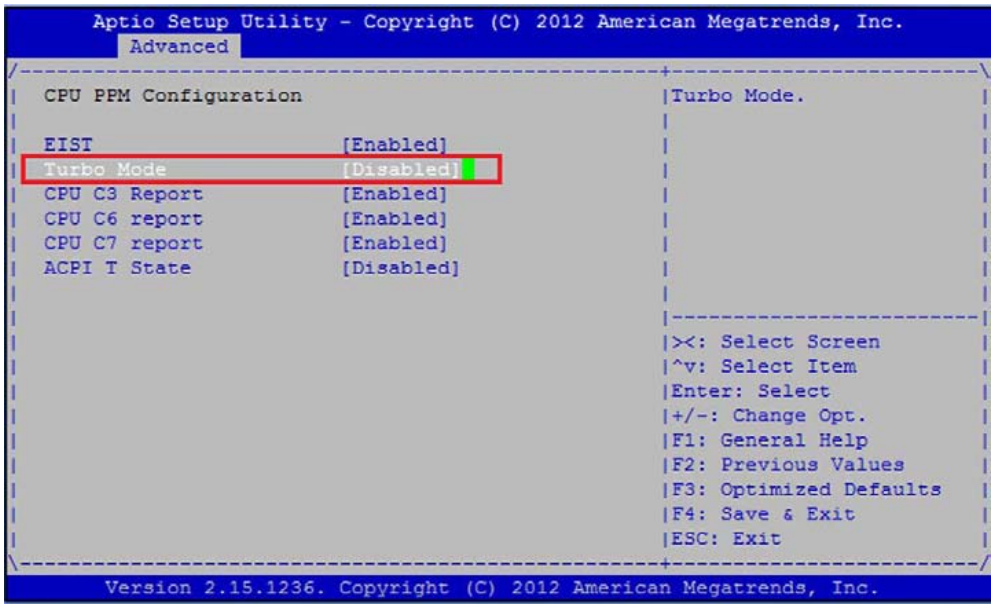


The microswitches SW3[1:2] allows CPU TDP configuration only for VX3042 board except for the SW3[1:2] = on:on combination: with this combination, the board generates PROHOT# signal to the CPU, and automatically, the CPU stays in low power mode. In this mode, the CPU speed is set to 800 MHz on VX3042 board and to 1200 MHz on VX3044 board. The BIOS does not perform any other action when this mode is selected and so, the CPU frequency parameter is no longer relevant.

If the user want to disable the Turbo mode on VX3044 board, we have to:

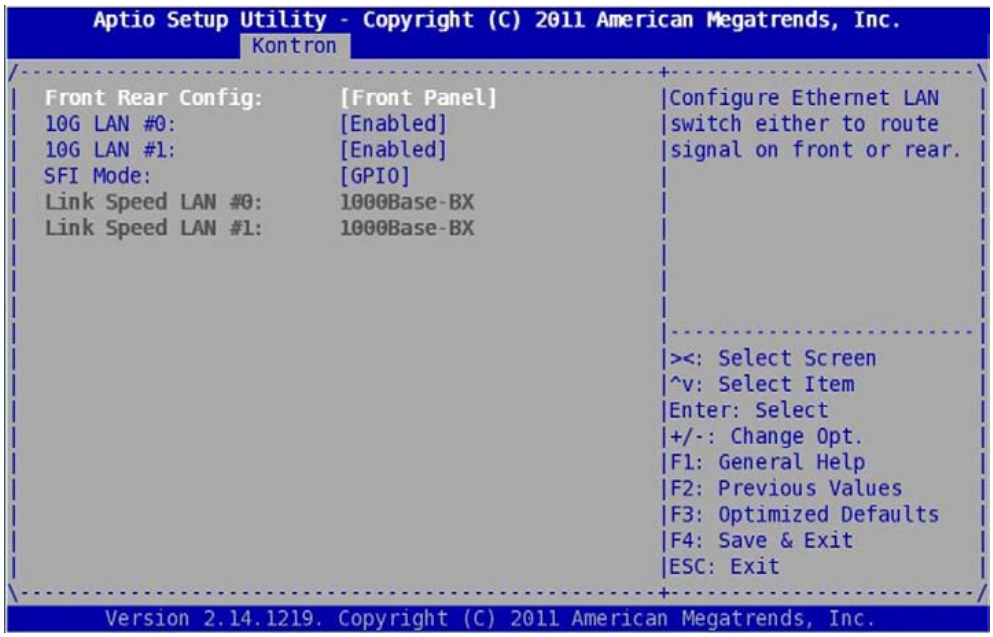
1. Set the desired CPU Frequency in **Kontron** > **CPU Configuration** menu
2. Save changes and Exit.

Automatically, the BIOS set to **[Disabled]** the Turbo Mode setting in **Advanced** -> **CPU PPM Configuration** menu.



The parameter CPU Frequency set to **TURBO** takes precedence on the CPU PPM Configuration Turbo Mode parameter.

5.2 Ethernet LAN Configuration



This menu is used to:

- Set the LAN#2 routed to the front panel or to the rear
- Enable/disable independently the 2 interfaces of the 10GbE i82599
- Display the link speed of the 2 10GbE interfaces according to the programmed EEPROM (refer to the `kmac` command in section 10.1.24 page 75)

By default, the `SFI mode` is set to `GPIO` and should not be changed.

5.3 USB Misc Configuration

The following option is displayed :

Set the `USB Keyboard Layout`:

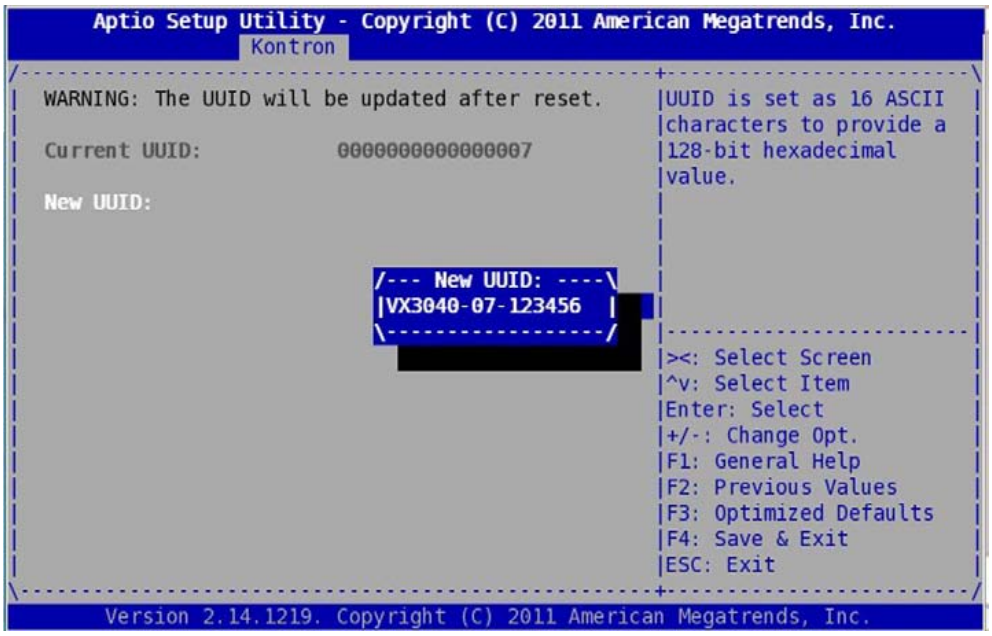
- `US Keyboard`
- `FR Keyboard`

Default is `US Keyboard`.

This option allows to set the type of USB keyboard used, Qwerty or Azerty.

As only the English language is supported under BIOS, accented characters are not managed. Moreover, the characters ° £ ¤ µ and § are not displayed either.

5.4 UUID Configuration

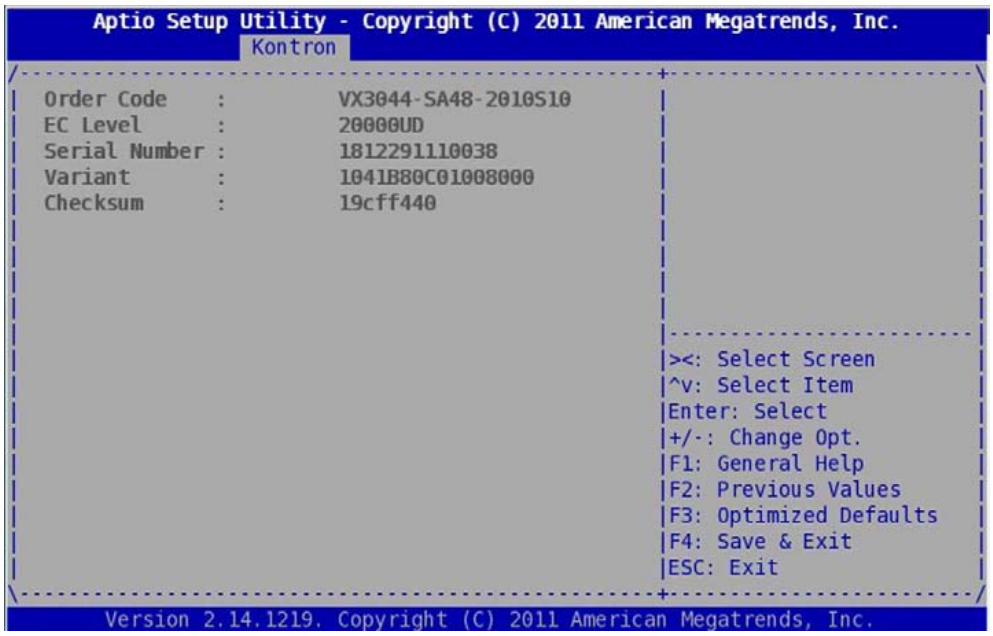


UUID stands for Universally Unique IDentifier also known as GUIDs (Globally Unique IDentifier). A UUID is 128 bits long, and can guarantee uniqueness across space and time. Please refer to RFC4122 documentation for more details about UUID.

The BIOS provides UUID to fill SMBIOS table and for PXE protocol. Default value of the UUID is set as an ASCII number equal to the Geographical Address of the board on the backplane.

This submenu provides ability for the user to modify the default value of the UUID (see picture above).

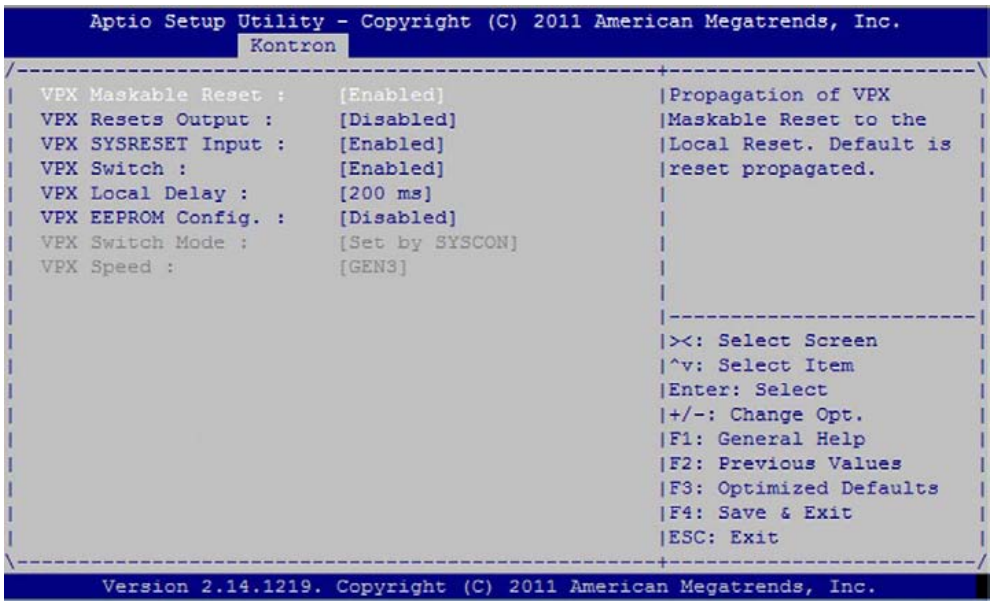
5.5 VPD – VITAL PRODUCT DATA



This menu displays the Vital Product Data (VPD) information for VX304x. VPD are stored in VX304x EEPROM.

- `Order Code`: Ordering code defining the type of Board
- `EC Level`: Engineering Change Level, gives the hardware level identification
- `Serial Number`: Board Serial Number
- `Variant`: A define coding the exact hardware configuration
- `Checksum`: Checksum value of VPD area

5.6 VPX Configuration



5.6.1 VPX Maskable Reset

The `VPX Maskable Reset` option allows to propagate or not the Maskable Reset from the VPX backplane to the board.

By default reset is `propagated`.

5.6.2 VPX Reset Propagation to VPX Backplane

The `VPX Resets Output` parameter allows to propagate the local resets of the board to the VPX backplane disregarding the state of the `VPX SYSCON#` signal.

Default is that only the VPX system controller board can control the propagation of the reset to the `VPX SYSRESET#` signal on VPX backplane.

Caution must be taken using this parameter in a multi-boards system because ALL boards plugged on the VPX backplane can be affected by the `VPX SYSRESET#` signal.
This parameter can be used in conjunction with the parameter `VPX SYSRESET` Input.

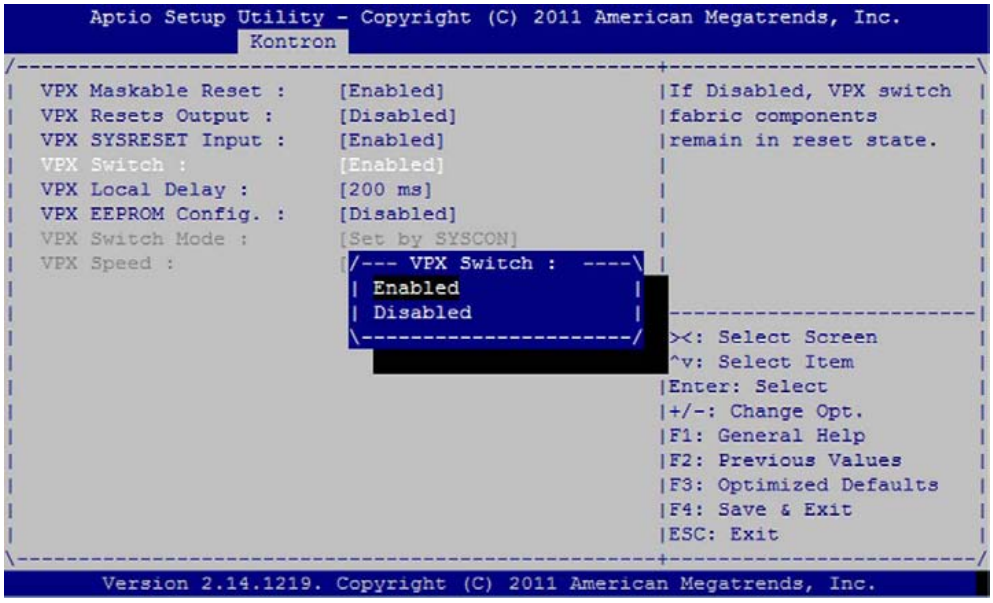
5.6.3 VPX SYSRESET Input

The `VPX SYSRESET Input` parameter allows to propagate or not the `VPX SYSRESET#` signal from the VPX backplane to the board.

If this parameter is set to `[Disabled]`, VPX backplane reset has no effect on the board.

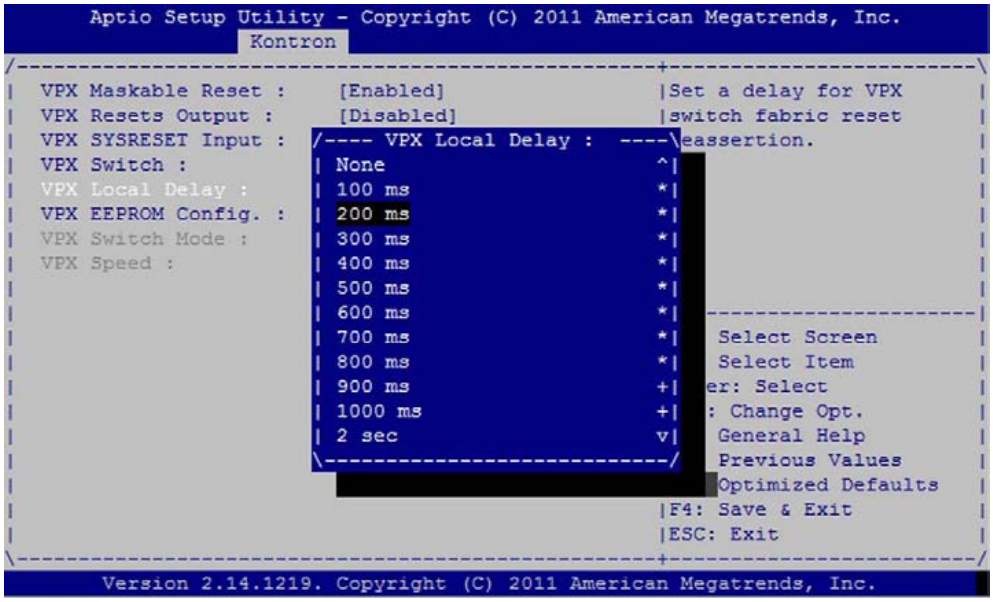
In a multi-boards configuration system, this parameter can be used in conjunction with the VPX Resets Output parameter.

5.6.4 VPX Switch



The `VPX Switch` option allows to maintain the VPX switch component in reset state or not. By default, this option must be set to `Enabled` in order to perform access on VPX backplane.

5.6.5 VPX Local Delay



Set `VPX Board delay`

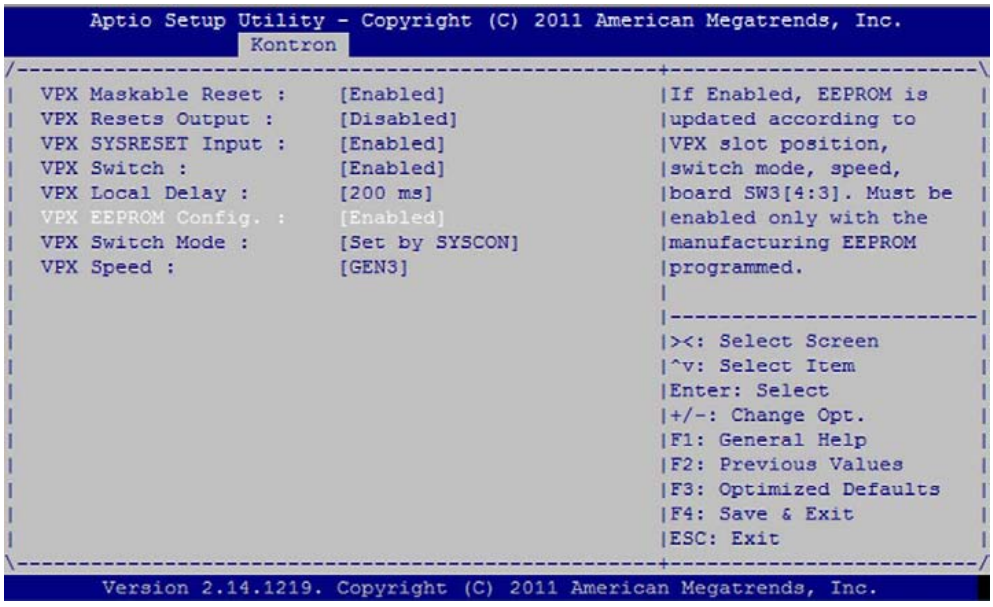
- Value are:

  `None`
  `100 ms`
  `200 ms`
  ..
  `1000 ms`
  `2 sec`
  `3 sec`
  `4 sec`
  `5 sec`

Default is `200 ms`.

This value should be tuned to delay the PCI-Express reset for VPX fabric discovery during boot process.

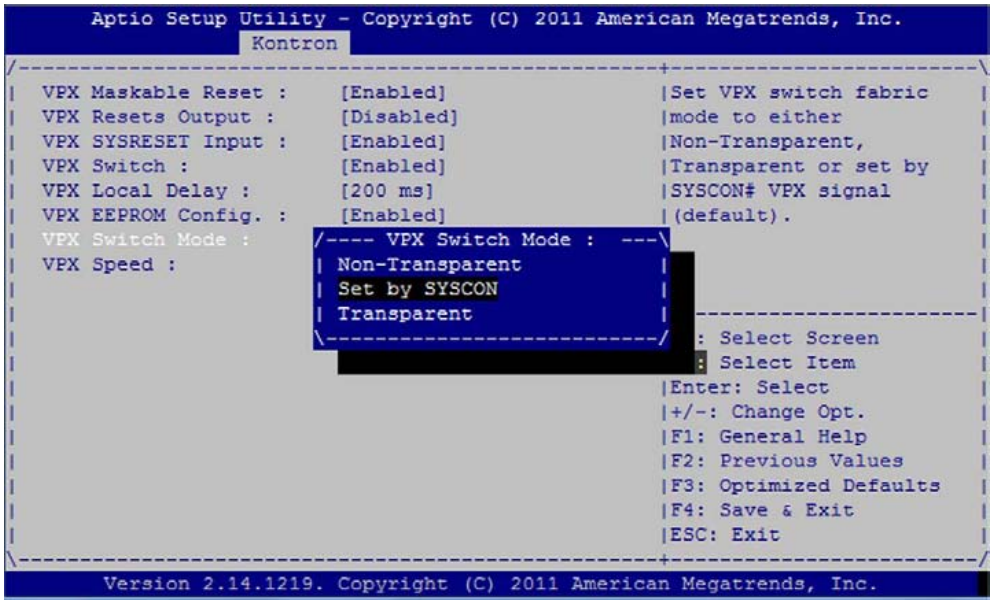
5.6.6 VPX EEPROM Configuration



The VPX Switch Fabric EEPROM can be configured dynamically by enabling this feature. By default, this parameter is set to `[Disabled]`: the EEPROM is programmed during manufacturing with a default binary image that configures the PCI-E switch in Non-Transparent mode, VPX port speed at Gen3 and a VPX link width of 1x8.

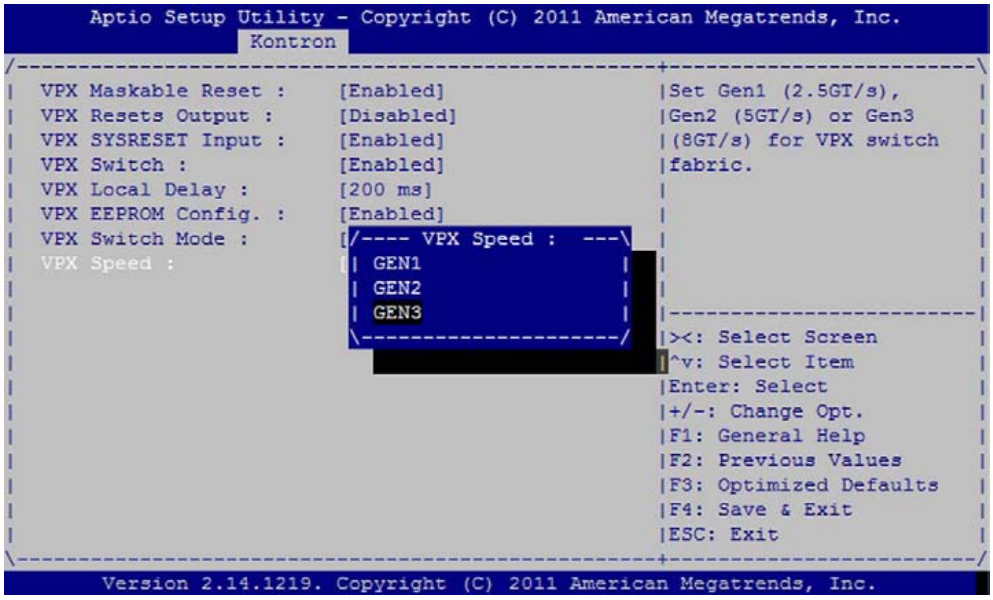
If this parameter is set to `[Enabled]`, the following features can be configured:

1. The Transparent Mode of the Switch Fabric:
   a. `Transparent`: the Switch Fabric is forced in Transparent Mode
   b. `Non-Transparent`: the Switch Fabric is forced in Non-Transparent Mode
   c. `Set by SYSCON`: the Switch Fabric is in Transparent Mode if the board is System Controller, and Non-transparent otherwise.



2. The VPX speed of the Link:
   a. `GEN1`: the speed is limited to 2.5 GT/s
   b. `GEN2`: the speed is limited to 5 GT/s
   c. `GEN3`: the speed is limited to 8 GT/s


This setting allows to force the VPX PCIe links to operate at a specific speed. This is only consistent if the hardware switch SW2.3 is set to OFF.



3. The VPX width of the Link

   This setting is related to the hardware switches SW3.3 and SW3.4:

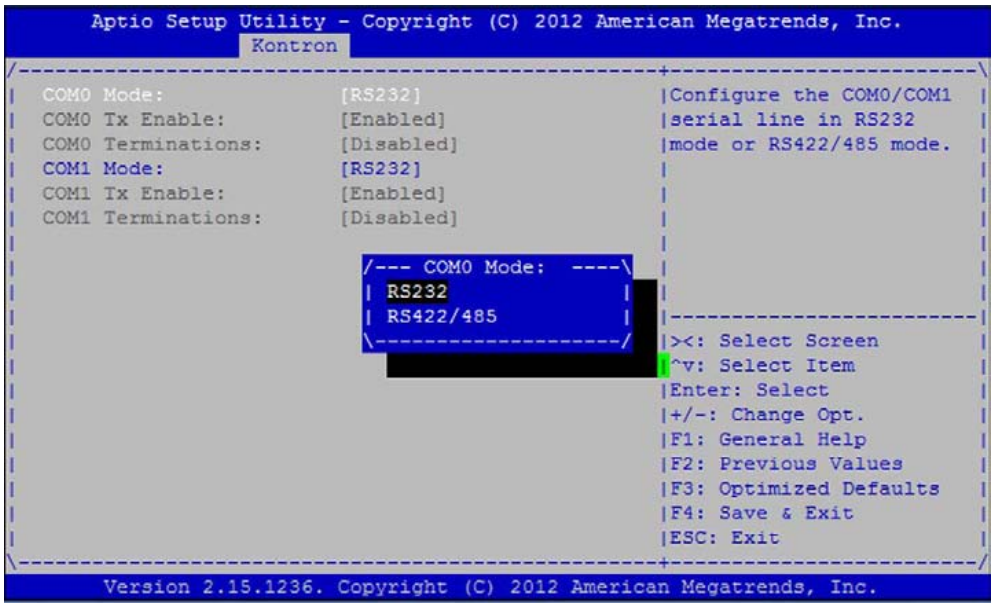
   a. [off:off]: 1x8
   b. [off:on]: 2x4
   c. [on:off]: 4x2
   d. [on:on]: reserved

5.7 ALARM Configuration

This menu allows user to prevent cPLD logic to turn off automatically the system in case of assertion of THRMTRIP# or PROCHOT# alerts.

It is highly recommended not to change the default setting for normal use. This parameter must be used with caution.

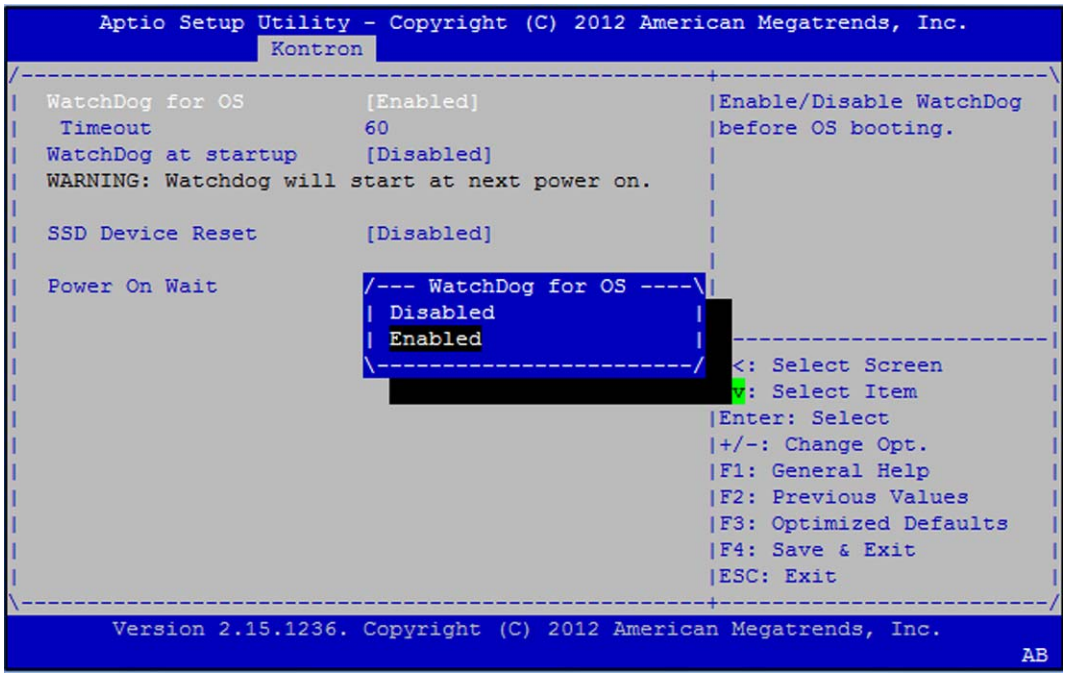
5.8 Serial Configuration



This menu allows user to select the mode for the COM0 or the COM1 serial port: the supported mode are RS-232 and RS-422/485.

User must turn off the system after saving to have the new Serial configuration taken into account.

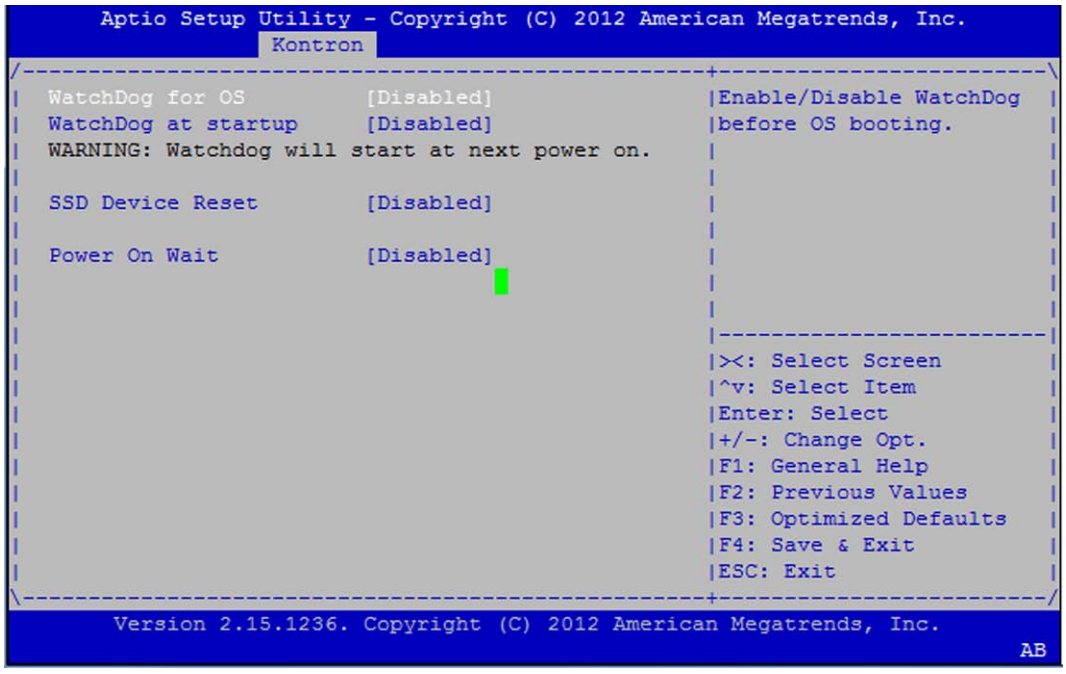
5.9 Board Misc Configuration



The WatchDog for OS option allows to disable (default setting) or enable the CPLD Watchdog Timer at OS boot time with a timeout value between 1 and 511 sec. The timeout value can be adjusted up and down by using the keys <+> or <->.

If enabled, the timer will be started at device boot time. Only the Power Mode mode is handled.

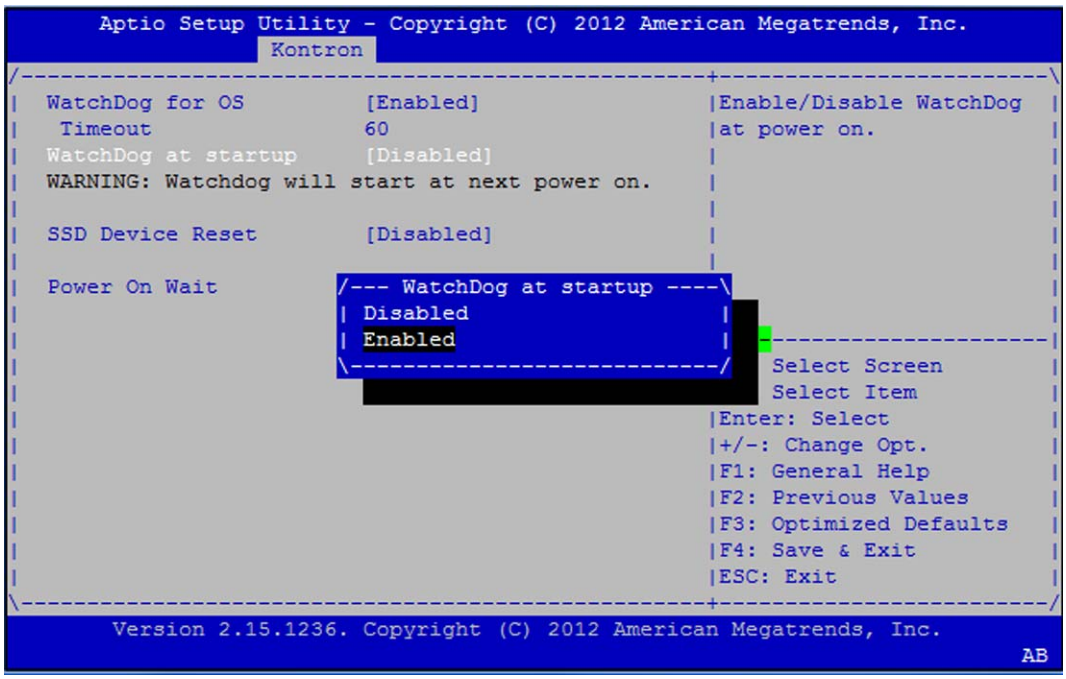
The WatchDog for OS setting is kept by the Setup even after a timeout has occurred.



The WatchDog at Startup option is a new feature that allows to disable (default setting) or enable the CPLD Watchdog Timer at Power-on.

The default timeout is 21 sec and is not configurable by setup.

The Watchdog at Startup will start only at next Power-On of the board. Only the Power Mode mode is handled.

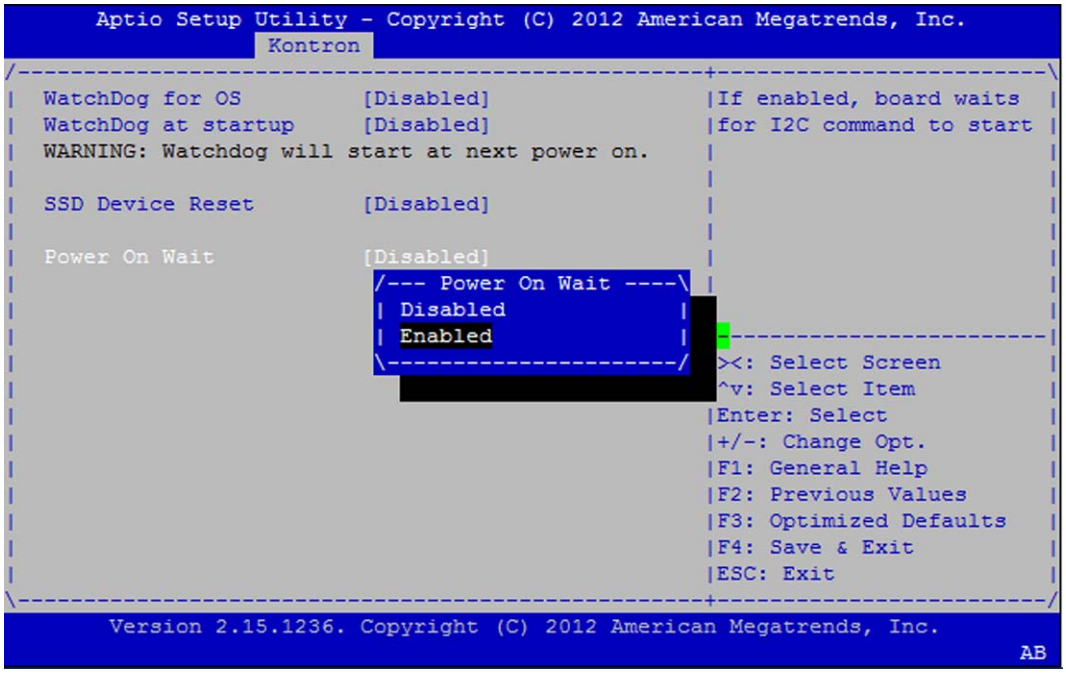


The SSD Device Reset parameter is a new feature that allows the onboard SSD device to be kept in reset state.

If enabled, the onboard SSD device will not appear in the SATA configuration menu at next power-on.

The Power On Wait parameter is a new feature that allows a VPX device to power-on/off the board using I2C backplane.

This feature is reserved for specific Hardware; DO NOT USE for normal operation.

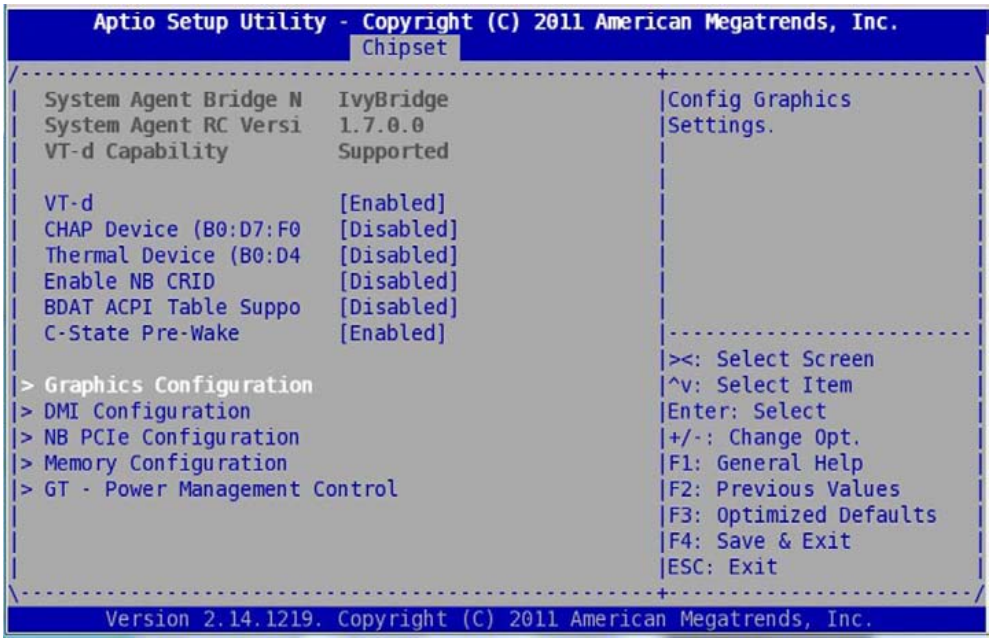


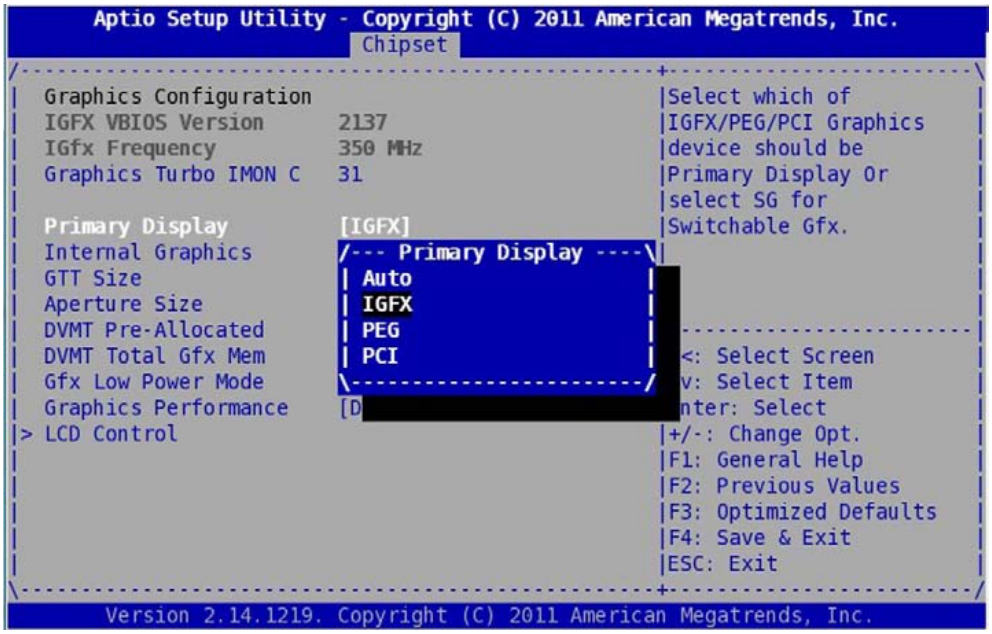
Chapter 6 - Chipset Menu

The **Chipset** menu provides system-level controls to configure the **chipset** devices settings.

The following paragraph describes 2 examples for setting graphic device as primary display and to force memory refresh cycle.

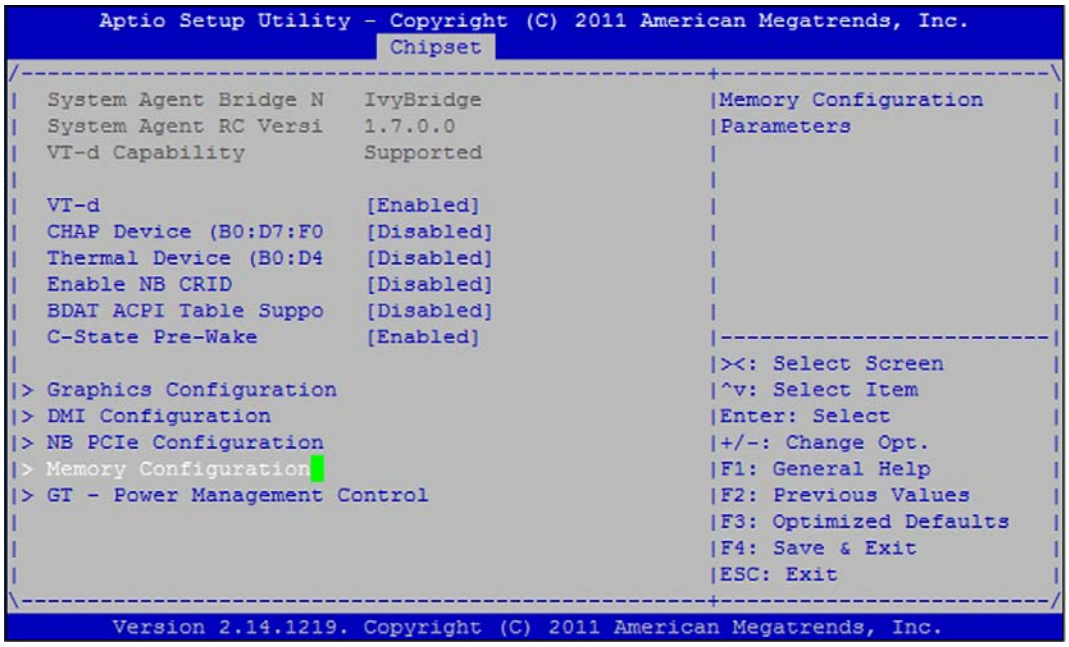
6.1 Graphics Configuration

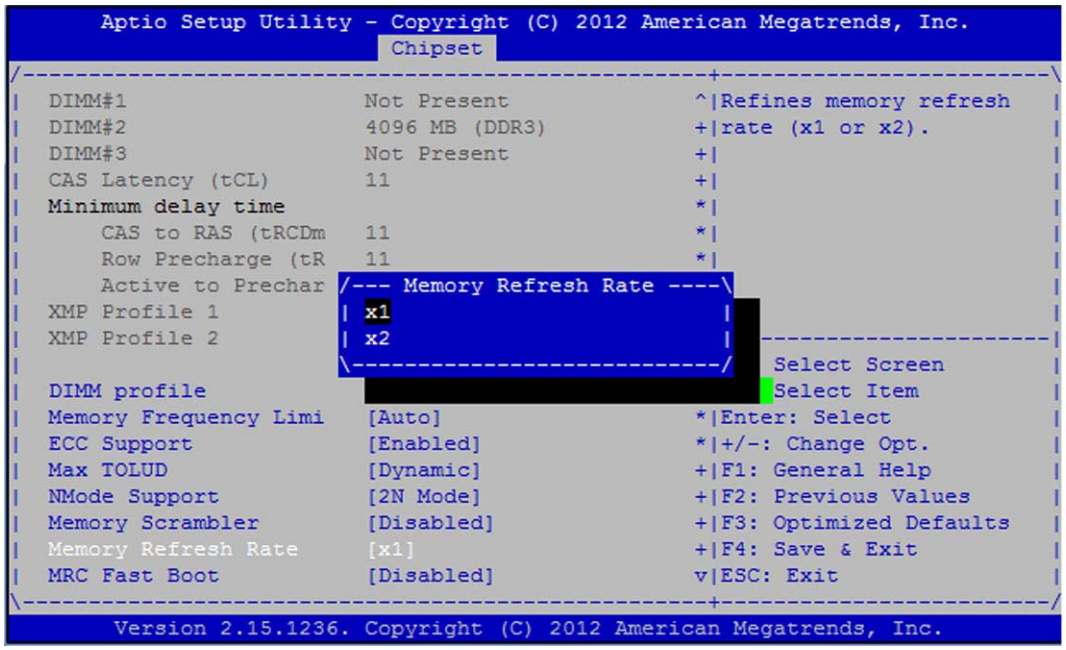




In this example, the IGFX, the internal video controller of the CPU, is selected as primary graphic display. The selected device will be available after saving changes and exiting setup.

6.2 Memory Configuration





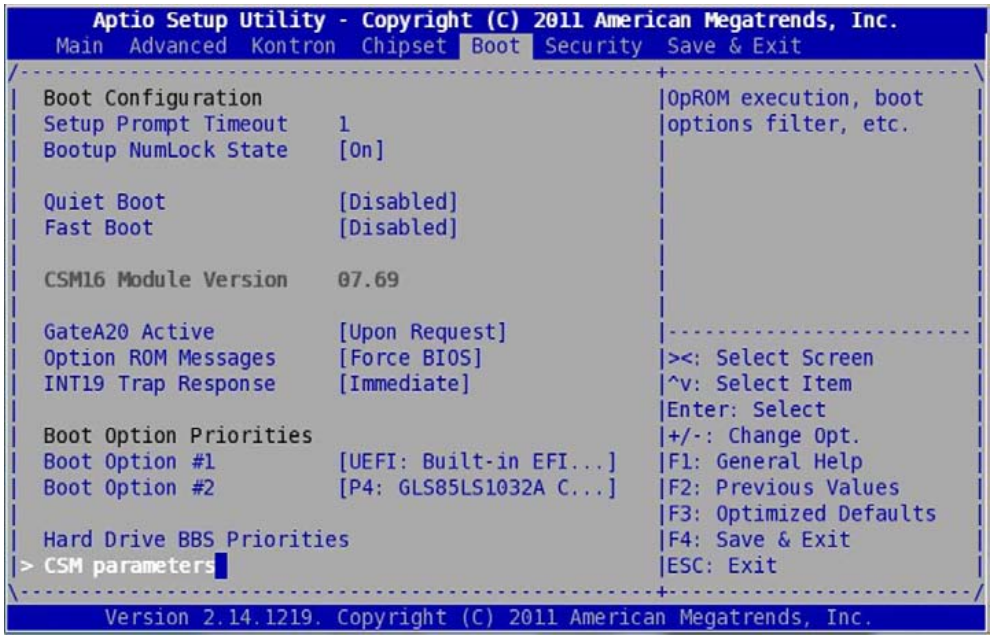
In this example, the Memory Refresh Rate refers to the DDR3 memory Refresh Rate mode which is by default set to ASR (Automatic Self-Refresh mode):

> if ASR is supported by memories part, the Refresh Rate is equal to 1X (64ms) if temperature is less than 85 °C and to 2X (32 ms) if temperature is above 85 °C.

> If the memories part does not support ASR, and temperature of the board exceed the 85 °C, it is necessary to refresh memory faster and so, user can set this parameter to `[Enabled]`: in this mode, the SRT (Self-Refresh Temperature) mode is used and the BIOS programs both the memory controller and DDR3 memories register to force SRT at 2X regardless of the temperature.

The other settings are Not intended to be changed.

Chapter 7 - Boot Menu



The Boot Menu allows user to configure the boot mode and to select the boot sequence of the available boot devices. Possible Boot settings are:

- `Quiet boot`: Section 7.1 page 41
- `Setup Prompt Timeout`: Section 7.2 page 41
- `Bootup NumLock State`: Section 7.3 page 41
- `Boot Option Priorities`: Section 7.4 page 42
- `Network Device BBS Priorities`: Section 7.5 page 43
- `Hard Drive BBS Priorities`: Section 7.6 page 45
- `CSM parameters` (for OpROM execution and boot options filter): Section 7.7 page 47

The other submenus are Not to be used.

The VX304x boot time is about 7s after a reset and 10s after a power on, assuming boot time end is when the EFI shell prompt appears. The boot time may change depending on whether a USB device is connected or not.

7.1 Quiet boot

Quiet Boot setting when enabled allows to hide BIOS boot message such as:

```
Version 2.14.1219. Copyright (C) 2011 American Megatrends, Inc.
BIOS Date: 12/14/2012 14:37:25 Ver: 1APTJ
```

Press <DEL> or <F2> to enter setup.

7.2 Setup Prompt Timeout

`Setup Prompt Timeout` menu sets the number of tenth of a second for setup up activation key.

Setup Prompt Timeout

- Enter the number of tenth of a second. For example 60 for 6 seconds.

7.3 Bootup Numlock State

This menu selects the keyboard numlock state

Set `Bootup NumLock State`

- `On`
- `Off`

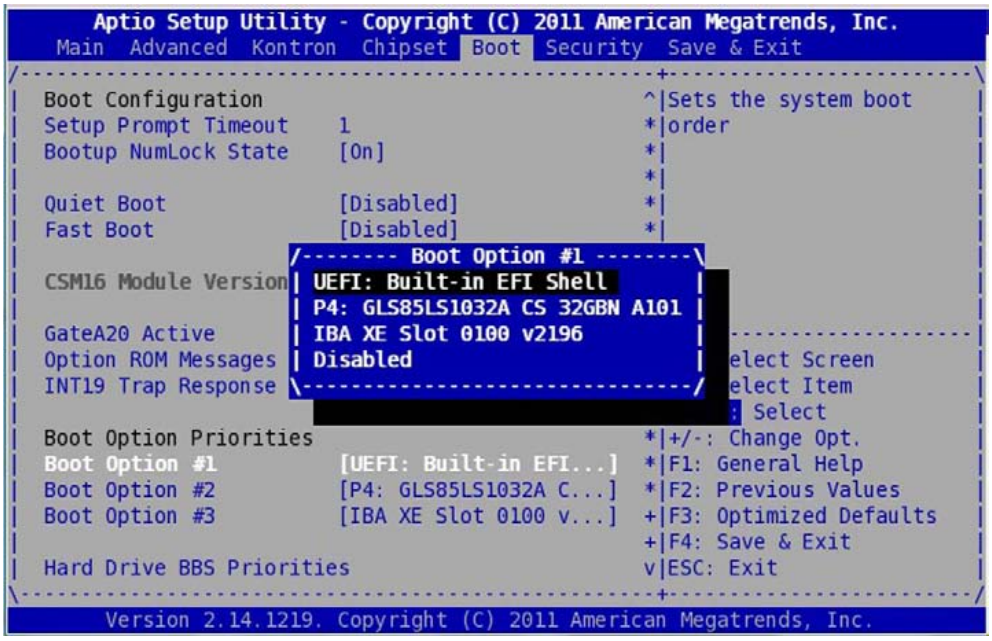
Default is `On`

7.4 Boot Option Priorities

This menu specifies the boot order from the available boot devices list.

The first device into the list is the first device that will be booted. If the boot is rejected (for example unsuccessful PXE boot) then the second device in the list will be used for boot and so on.

Here is a example of boot device list:



To change the boot device ordering

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item)
- Use <+> or <-> control keys to move up/down the selected device item into the list

The possible family boot device can be SATA, USB or Gigabit Ethernet (Gbe). In the boot device item list only one item per family will appear. If more than one device is available for booting (for example 2 SATA disk or 3 Ethernets for PXE) then 2 new submenus can appear below the item list. So it can be:

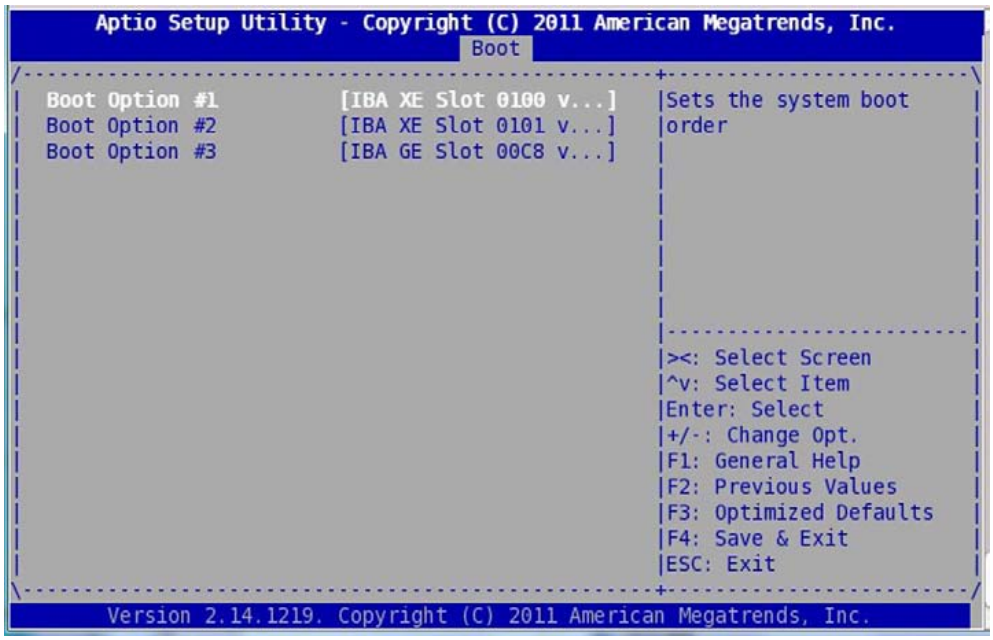
- Hard Drive BBS Priorities → This is the submenu for setting a SATA or USB boot order or deleting a SATA & USB boot possibility.
- Network Device BBS Priorities → This is the submenu for setting a Gbe boot order or deleting a Gbe boot possibility

7.5 Network Device BBS Priorities (when PXE ROM Enabled)

The setting allows to configure the Ethernet boot device sequence for PXE.

When PXE ROM has been enabled, Ethernet devices become available for PXE booting (3 Ethernets interfaces). In this case a new submenu is displayed in Boot Setup menu. See image below:

Select this parameter to display available Ethernet Devices.



The Network Device "IBA GE Slot 00C8" is related to the Ethernet Interface of the Intel® 82579 device, LAN#2.

The Network Devices "IBA XE Slot 0100" and "IBA XE Slot 0101" are related to the Ethernet Interfaces of the Intel® 82599 Dual Port device, LAN#0 and LAN#1.

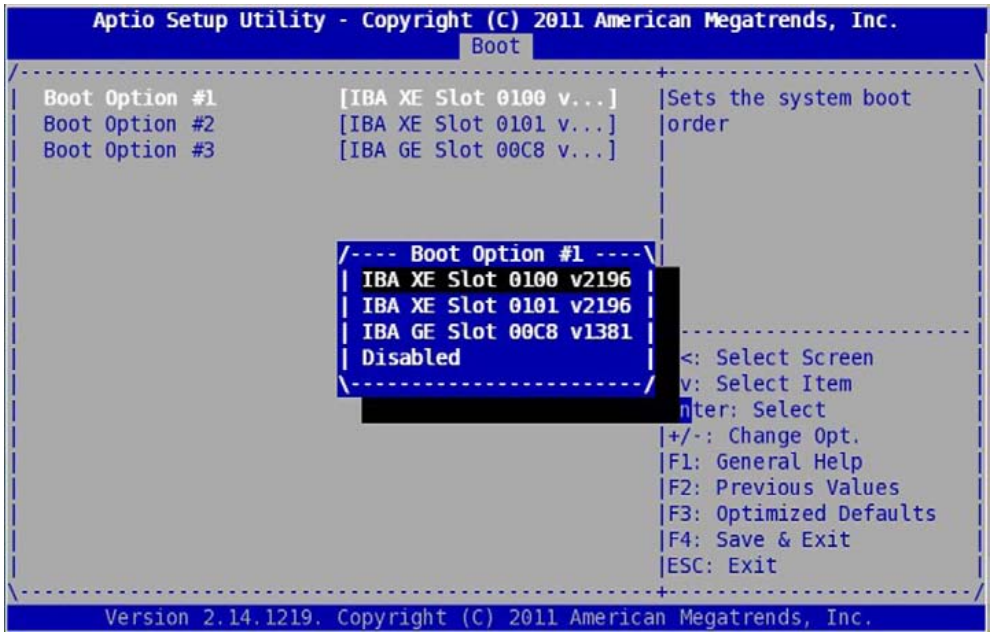
To change the PXE boot device ordering

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item)
- Use <+> or <-> control keys to move up/down the selected device item in the list

To disable one of the PXE boot device

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item)
- <Enter> to validate the choice

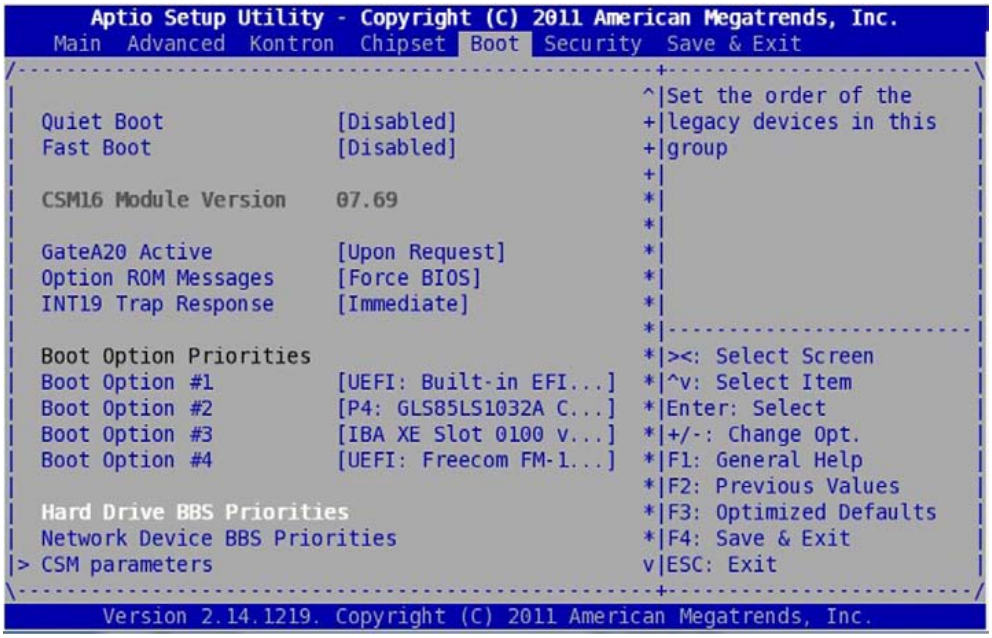
A new submenu appears (see image) , select Disabled to disable the PXE device



7.6 Hard Drive BBS Priorities

The setting allows to configure the SATA, USB boot device sequence.

This submenu appears when several SATA disk or USB device are present. See image:



Select this menu to see the SATA & USB boot devices available and to be able to disable it or to reorganize the boot sequence.

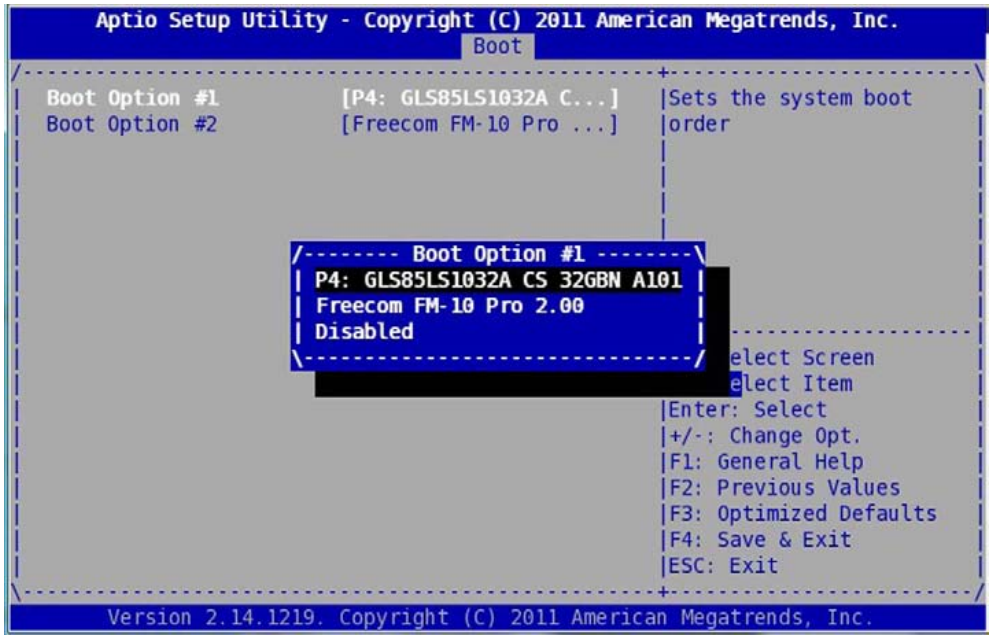
To change the boot devices ordering

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item
- Use <+> or <-> control keys to move up/down the selected device item in the list

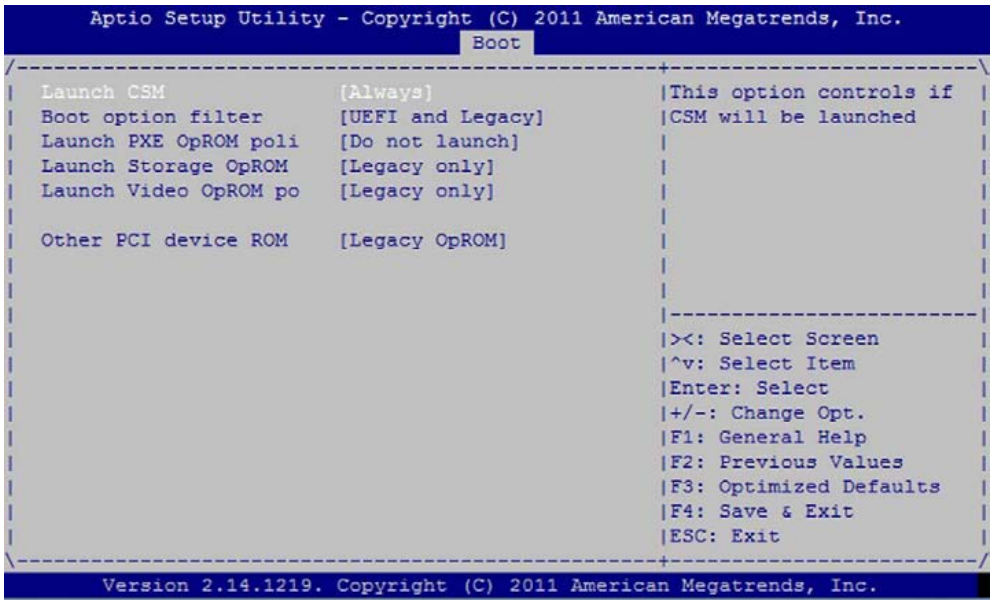
To disable one of the boot devices

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item
- <Enter> to validate the choice

A new submenu appears (see image) , select `Disabled` to disable the SATA or USB device



7.7 CSM Parameters



7.7.1 Launch CSM Parameter

This parameter must be set to [Always] in order to have bootable Legacy devices.

7.7.2 Boot Option Filter

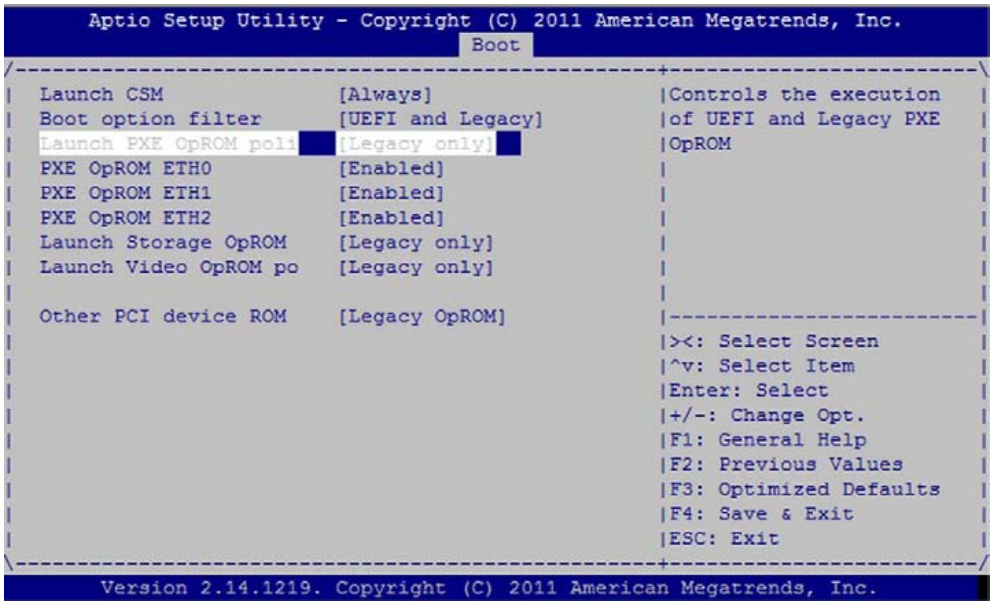
This parameter must be set to [UEFI and Legacy] in order to have bootable Legacy devices.

7.7.3 Launch PXE OpROM Policy

Defaut is [Do not launch]; set this parameter to [Legacy Only] to have Network Bootable Devices access in boot menu.

PXE Option ROM can be individually Enabled or Disabled for each Ethernet Interface.

By default, all PXE Option ROM are Enabled.



7.7.4 Launch Storage OpROM

Default is [Legacy only]; SATA RAID devices are allowed to boot to. Set to [Do no launch] if you want to disable boot to SATA RAID devices.

7.7.5 Launch Video OpROM Policy

This parameter must be set to [Legacy Only] to enable graphics on Legacy OSes.

7.7.6 Other PCI Device ROM

This parameter must be set to [Legacy OpROM] to enable Option ROM for Legacy PCI devices other than Network, Storage of Video Option ROMs.

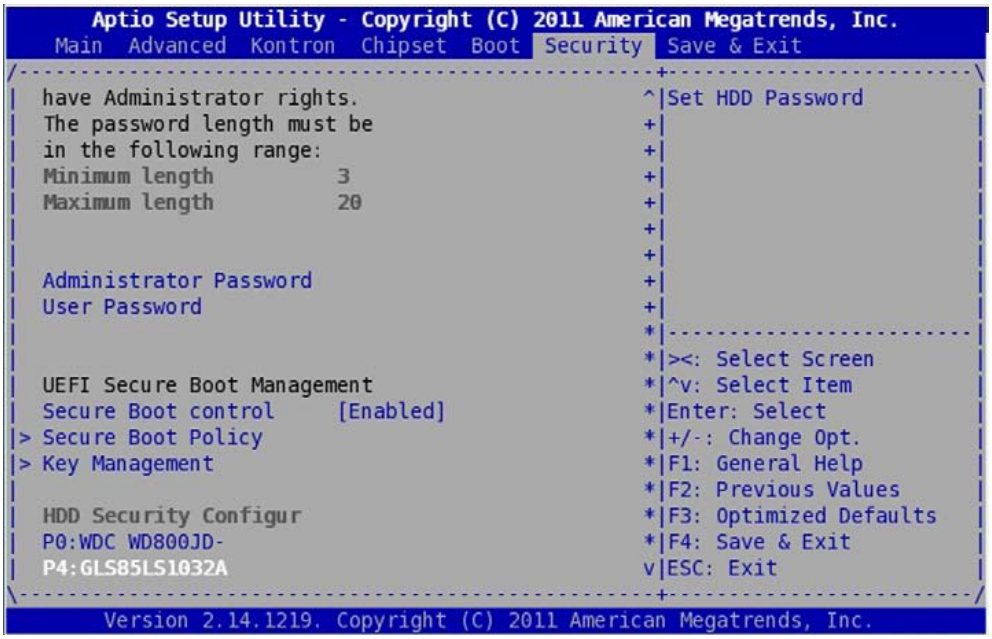
Chapter 8 - Security Menu

The `Security` Menu allows the user to set a password for SETUP or boot access.

If ONLY the Administrator's password is set, then this only limits access to Setup and is only asked for when entering Setup. If ONLY the User's password is set, then this is a power on password and must be entered both to boot or enter Setup. In Setup, the User will have Administrator rights.

A HDD Security Configure submenu can appear when a SATA disk is connected.

This submenu is Reserved and Not To Be Used



Note UEFI Secure Boot Management settings are also displayed but at this moment the Secure Boot module is not enabled in the BIOS release.

8.1 Enter Administrator or user password

To enter the password:

- Select the administrator or user password item
- A pop-up window appears and proposes to create a new password
- Enter a password with lenght between 1 to 20 characters
- You will have to confirm the password
- The password will then be saved if the command "`Save changes`" is launched in `Save & Exit` Menu.

During the next reboot, if the **<F2>** key is pressed, then the password becomes mandatory to enter the SETUP menu.

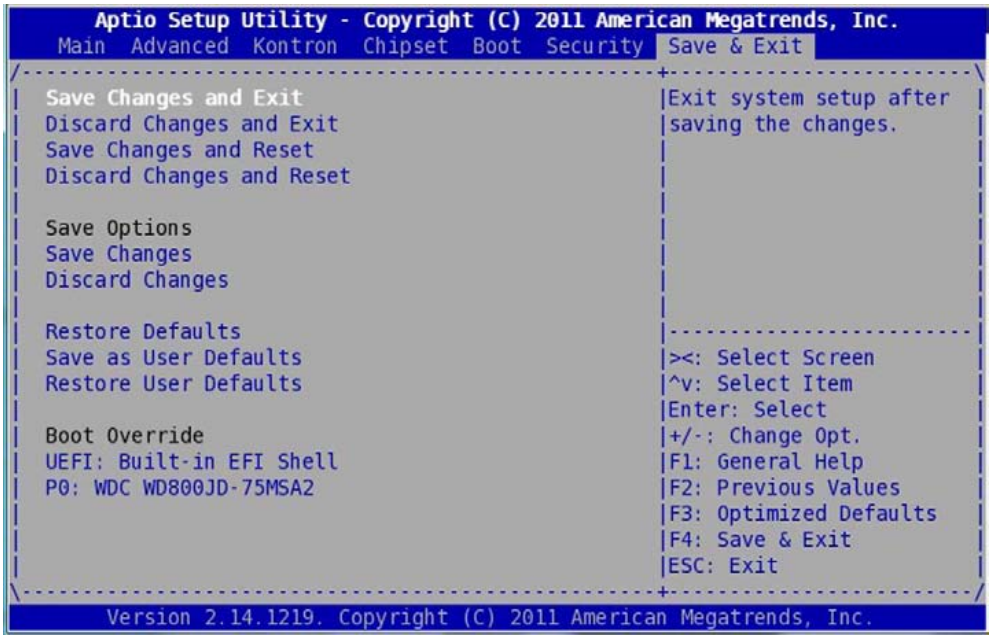
When the user password is set, the password is required to enter setup menu and to execute the BIOS boot device selection.

To suppress the password

- Select the administrator or user password item
- A pop-up window appears and proposes to enter a password
- Enter previous password
- A pop-up window appears and proposes to enter a new password
- Then type an empty password
- You will have to confirm empty password
- The password will be deleted if the command "`Save changes`" is launched in the `Save & Exit` Menu.

If the password is lost, the solution to unlock is to flash the BIOS again.

Chapter 9 - Save & Exit Menu



This Menu is used to save a new SETUP configuration, discard changes, restore default SETUP values, record a customized SETUP and override the boot device sequence. This menu does not appear as the first window when entering SETUP. It is necessary to navigate from the main menu to find it.

Available submenus are

- `Save Changes and Exit`: section 9.1 page 54
- `Discard Changes and Exit`: section 9.1 page 54
- `Save Changes and Reset`: section 9.1 page 54
- `Discard Changes and Reset`: section 9.1 page 54
- `Save Changes`: section 9.2 page 54
- `Discard Changes`: section 9.2 page 54
- `Restore Defaults`: section 9.2 page 54
- `Save as User Defaults`: section 9.3 page 54
- `Restore User Defaults`: section 9.3 page 54
- `Boot Override`: section 9.4 page 54

9.1 Option with Exit or Reset

With one of the following options the user can choose to save or record the changes in SETUP and to reset or exit SETUP. Reset will perform a complete board reset while Exit will execute the Boot Device Selection for booting. To apply SETUP parameter modifications a reset is mandatory.

Select desired item and <Enter>

- `Save Changes and Exit`
- `Discard Changes and Exit`
- `Saving the changes and reset`
- `Save Changes and Reset`

9.2 Option to Save Discard Restore SETUP

SETUP modification can simply be Saved or Discarded without exiting BIOS SETUP. Also manufacturing default SETUP parameters can be restored with Restore Defaults menu.

Select desired item and <Enter>

- `Save Changes`
- `Discard Changes`
- `Restore Defaults`

9.3 Saving a User Configuration

The current SETUP configuration can be saved as the user configuration and can be restored the same way as the default one.

Select desired item and <Enter>

- `Save as User Defaults`
- `Restore User Defaults`

9.4 Boot Override

The current sequence of boot devices can be overriden by this menu.

- Select a device from the list (Use the <↑> or <↓> to highlight the desired item
- <Enter> to immediately Boot on this device

Chapter 10 - EFI SHELL

EFI Shell is a boot shell available on the VX304x that is accessible in the boot device list. EFI Shell is launched automatically if no other boot device is connected to the VX304x. If EFI shell is not the primary boot device then it is necessary to enter the SETUP menu to access it. For this, enter <F2> during boot process to enter SETUP. Then navigate to `Save & Exit` Menu and select `UEFI shell` in Boot override menu.

EFI SHELL is available by default on the graphical display or serial line COM0 configured at 115200 bauds.

EFI SHELL implements a set of command utilities and can be used to access or display various resources, to flash a new BIOS image or execute a start-up script.

10.1 EFI Shell Command

The `Help` command or `(?)` displays all the available command. Use option `-b` to display command screen by screen. Use `help` + command (like `VX304x> help help`) to have the detail of a command syntax

» VX304x> help

Command Name	Description	See Section
`?`	Displays the EFI Shell command list or verbose command help	10.1.19 page 71
`alias`	Displays, creates, or deletes EFI Shell aliases	10.1.1 page 57
`amlview`	AML view utility	10.1.2 page 58
`bcfg`	Boot configuration utility	10.1.3 page 59
`cd`	Displays or changes the current directory	10.1.4 page 60
`cls`	Clears standard output and optionally changes background color	10.1.5 page 61
`connect`	Connects one or more EFI drivers to a device	10.1.6 page 61
`cpuutil`	CPU information utility	10.1.7 page 61
`date`	Displays or changes the current system date	10.1.8 page 62
`devices`	Displays the list of devices managed by EFI drivers	10.1.9 page 62
`dh`	Displays EFI handle information	10.1.10 page 63
`disconnect`	Disconnects one or more EFI drivers from a device	10.1.11 page 65
`drivers`	Displays the EFI driver list	10.1.12 page 66
`dumpacpi`	Prints ACPI Tables	10.1.13 page 67
`dumpaml`	Prints AML dump	10.1.14 page 67
`echo`	Controls batch file command echoing or displays a message	10.1.15 page 68
`exit`	Exits the EFI Shell environment	10.1.16 page 68
`for`	Executes commands for each item in a set of items	10.1.17 page 69
`goto`	Forces batch file execution to jump to specified location	10.1.18 page 70
`help`	Displays the EFI Shell command list or verbose command help	10.1.19 page 71
`if`	Executes commands in specified conditions	10.1.20 page 72

Command Name	Description	See Section
ifconfig	UEFI network modification utility	10.1.21 page 73
kdiag	Performs board diagnostics - **Available ONLY if ordered.**	10.1.22 page 73
kflash	Kontron SPI flasher	10.1.23 page 74
kmac	Kontron MAC Address viewer	10.1.24 page 75
kpld	Kontron PLD Commands	10.1.25 page 76
ksata	Kontron SATA Configurator	10.1.26 page 76
ktemp	Kontron Board Temperature	10.1.27 page 77
kvpd	Kontron VPD Information	10.1.28 page 78
kvpx	Kontron VPX Configurator	10.1.29 page 79
ls	Displays a list of files and subdirectories in a directory	10.1.30 page 82
map	Displays or defines mappings	10.1.31 page 84
mem	Displays the contents of memory	10.1.32 page 88
memmap	Displays the memory map	10.1.33 page 90
mm	Displays or modifies MEM/MMIO/IO/PCI/PCIE address space	10.1.34 page 92
pause	Prints a message and waits for keyboard input	10.1.35 page 94
pci	Displays PCI device list or PCI function configuration space	10.1.36 page 96
reconnect	Reconnects one or more EFI drivers to a device	10.1.37 page 101
reset	Resets the system	10.1.38 page 101
set	Displays or modifies EFI Shell environment variables	10.1.39 page 102
shift	Shifts batch file input parameter positions	10.1.40 page 103
smbiosview	Displays SMBIOS information	10.1.41 page 104
smbutil	SMBus utility	10.1.42 page 105
time	Displays or changes the current system time	10.1.43 page 105
timezone	Displays or sets time zone information	10.1.44 page 105

10.1.1 alias

Displays, creates, or deletes aliases in the EFI Shell environment.

```
ALIAS [-d|-v] [sname] [value]
```

`-d`	Deletes an alias
`-v`	Volatile variable
`sname`	Alias name
`value`	Original name

4. '`sname`' should not be an internal EFI Shell command.
5. '`value`' can be an internal EFI Shell command, a script, or an EFI application. However, any other values are also acceptable.
6. `ALIAS` values are stored in EFI NVRAM and will be retained between boots unless the '`-v`' option is specified.
7. `ALIAS` will not add a nonvolatile alias when a volatile alias of the same name already exists, or vice versa.

> Examples:

- To display all aliases in the EFI Shell environment:

```
Shell> alias
             md   : mkdir
             rd   : rm
```

- To create an alias in the EFI Shell environment:

```
Shell> alias myguid guid
Shell> alias
             md           : mkdir
             rd           : rm
             myguid       : guid
```

- To delete an alias in the EFI Shell environment:

```
Shell> alias -d myguid
Shell> alias
             md   : mkdir
             rd   : rm
```

- To add a volatile alias in the current EFI environment, which has a star * at the line head. This volatile alias will disappear at next boot.

```
Shell> alias -v fs0 floppy
Shell> alias
             md           : mkdir
             rd           : rm
             * fs0        : floppy
```

10.1.2 amlview

Views ACPI1.0b, ACPI2.0, or ACPI3.0 AML in EFI Shell Environment.

```
usage: AMLView [<AML file>]
```

Also AmlView proposes its own shell syntax

```
Shell> amlview
Welcome to AmlView on EFI Shell (Version 0.01)
DefinitionBlock ("Dsdt.aml", "DSDT", 2, "ALASKA", "A M I", 24)
```

AmlView > help

```
EXEC      <NodeName>          : Prints the result of the method node.
CAT       <NodeName>          : Prints the node content.
LS [-R] [<NodeName>]          : Lists the node name. (-R means recursive)
CD       [<NodeName>]         : Changes current node dir.
QUIT                          : Quits Current Command Prompt.
HELP                          : Prints Help Information.
(NodeName format - [\]AAAA[.BBBB[...]])
```

10.1.3 bcfg

`bcfg` is an utility for boot configuration.

```
bcfg driver|boot [dump [-v]][add # file "desc"][rm #] [mv # #]
```

`driver`	selects boot driver list
`boot`	selects boot option list
`dump`	dumps selected list
`-v`	dumps verbose (includes load options)
`add`	adds '`file`' with '`desc`' at position #
`addp`	adds '`file`' with '`desc`' at position #.Use hard drive path
`addh`	adds '`handle`' with '`desc`' at position #.Use Handle
`rm`	removes #
`mv`	moves # to #

> Example:

The following example shows the ability to change boot device order without entering in BIOS setup.

```
Shell > bcfg boot dump
The boot option list is:
01. VenMedia(5023b95c-db26-429b-a648-bd47664c8012) "UEFI: Built-in EFI Shell " OPT
02. BBS(HD,,0x0) "Hard Drive " OPT
03.
PciRoot(0x0)/Pci(0x1,0x0)/Pci(0x0,0x0)/MAC(0000de404176,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.
0.0) "UEFI: Intel(R) 10 Gigabit Network Connection" OPT
04.
PciRoot(0x0)/Pci(0x1,0x0)/Pci(0x0,0x1)/MAC(0000de404177,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.
0.0) "UEFI: Intel(R) 10 Gigabit Network Connection" OPT
05.  PciRoot(0x0)/Pci(0x19,0x0)/MAC(0000de404175,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.0.0)
"UEFI: Intel(R) 82579LM Gigabit Network Connection" OPT

Shell > bcfg boot mv 5 2
bcfg: boot option 5 moved to 2

Shell > bcfg boot dump
The boot option list is:
01. VenMedia(5023b95c-db26-429b-a648-bd47664c8012) "UEFI: Built-in EFI Shell " OPT
02.  PciRoot(0x0)/Pci(0x19,0x0)/MAC(0000de404175,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.0.0)
"UEFI: Intel(R) 82579LM Gigabit Network Connection" OPT
03. BBS(HD,,0x0) "Hard Drive " OPT
04.
PciRoot(0x0)/Pci(0x1,0x0)/Pci(0x0,0x0)/MAC(0000de404176,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.
0.0) "UEFI: Intel(R) 10 Gigabit Network Connection" OPT
05.
PciRoot(0x0)/Pci(0x1,0x0)/Pci(0x0,0x1)/MAC(0000de404177,0x0)/IPv4(0.0.0.0,0x0,DHCP,0.0.0.0,0.0.0.0,0.0.
0.0) "UEFI: Intel(R) 10 Gigabit Network Connection" OPT
```

10.1.4 cd

Displays or changes the current directory.

```
CD [path]
```

`path` The relative or absolute directory path

1. Type CD without parameters to display the current fs and directory.
2. There must be at least one blank space between CD and path.
3. The 'path' parameter supports certain special characters:
   - '.' refers to the current directory.
   - '..' refers to the parent directory.
   - '\' used at the beginning of the path refers to the root directory of the current filesystem.
4. CD can only be used to change directories in the current file system.

> Examples:

- To change the current filesystem to the mapped fs0 filesystem:

```
Shell> fs0:
```

- To change the current directory to subdirectory 'efi':

```
fs0:\> cd efi
```

- To change the current directory to the parent directory (fs0:\):

```
fs0:\efi\> cd ..
```

- To change the current directory to 'fs0:\efi\tools':

```
fs0:\> cd efi\tools
```

- To change the current directory to the root of the current fs (fs0):

```
fs0:\efi\tools\> cd \
fs0:\>
```

- To change volumes with cd will not work!! For example:

```
fs0:\efi\tools\> cd fs1:\  !!!! will not work !!!!
must first type fs1: then cd to desired directory
```

- To move between volumes and maintain the current path.

```
fs0:\> cd \efi\tools
fs0:\efi\tools\> fs1:
fs1:\> cd tmp
fs1:\tmp> cp fs0:*.* .
copies all of files in fs0:\efi\tools into fs1:\tmp directory
fs0:\>
```

10.1.5 cls

Clears the standard output and optionally changes the background color.

```
CLS [color]
```

color	New background color
0	Black
1	Blue
2	Green
3	Cyan
4	Red
5	Magenta
6	Yellow
7	Light gray

1. If no parameters are specified, this command clears the standard output device. The background color is not changed.

> Examples:

- To clear standard output without changing the background color:

```
fs0:\> cls
```

- To clear standard output and change the background color to cyan:

```
fs0:\> cls 3
```

- To clear standard output and change the background to the default color:

```
fs0:\> cls 0
fs0:\>
```

10.1.6 connect

Reserved - Not To be Used

10.1.7 cpuutil

Reserved - Not To be Used

10.1.8 date

Displays or changes the current system date.

```
date [mm/dd/[yy]yy]
```

mm	Month of date to set, range: 1 - 12
dd	Day of date to set, range: 1 - 31
yyyy	Year of date to set, range: 1998 - 2099

1. Short year format:
   yy: 98=1998, 99=1999, 00=2000, 01=2001, ..., 97=2097.
2. Long year format:
   yyyy: 1998 - 2099, other values are invalid.
3. EFI may behave unpredictably if illegal date values are used.

10.1.9 devices

Displays the list of devices managed by EFI drivers.

```
DEVICES [-b] [-l XXX]
```

-b	Displays one screen at a time
l XXX	Displays devices using the specified ISO 639-2 language

Display Format:

CTRL	The handle number of the EFI device
TYPE	The device type: [R] Root Controller [B] Bus Controller [D] Device Controller
CFG	A managing driver supports the Driver Configuration Protocol
DIAG	A managing driver supports the Driver Diagnostics Protocol
#P	The number of parent controllers for this device
#D	The number of drivers managing the device
#C	The number of child controllers produced by this device
DEVICE NAME	The name of the device from the Component Name Protocol

10.1.10 dh

Displays EFI handle information.

```
DH [-l lang] [handle | -p prot_id] [-d] [-v]
```

handle	Handles number in hexadecimal format
-p	Protocol ID
-d	Displays EFI Driver Model related information
-l	Displays information in the specified ISO 639-2 language
-v	Displays verbose information

1. When neither '**handle**' nor '**prot_id**' is specified, a list of all the device handles in the EFI environment is displayed.
2. The '**-d**' option displays EFI Driver Model related information including parent handles, child handles, all drivers installed on the handle, etc.
3. The '**-v**' option displays verbose information for the specified handle including all the protocols on the handle and their details.
4. If the '**-p**' option is specified, all handles containing the specified protocol will be displayed. Otherwise, the '**handle**' parameter has to be specified for display. In this case, the '**-d**' option will be enabled automatically if the '**-v**' option is not specified.

> **Examples:**

- To display all handles one screen at a time:

```
Shell > dh -b
```

Handle dump

```
 1: Image(CORE_DXE)
 2:
 3: DevPath (..d(0xb,0xdaf51000,0xdaff0fff))
 4: DevPath (..d(0xb,0xdad90000,0xdafaffff))
 5: DevPath (..d(0xb,0xda6a5004,0xdad60003))
 6:
 7: DpathUtil DpathToText DpathFromText Decompress
 8:
 9:
 A:
 B: UnicodeCollation2
 C: HiiFont HiiString HiiDatabase HiiConfRouting
 D:
 E:
 F: Image(CpuDxe) ImageDevPath (..e536-4e88-b3a0-b77f78eb34fe))
10:Image(Runtime) ImageDevPath (..383a-41eb-a8ee-4498aea567e4))
11:
12:
(...)
```

- To display detailed information for handle 10:

```
Shell > dh 10
```

```
Handle 10 (D8576F98)
    Image (D87D9E40)   File:Runtime
        ParentHandle..: D931BF18
        SystemTable...: DA4B5F18
        DeviceHandle..: D930E918
        FilePath.....: FvFile(cbc59c4a-383a-41eb-a8ee-4498aea567e4)
        ImageBase.....: DA4FD000 - DA50F4C0
        ImageSize.....: 124C0
        CodeType.....: RT_code
        DataType.....: RT_data
    ImageDpath (D8576E98)
        Hardware Device Path for Memory Mapped
        Memory Type (11: DA6A5004-DAD60003)
        Media Device Path for PIWG FV
        AsStr: 'MemoryMapped(0xb,0xda6a5004,0xdad60003)/FvFile(cbc59c4a-383a-41eb-a8ee-
        4498aea567e4)
```

- To display all handles associated with the 'diskio' protocol:

```
Shell > dh -p diskio
```

```
Handle dump by protocol 'DiskIo'
    194: DevPath (../Pci(0x1f,0x2)/Sata(0x0,0x0))DiskIo BlkIo
    196: DevPath (..BR,0xed32b4ef,0x800,0xfa000))DiskIo BlkIo
    197: DevPath (..xed32b4ef,0xfa800,0x9408000))DiskIo BlkIo
    195: DevPath (../Pci(0x1f,0x2)/Sata(0x4,0x0))DiskIo BlkIo
    198: DevPath (..cd-5e2eaf41eed3,0x800,0x800))DiskIo BlkIo
    199: DevPath (..b1ac8b8cd38b,0x1000,0xfa000))DiskIo BlkIo ESP
    19A: DevPath (..06bd43318,0xfb000,0x3aa7800))DiskIo BlkIo
```

- To display all handles associated with the 'Image' protocol and break when the screen is full:

```
Shell > dh -p Image -b
```

```
Handle dump by protocol 'Image'
     1: Image(CORE_DXE)
     F: Image(CpuDxe) ImageDevPath (..e536-4e88-b3a0-b77f78eb34fe))
    10: Image(Runtime) ImageDevPath (..383a-41eb-a8ee-4498aea567e4))
    19: Image(AmiBoardInfo) ImageDevPath (..ae55-4288-829d-d22fd344c347))
    1B: Image(EBC) ImageDevPath (..73d0-11d4-b06b-00aa00bd6de7))DebugSupport EbcInterp
    1D: Image(CpuPolicyDxe) ImageDevPath (..00a9-4de7-b8e8-ed7afb88f16e))
    1E: Image(CpuSmmSaveRes) ImageDevPath (..2133-1ba2-800a-b9c00accb17d))
    1F: Image(MiscSubclassDxe) ImageDevPath (..55d9-4a33-93fc-5a3eb128de21))
    20: Image(SBRun) ImageDevPath (..056e-4888-b685-cfcd67c179d4))
```

22: Image(ActiveBios) ImageDevPath (..fe0f-4251-b772-4b098a1aec85))
24: Image(PchReset) ImageDevPath (..2e30-4793-9bed-74f672bc8ffe))
26: Image(PchSerialGpio) ImageDevPath (..3466-4c06-b1cc-1c935394b5c2))
28: Image(SmmControl) ImageDevPath (..ab78-491b-b583-c52b7f84b9e0))
29: Image(WdtDxe) ImageDevPath (..f027-4ca7-bfd0-16358cc9e453))
2A: Image(iFfsDxePolicyInit) ImageDevPath (..e3f3-4e9e-90a3-2a991270219c))
2B: Image(AsfTable) ImageDevPath (..505b-4b50-99cd-a32467fa4aa4))
2C: Image(PlatformInfo) ImageDevPath (..cb8d-421c-b854-06231386e642))
2D: Image(IdeSMART) ImageDevPath (..809f-45cf-a377-d77bc0cb78ee))
2F: Image(SmbiosGetFlashData64) ImageDevPath (..7e20-4f20-91a1-190439b04d5b))
30: Image(S3Save) ImageDevPath (..4424-46a2-9943-cc4039ead8f8))
31: Image(CpuInitDxe) ImageDevPath (..78cd-4480-8678-c6a2a797a8de))
37: Image(PciHostBridge) ImageDevPath (..e55e-4d6a-a3a5-5e4d72ddf772))

Press ENTER to continue, 'q' to exit: ...

10.1.11 disconnect

Reserved - Not To Be Used

10.1.12 drivers

Displays the EFI drivers list.

```
DRIVERS [-l XXX]
```

-l	Displays drivers using the specified ISO 639-2 language
Display Format:	
DRV	Handles number of the EFI driver
TYPE	Driver type:
	[B] - Bus Driver
	[D] - Device Driver
CFG	Driver supports the Driver Configuration Protocol
DIAG	Driver supports the Driver Diagnostics Protocol
#D	Number of devices managed by the driver
#C	Number of child devices produced by the driver
DRIVER NAME	Name of the driver from the Component Name Protocol
IMAGE NAME	File path from which the driver was loaded

> Example:

- To display the list:

```
Shell> drivers
            T   D
D           Y C I
R           P F A
V  VERSION  E G G #D #C DRIVER NAME                          IMAGE NAME
== ======== = = = == == ================================== ==================
3F 00000010 B - -  1  2 AMI Generic LPC Super I/O Driver    CORE_DXE
9C 000C03F4 ? - -  -  - Intel(R) GOP Driver [3.0.12.1012]   IntelIvbGopDriver
9D 001B03EF ? - -  -  - Intel(R) GOP Driver [1.0.27.1007]   IntelSnbGopDriver
9E 00010000 ? - -  -  - AMI File System Driver              FileSystem
A0 00020502 B - -  1 24 <UNKNOWN>                           PciBus
B7 00000010 D - -  1  - PCH Serial ATA Controller Initializ SataController
B9 00000001 B - -  1  2 AMI AHCI BUS Driver                 AHCI
BA 03011000 B - X  2  2 Intel(R) 10GbE Driver 3.1.10 EFIx64 E3110X4
BB 05001200 B X X  1  1 Intel(R) PRO/1000 5.0.12 PCI-E      IntelGigabitLanx64
C0 00000001 ? - -  -  - IDER Controller Init Driver         IdeRController
C1 00000010 ? - -  -  - PCI Serial Driver                   PciSerial
D4 00000010 B - -  2  2 <UNKNOWN>                           Terminal
D5 00000010 B - -  1  1 <UNKNOWN>                           Terminal
D8 0000000A B - -  3  3 ARP Network Service Driver          ArpDxe
D9 0000000A D - -  3  - Simple Network Protocol Driver      SnpDxe
DA 0000000A B - -  3 12 MNP Network Service Driver          MnpDxe
DB 0000000A D - - 21  - UEFI PXE Base Code Driver           UefiPxeBcDxe
DD 0000000A D - -  3  - TCP Network Service Driver          TcpDxe
DE 0000000A B - -  3  3 DHCP Protocol Driver                Dhcp4Dxe
DF 0000000A D - -  3  - IP4 CONFIG Network Service Driver   Ip4ConfigDxe
E0 0000000A B - -  3 21 IP4 Network Service Driver          Ip4Dxe
E1 0000000A B - -  6  3 MTFTP4 Network Service              Mtftp4Dxe
E2 0000000A B - - 18 15 UDP Network Service Driver          Udp4Dxe
E3 0000000A D - -  3  - DHCP6 Protocol Driver               Dhcp6Dxe
E4 0000000A B - -  3 12 IP6 Network Service Driver          Ip6Dxe
...
```

```
...
E5 0000000A D - -  3  - MTFTP6 Network Service Driver       Mtftp6Dxe
E6 0000000A B - -  9  6 UDP6 Network Service Driver         Udp6Dxe
E7 0000008A D - -  3  - AMI USB Driver                      UHCD
E9 0000008A B - -  3  2 USB bus                             UHCD
EA 00000001 ? - -  -  - USB Hid driver                      UHCD
EB 00000001 ? - -  -  - USB Mass Storage driver             UHCD
EC 00000001 ? - -  -  - AMI USB CCID driver                 UHCD
111 00000010 ? - -  -  - <UNKNOWN>                           BIOSBLKIO
112 00000024 B - -  1  1 BIOS[INT10] Video Driver            CsmVideo
113 00000010 ? - -  -  - <UNKNOWN>                           <UNKNOWN>
118 00000010 D - -  7  - <UNKNOWN>                           CORE_DXE
119 00000010 D - -  1  - <UNKNOWN>                           CORE_DXE
11A 00000010 B - -  2  2 <UNKNOWN>                           CORE_DXE
11C 00000010 B - -  2  5 <UNKNOWN>                           CORE_DXE
11D 00000010 ? - -  -  - AMI PS/2 Driver                     CORE_DXE
11E 00000001 ? - -  -  - AMI IDE BUS Driver                  CORE_DXE
```

10.1.13 dumpacpi

Dumps ACPI1.0b, ACPI2.0, or ACPI3.0 Table in EFI Shell Environment.

Usage:

```
DumpACPI [-d] [-v] [-p] [-b]
```

-d	Dumps ACPI Table Raw Data.
-v	Dumps ACPI Table Verbose Data.
-s	Dumps ACPI Table with signature being <SIGN>. The signature should be defined value in ACPI spec. One exception is RSDP, please use RSDP instead of 'RSD PTR '.
-p	Dumps the parsed AML Code.
-b	Displays one screen at a time.

10.1.14 dumpaml

Dumps ACPI1.0b, ACPI2.0, or ACPI3.0 AML in EFI Shell Environment.

Usage:

```
DumpAML [-b] <AML file>
DumpAML <AML file> -e <AML Method Name> [<Argument>...]
```

-b	Displays one screen at a time.
-e	Executes AML method.

<AML Method Name> format: \AAAA.BBBB.CCCC.

<Argument> format: memory content in string. (eg. 34120000 means 0x1234)

10.1.15 echo

Controls batch file command echoing or displays a message.

```
ECHO [-on|-off]
ECHO [message]
```

`-on`	Enables echo when executing batch file commands
`-off`	Disables echo when executing batch file commands
`message`	Displays a message string

1. `Echo -off` disables the echo feature when executing batch file commands. This command is not like the MS-DOS echo command.
2. `Echo` without a parameter shows the current echo setting.

> Examples:

- To display the current echo setting:

```
fs0:\> echo
Echo is off
```

- To enable command echoing:

```
fs0:\> echo -on
```

- To disable command echoing:

```
fs0:\> echo -off
```

- To execute HelloWorld.nsh batch file and echo commands when executing:

```
fs0:\> HelloWorld.nsh
+HelloWorld.nsh> echo Hello World
Hello World
```

- To display a message string of 'Hello World':

```
fs0:\> echo Hello World
Hello World
```

10.1.16 exit

Exits the EFI Shell environment and returns control to the parent process. This command allows to exit the EFI shell and boot the next or first boot device in the boot list.

10.1.17 for

Executes one or more commands for each item in a set of items.

```
              FOR %indexvar IN set
      command [arguments]
      [command [arguments]]       ...
              ENDFOR
              FOR %indexvar RUN (start end[ step])
      command [arguments]
      [command [arguments]]      ...
              ENDFOR
```

%indexvar	Variable name used to index a set
set	Set to be searched
command [arguments]	Command to be executed with optional arguments

1. The FOR command is only available in batch script files.
2. FOR shall be matched with ENDFOR.
3. Start and end can be any integer. Up to 6 digits allowed.
4. Step can be any integer but zero. Up to 6 digits allowed.
5. step is optional, if step is not specified, step will be automatically determined as below:
   if start <= end, then step = 1
   if start > end, then step = -1

> Examples:

```
#
#  Sample for loop type contents of all *.txt files
#
for %a in *.txt
              type %a
              echo ===== %a done =====
endfor
#
#  To repeat operations, supporting multiple loop:
#
              for %a in 1 2 3 4 5 6 7 8 9
               for %b in a b c d e f g h i j k l m n o p q r s t u v w x y z
                alias %a a%a
                alias %b %b%a
              endfor
endfor
              for %a run (1 3)
               echo %a
              endfor
Output:
1
2
3
              for %a run (3 1)
               echo %a
              endfor
Output:
3
2
1
```

10.1.18 goto

Forces batch file execution to unconditionally jump to specified location.

```
GOTO label
```

`label` Specifies a location in batch file

1. The `GOTO` command is only available in batch script files.
2. Execution of batch file will jump to the line immediately following the specified label name.
3. `GOTO` cannot jump from outside into a FOR cycle block.

> Example:

```
            #
#  Example script for "goto" command
#
goto Done
...
:Done
cleanup.nsh
```

10.1.19 help

Displays the EFI Shell command list or verbose help for specific commands.

```
HELP [cmd | pattern]
```

cmd	Shell command name
pattern	Wildmatch pattern

1. 'cmd -?' also displays the verbose help of cmd, the same as 'help cmd'.
2. If the specified command has no verbose help, its line help will be displayed instead.

> Examples:

- To display the EFI Shell command list and break after one screen:

```
Shell> help -b
```

?	Displays the EFI Shell command list or verbose command help
alias	Displays, creates, or deletes aliases in the EFI Shell
attrib	Displays or changes the attributes of files or directories
cd	Displays or changes the current directory
cls	Clears the standard output with an optional background color
connect	Connects one or more EFI drivers to a device
copy	Copies one or more files or directories to another location

...

- To display help information for the ls shell command:

```
Shell> help ls
Shell> ? ls
Shell> ls -?
```

- To display the list of commands starting with the character 'p'

```
Shell> help p*
pause          Prints a message and waits for keyboard input
pci
```

10.1.20 if

Executes one or more commands in specified conditions.

```
IF [NOT] EXIST file THEN
    command [arguments]
[ELSE
    command [arguments]]
ENDIF
IF [NOT] string1 == string2 THEN
    command [arguments]
    [command [arguments]]      ...
[ELSE
    command [arguments]
    [command [arguments]]      ...]
ENDIF
```

EXIST file	TRUE if file exists in the directory
string1 == string2	TRUE if the two stings are same

1. The IF command is only available in batch script files.
2. If condition is TRUE, commands between IF and ELSE will be executed.
3. If condition is FALSE but keyword 'NOT' is not prefixed, commands between ELSE and ENDIF will also be executed.

> Example:

```
            #
            #  Example script for "if" command
#
if exist fs0:\myscript.sc then
myscript myarg1 myarg2
endif
if %myvar% == runboth then
myscript1
myscript2
endif
```

10.1.21 ifconfig

Ifconfig© Intel Corporation 2006 modifies the default IP address of UEFI network stack.

- To list the current address:

```
IfConfig -l [Name]
```

Shows the configuration for all or the interface

- To set the default address use:

```
IfConfig -s <Name> dhcp [perment]
```

Uses the EFI_DHCP4_PROTOCOL to request address dynamically

```
IfConfig -s <Name> <static> <IP> <Mask> <Gateway> [perment]
```

Uses the static IP4 address configuration

`perment` is optional. If present, the configuration survives the network stack reload. Otherwise, it is for this time only.

- To clear the current address:

```
IfConfig -c [Name]
```

Clears the configuration for all or the interface. Although the configuration is cleared, the network stack will fall back to the DHCP as default.

- Other:

```
IfConfig -?
```

Shows this help message.

> Example:

```
IfConfig -s eth0 dhcp
IfConfig -l eth0
IfConfig -s eth0 static 192.168.0.5 255.255.255.0 192.168.0.1 perment
```

The “Network stack”must be enabled in the Advanced menu to have this command available.

10.1.22 kdiag

Performs board diagnostics. Available ONLY if ordered.

10.1.23 kflash

Kontron SPI flasher

Usage:

```
kflash [-ver] [ -p|-i|-v|-s|-h|-? ]  [-f] [-r] [file]
```

- Operation mode
  - `-ver` Displays current BIOS ID
  - `-p` Programs flash
  - `-i` Shows information string and check CRC
  - `-v` Verifies flashed image
  - `-s` Saves current ROM image to file
  - `-c` Clones flash content to second flash (Only in RESCUE mode)
  - `-h` Shows this help
- Options
  - `-f` Forces write
- Expert options: Not recommended for standard use
  - `-r` Raw image mode (.bin, .rom)
  - `-e` Erases all flash without preserving Ethernet area
  - `-sp` Setup preserve NVRAM settings

10.1.24 kmac

Kontron MAC Address utility
Usage:

```
kmac [-h|-v|-r|-dump] [-w value] [-save|-load [filename]] [-prog [0|1]]
kmac [-lan [0|1|2]] [read|write value]
```

- Operation mode

`-h`	Shows this help
`-v`	Displays the versions of the i82599 EEPROM
`-r`	Shows all MAC Addresses of the board
`-w value`	Updates all MAC Addresses by auto-increment ETH0 = i82599 LAN0 = value+1 ETH1 = i82599 LAN1 = value+2 ETH2 = i82579 LAN = value
`-lan [0|1|2] [read|write value]`	Reads or writes MAC Address specified by LAN number 0 = i82599 LAN0, 1 = i82599 LAN1, 2 = i82579 value is a 6-bytes hexa number prefixed with "0x"
`-prog [0|1|2]`	Programs i82599 EEPROM with image specified by number 0 = channels 0 and 1 in 1000BASE-BX/KX 1 = channel 0 in 10GBASE-KR/SFI, channel 1 in 1000BASE-BX/KX 2 = channels 0 and 1 in 10GBASE-KR/SFI
`-dump`	Dumps the first 1024 words of the i82599 EEPROM
`-save`	Saves content of i82599 EEPROM into a binary file
`-load`	Loads content of i82599 EEPROM from a binary file
`-stat`	Displays MAC link status information

> Example:

```
Shell> kmac -r

MAC Address LAN ETH0 (Intel 82599) = 00:00:DE:40:41:76
MAC Address LAN ETH1 (Intel 82599) = 00:00:DE:40:41:77
MAC Address LAN ETH2 (Intel 82579) = 00:00:DE:40:41:75
```

10.1.25 kpld

Kontron PLD Command

Usage:

```
kpld [-h|-?] [-b] [-v] [-m] [-r Offset] [-w Offset Value]
kpld -i2cr busNum Add Offset Type [count]
kpld -i2cw busNum Add Offset Type Data [count]
```

- Operation mode
  - `-h|-?` Shows this help
  - `-v` Shows CPLD revision
  - `-m` Boot Flash information
  - `-r` Reads CPLD register
    -> kpld -r Offset
  - `-w` Writes CPLD register
    -> kpld -w Offset Value
  - `-i2cr` Reads Access to I2C bus
    -> kpld -i2cr busNum Add Offset Type [count]
  - `-i2cw` Writes Access to I2C bus
    -> kpld -i2cw busNum Add Offset Type Data [count]

10.1.26 ksata

Kontron SATA Configurator

Usage:

```
ksata [-b|-h|-?] [-p <on|off> <num_port> [-f]]
```

- Operation mode:
  - `-b` enable page break
  - `-h|-?` Show this help
  - `-p` program Early Power-Down or Write-Protect mode on SATA device
    Argument List:
    - `on` Power-Down mode
    - `off` Write-Protect mode
    - `num_port` SATA port number on which SATA device is plugged on
      (4 = on-board SSD or FDM-SATA device depending on the board)
  - `-f` force selected mode on SATA device

> **Example:**

```
VX304x> ksata -p off 4 -f
Port 4: GLS85LS1032A CS 32GBN A101D3
Program Write-Protect mode (FORCED) to SATA Port 4...OK
```

This command is not compatible with all SATA devices and must be used with caution. On VX3042 & VX3044 boards, only the onboard SSD device on SATA Port 4 has been validated along with FDM-SATA equiped with Greenliant Model.

If `-f` parameter is not added to the command, only SATA Port 4 and SATA Port 5 are allowed to be programmed. By default, the Write-Protect Mode is programmed on such devices supporting the Early Power Down feature but for VX3042 & VX3044 boards onboard SSD device, the Write-Protect is only enabled by switching on hardware switch SW1[5]. See CA.DT.A98 document for further information.

10.1.27 ktemp

Usage:

```
ktemp [ -h|-? ]
```

- Operation mode
  - -h Shows this help
  - -p Prints PCH temperature

> Example:

```
VX304x> ktemp -p
Thermal Characteristic:
   TM1(TCC) is supported AND enabled.
   TM2 is NOT enabled.
=======================================================

+------------------------+-------------+
| CPU Temperature        |    41 C     |
+------------------------+-------------+
| PKG Temperature        |    41 C     |
+------------------------+-------------+
| PCH Temperature        |    43 C     |
+------------------------+-------------+

+------------------------+-------------+
| PKG Power              |    10563 mW     |
+------------------------+-------------+
| Core0 Power            |    4540 mW     |
+------------------------+-------------+
```

10.1.28 kvpd

Kontron VPD Information: displays Vital Product Information

Usage:

```
kvpd [ -p|-m|-h|-? ]
```

- Operation mode

  `-p` Displays VPD information

  `-m` Modifies or enters VPD information (Rescue Only)

  `-h|-?` Shows this help

> Example:

```
Shell> kvpd -p

================ BOARD CONFIGURATION ==============

             Order Code     : PROTO-VX3040E-LOT1-A
             EC Level       : EC10002
             Serial Number  : 181211103006
             Variant        : 0000910480204000
             Check Sum      : 97DC8531

================ MAC ADDRESS ==============

             LAN ETH0: 00:00:DE:40:41:76
             LAN ETH1: 00:00:DE:40:41:77
             LAN ETH2: 00:00:DE:40:41:75
```

10.1.29 kvpx

Kontron VPX Configurator

Usage:

```
kvpx [-b|-h|-?] [-plx_eeprom [parameter]] [filename]
```

`-b`:	Enables page break
`-h|-?`:	Shows this help
`-plx_eeprom`:	Manages PCIe switch Serial EEPROM

Parameter list:

`prog`	Programs PCIe switch serial EEPROM
`dump`	Dumps PCIe switch serial EEPROM
`conf`	Display PCIe switch configuration
`ver`	Display version of PCIe switch serial EEPROM

Options:

`filename`:	Custom configuration filename in binary format
or	content of EEPROM filename in binary format

> Example:

```
VX304x> kvpx -plx_eeprom prog
Program EEPROM ID13200 for PLX Silicon Revision = BA
Write @0x0000 = 0x5A002406
Write @0x0004 = 0xA7000001
Write @0x0008 = 0x0501FF02
Write @0x000C = 0x03318E00
Write @0x0010 = 0xFF02302E
Write @0x0014 = 0x8E00FF02
Write @0x0018 = 0x0331AE00
Write @0x001C = 0xFF02302E
Write @0x0020 = 0xAE00FF02
Write @0x0024 = 0x0331CE00
Write @0x0028 = 0xFF02302E
Write @0x002C = 0xCE00FF02
Write @0x0030 = 0x0331EE00
Write @0x0034 = 0xFF02302E
Write @0x0038 = 0xEE00FF22
Write @0x003C = 0x03318E00
Write @0x0040 = 0xFF22302E
Write @0x0044 = 0x8E00FF22
Write @0x0048 = 0x0331AE00
Write @0x004C = 0xFF22302E
Write @0x0050 = 0xAE00FF22
Write @0x0054 = 0x0331CE00
Write @0x0058 = 0xFF22302E
```

```
...
Write @0x05FC = 0x00C011E7
Write @0x0600 = 0x00000040
Write @0x0604 = 0x3AE00800
Write @0x0608 = 0x00FC0FE3
Write @0x060C = 0x00000004
Write @0x0610 = 0x8C060000
Write @0x0614 = 0x000326E0
Write @0x0618 = 0x03000000
Write @0x061C = 0x1EE00000
Write @0x0620 = 0x00001EE4
Write @0x0624 = 0x00000000
Write @0x0628 = 0xFFFFFFFF
Write @0x062C = 0xFFFFFFFF
Write @0x7000 = 0x4B534120
Write @0x7004 = 0x56505820
Write @0x7008 = 0x45455052
Write @0x700C = 0x4F4D0000
Write @0x7010 = 0x33900019
Write @0x7014 = 0x05DC0624
Write @0x7018 = 0x058C05AA
Write @0x701C = 0x05DAB535
Writing Backplane PCI-E Switch serial EEPROM OK
WARNING: reset the system to load the new PCI-E Switch Serial EEPROM image.

VX304x> kvpx -b -plx_eeprom dump
@0x0000 = 0x5A002406
@0x0004 = 0xA7000001
@0x0008 = 0x0501FF02
@0x000C = 0x03318E00
@0x0010 = 0xFF02302E
@0x0014 = 0x8E00FF02
@0x0018 = 0x0331AE00
@0x001C = 0xFF02302E
@0x0020 = 0xAE00FF02
@0x0024 = 0x0331CE00
@0x0028 = 0xFF02302E
@0x002C = 0xCE00FF02
@0x0030 = 0x0331EE00
@0x0034 = 0xFF02302E
@0x0038 = 0xEE00FF22
@0x003C = 0x03318E00
@0x0040 = 0xFF22302E
@0x0044 = 0x8E00FF22
@0x0048 = 0x0331AE00
@0x004C = 0xFF22302E
@0x0050 = 0xAE00FF22
@0x0054 = 0x0331CE00
@0x0058 = 0xFF22302E
@0x005C = 0xCE00FF22
```

```
...
Press ENTER to continue, ´q´ to exit:
VX304x> kvpx -plx_eeprom conf

=========================================================================
                    User Configuration      |   EEPROM Configuration
-------------------------------------------------------------------------
  Status Mode :     Transparent             |   Transparent
-------------------------------------------------------------------------
  Link Width  :     x8                      |   x8
-------------------------------------------------------------------------
  Link Speed  :     8.0 GT/s (Gen3)         |   8.0 GT/s (Gen3)
-------------------------------------------------------------------------
  EEPROM CRC  :                             |   5912
=========================================================================
PEX8725 Silicon Revision(08h):       BA
-------------------------------------------------------------------------
                            Port 9 Configuration
-------------------------------------------------------------------------
PEX8725 Link Capabilities(74h):      09796083
PEX8725 Link Status(78h):            00000001
PEX8725 Maximum Link Speeds(3:0):    8.0 GT/s (Gen3)
PEX8725 Maximum Link Width(9:4):     x8
PEX8725 Current Link Speed(3:0):     2.5 GT/s (Gen1)
PEX8725 Negotiated Link Width(9:4):  x0 (Link Down)
PEX8725 Current Status Mode:         Transparent
```

In the "`kvpx -plx_eeprom conf`" command, the "User Configuration" column reflects both the status of board microswitches (SW2.3 for VPX PCI-E speed limit and SW3. [3:4] for VPX PCI-E Port Size) and also the BIOS setup configuration for VPX EEPROM (see section 5.6.6 page 31) while the "EEPROM Configuration" column reflects the **current** VPX configuration (loaded in the EEPROM non-volatile device of the PCI-E switch).

The "`Port 9 Configuration`" reflects the actual configuration on the PLX downstream port connected to the VPX Backplane for a PCI-E Port Size set to x8.

If the PCI-E Port Size is set to 2x4, then "`Port 10 Configuration`" will be displayed also, and if the PCI-E Port size is set to 4x2, then "`Port 9,10,11 and 12 Configurations`" will be displayed.

The `Maximum Link Speeds` and `Maximum Link Width` indicate the capabilities of the PCI-E link on VPX backplane.

The `Current Link Speed` and `Negociated Link Width` indicate respectively the actual speed and width of the PCI-E link on VPX backplane.

The `Current Status Mode` indicates if the PLX Bridge is in Transparent or Non-Transparent Mode.

If `Transparent` is displayed, then the board will see all VPX boards connected on the VPX backplane.

10.1.30 ls

Displays a list of files and subdirectories in a directory.

```
LS [-b] [-r] [-a[attrib]] [file]
```

-b	Displays one screen at a time
-r	Displays recursively (including subdirectories)
-a	Displays files with attributes of type attrib
attrib	File attribute list:
a	Archive
s	System
h	Hidden
r	Read-only
d	Directory
file	Name of file or directory (wildcards are permitted)

1. Files and directories with the system and hidden attributes are not displayed unless the 's' and 'h' attributes are specified.

> Examples:

- To hide files by adding the hidden and system attributes:

```
fs0:\> attrib +h +s *.efi
    ASH  fs0:\IsaBus.efi
    ASH  fs0:\IsaSerial.efi
```

- To display all files in the current directory:

```
fs0:\> ls
  Directory of: fs0:\
    06/18/01  09:32p                 153  for.nsh
    06/18/01  01:02p <DIR>           512  efi
    06/18/01  01:02p <DIR>           512  test1
    06/18/01  01:02p <DIR>           512  test2
    06/18/01  08:04p                  29  temp.txt
    06/18/01  08:05p <DIR>           512  test
    01/28/01  08:24p       r          29  readme.txt
            3 File(s)         211 bytes
            4 Dir(s)
```

- To display all files in the current directory:

```
fs0:\> ls -a
  Directory of: fs0:\
    06/18/01  09:32p                  153  for.nsh
    06/18/01  01:02p <DIR>            512  efi
    06/18/01  01:02p <DIR>            512  test1
    06/18/01  01:02p <DIR>            512  test2
    06/18/01  10:59p               28,739  IsaBus.efi
    06/18/01  10:59p               32,838  IsaSerial.efi
    06/18/01  08:04p                   29  temp.txt
    06/18/01  08:05p <DIR>            512  test
    01/28/01  08:24p       r           29  readme.txt
            5 File(s)      61,788 bytes
            4 Dir(s)
```

- To display all read-only files in the current directory:

```
fs0:\> ls -ar
 Directory of: fs0:\
   06/18/01  11:14p       r           29  readme.txt
           1 File(s)          29 bytes
           0 Dir(s)
```

- To display the file '`isabus.efi`' with the system attribute:

```
 fs0:\> ls -as isabus.efi
 Directory of: fs0:\
   06/18/01  10:59p               28,739  IsaBus.efi
           1 File(s)      28,739 bytes
           0 Dir(s)
```

- To display all files in the `fs0:\efi` directory recursively:

```
 fs0:\> ls -r -a efi
```

- To display all files with the '`*.efi`' extension recursively one screen at a time:

```
fs0:\> ls -b -r -a *.efi
```

10.1.31 map

Displays or defines mappings between user defined names and device handles.

```
MAP [-d <sname>]
MAP [[-r][-v][-c][-f][-t <type[,type...]>][sname]]
MAP [sname handle | mapname]
```

-d	Deletes a mapping
-r	Resets to default mappings
-v	Displays verbose mapping information
sname	User defined mapping name (wildcards are permitted)
handle	The number of handle, which is same as dumped from 'dh' command
-c	Displays the consistent mapping name
-f	Displays the normal mapping name(not consistent mapping)
-t	Displays the device mapping name according to the device type: fp Floppy hd Hard Disk cd CD-ROM Types can be combined by putting a comma between two types. Spaces are not allowed between types.
mapname	Mapped name for the device followed by a postfix ':'

1. The consistent mapping is persistent across the mapping reset and the system reboot.
2. Only characters and numbers are allowed inside of sname.
3. Redirection is not allowed when running map because we do not know the file system before mapping is done.
4. Output redirection is not supported for 'map -r' usage.

> Examples:

- To reset the mapping table to the default mappings:

```
Shell> map -r
Device mapping table
  fs0  :Removable HardDisk - Alias hd29b0b0b blk0
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk0 :Removable HardDisk - Alias hd29b0b0b fs0
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk1 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0xed32b4ef,0x800,0xfa000)
  blk2 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0xed32b4ef,0xfa800,0x9408000)
  blk3 :HardDisk - Alias (null)
PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(1,GPT,b2d5d098-ba66-4477-9ccd-5e2eaf41eed3,0x800,0x800)
  blk4 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(2,GPT,439ced9e-fdd2-408c-8bd4-b1ac8b8cd38b,0x1000,0xfa000)
  blk5 :HardDisk - Alias (null)
       PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(3,GPT,4e5b73d4-90b5-46ce-909a-3bf06bd43318,0xfb000,0x3aa7800)
```

```
  blk6 :BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)
  blk7 :BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)
  blk8 :Removable BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)
  hd29b0b0b :Removable HardDisk - Alias fs0 blk0
             PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

- To display all mappings in the device mapping table:

```
Shell> map
Device mapping table
  fs0  :Removable HardDisk - Alias hd29b0b0b blk0
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk0 :Removable HardDisk - Alias hd29b0b0b fs0
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk1 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0xed32b4ef,0x800,0xfa000)
  blk2 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0xed32b4ef,0xfa800,0x9408000)
  blk3 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(1,GPT,b2d5d098-ba66-4477-9ccd-5e2eaf41eed3,0x800,0x800)
  blk4 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(2,GPT,439ced9e-fdd2-408c-8bd4-b1ac8b8cd38b,0x1000,0xfa000)
  blk5 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(3,GPT,4e5b73d4-90b5-46ce-909a-3bf06bd43318,0xfb000,0x3aa7800)
  blk6 :BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)
  blk7 :BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)
  blk8 :Removable BlockDevice - Alias (null)
        PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)
  hd29b0b0b :Removable HardDisk - Alias fs0 blk0
             PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

- To display verbose mapping table information:

```
Shell> map -v
Device mapping table
  fs0   Consistent Name hd29b0b0b
        Other Name    blk0
        Handle        1A2: Fs DiskIo BlkIo
        Media Type    HardDisk
        Removable     YES
        Current Dir   \
         PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk0  Consistent Name hd29b0b0b
        Other Name    fs0
        Handle        1A2: Fs DiskIo BlkIo
        Media Type    HardDisk
        Removable     YES
        Current Dir   \
         PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

```
  blk1  Consistent Name (null)
        Other Name   (null)
        Handle       196: DiskIo BlkIo
        Media Type   HardDisk
        Removable    NO
        Current Dir  \
         PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0xed32b4ef,0x800,0xfa000)
  blk2  Consistent Name (null)
        Other Name   (null)
        Handle       197: DiskIo BlkIo
        Media Type   HardDisk
        Removable    NO
        Current Dir  \
         PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0xed32b4ef,0xfa800,0x9408000)
(...)
```

- To assign fs0 another name:

```
Shell > map floppy fs0:
Device mapping table
  floppy :Removable HardDisk - Alias hd29b0b0b fs0 blk0
           PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

- To display information about the mapped name:

```
Shell > map floppy
Device mapping table
  floppy :Removable HardDisk - Alias hd29b0b0b fs0 blk0
           PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

- To operate with the mapped name:

```
Shell > floppy:
floppy:\> ls
Directory of: floppy:\
(...)
```

- To delete a mapped name:

```
floppy:\> map -d floppy
Shell > map
Device mapping table
  fs0  :Removable HardDisk - Alias hd29b0b0b blk0
         PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk0 :Removable HardDisk - Alias hd29b0b0b fs0
         PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk1 :HardDisk - Alias (null)
         PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0xed32b4ef,0x800,0xfa000)
  blk2 :HardDisk - Alias (null)
         PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0xed32b4ef,0xfa800,0x9408000)
```

```
  blk3 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(1,GPT,b2d5d098-ba66-4477-9ccd-5e2eaf41eed3,0x800,0x800)
  blk4 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(2,GPT,439ced9e-fdd2-408c-8bd4-b1ac8b8cd38b,0x1000,0xfa000)
  blk5 :HardDisk - Alias (null)
        PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(3,GPT,4e5b73d4-90b5-46ce-909a-3bf06bd43318,0xfb000,0x3aa7800)
  blk6 :BlockDevice - Alias (null)
       PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)
  blk7 :BlockDevice - Alias (null)
       PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)
  blk8 :Removable BlockDevice - Alias (null)
       PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)
  hd29b0b0b :Removable HardDisk - Alias fs0 blk0
             PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
```

- To display all the mapped names starting with ´b´:

```
Shell> map b*
Device mapping table
  blk0   :Removable HardDisk - Alias hd29b0b0b fs0 floppy
          PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)/HD(1,MBR,0x000b9400,0x38,0x7ce10)
  blk1   :HardDisk - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0xed32b4ef,0x800,0xfa000)
  blk2   :HardDisk - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0xed32b4ef,0xfa800,0x9408000)
  blk3   :HardDisk - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(1,GPT,b2d5d098-ba66-4477-9ccd-5e2eaf41eed3,0x800,0x800)
  blk4   :HardDisk - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(2,GPT,439ced9e-fdd2-408c-8bd4-b1ac8b8cd38b,0x1000,0xfa000)
  blk5   :HardDisk - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)/HD(3,GPT,4e5b73d4-90b5-46ce-909a-3bf06bd43318,0xfb000,0x3aa7800)
  blk6   :BlockDevice - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)
  blk7   :BlockDevice - Alias (null)
          PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x4,0x0)
  blk8   :Removable BlockDevice - Alias (null)
          PciRoot(0x0)/Pci(0x1d,0x0)/USB(0x1,0x0)/USB(0x1,0x0)
```

10.1.32 mem

Displays the contents of system or device memory.

```
MEM [-b] [Address] [Size] [-MMIO]
```

-b	Displays one screen at a time
address	Starting address in hexadecimal format
size	Number of bytes to display in hexadecimal format
-MMIO	Forces address cycles to the PCI bus

1. All units are in hexadecimal format.
2. Address must be aligned on an even processor address boundary.
3. If the 'address' parameter is not specified, DMEM will display the all system table pointer entries by default.

> Examples:

- To display the EFI system table pointer entries:

```
Shell> mem
 Memory Address 000000007ADB7F18 200 Bytes
 7ADB7F18: 49 42 49 20 53 59 53 54-00 00 02 00 78 00 00 00  *IBI SYST....x...*
 7ADB7F28: 51 E1 C4 FF 00 00 00 00-00 B6 59 7A 00 00 00 00  *Q.........Yz....*
 7ADB7F38: 7B 02 04 00 00 00 00 00-18 EE AF 01 00 00 00 00  *{...............*
 7ADB7F48: F0 9A 1A 12 00 00 00 00-18 EE AF 01 00 00 00 00  *................*
 7ADB7F58: C0 9B 1A 12 00 00 00 00-18 EE AF 01 00 00 00 00  *................*
 7ADB7F68: A0 EB 59 7A 00 00 00 00-18 7E DB 7A 00 00 00 00  *..Yz.....z....*
 7ADB7F78: 40 D2 59 7A 00 00 00 00-06 00 00 00 00 00 00 00  *@.Yz............*
 7ADB7F88: 18 5E DB 7A 00 00 00 00-70 74 61 6C 98 00 00 00  *.^.z....ptal....*
 7ADB7F98: 7A 85 16 BB 02 1A 70 DB-64 75 FC 1F 63 C5 DE 0B  *z.....p.du..c...*
 7ADB7FA8: 6B C6 2B 63 56 7E 6B 5A-69 46 2C 40 DD 98 F3 E0  *k.+cV.kZiF,@....*
 7ADB7FB8: F4 41 B6 4E C3 BA 08 D1-36 6D 03 05 CF E8 1D 0C  *.A.N....6m......*
 7ADB7FC8: D7 37 16 91 DD 4B 10 45-4C FF 38 3D 01 B8 87 2A  *.7...K.EL.8=...**
 7ADB7FD8: E6 21 D6 6B 02 89 8A BD-FE ED 76 FA 3C A6 67 3D  *.!.k......v.<.g=*
 7ADB7FE8: 97 B7 7C 7F 6B B1 4C 9E-ED 50 D2 FC 75 9B 34 3E  *....k.L..P..u.4>*
 7ADB7FF8: 96 5E 4F 60 BE AD 1A 81-00 00 00 00 00 00 00 00  *.^O`............*
 7ADB8008: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8018: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8028: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8038: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8048: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8058: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8068: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8078: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8088: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB8098: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB80A8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB80B8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
 7ADB80C8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
```

```
7ADB80D8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
7ADB80E8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
7ADB80F8: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
7ADB8108: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*

Valid EFI Header at Address 000000007ADB7F18
----------------------------------------
System: Table Structure size 00000078 revision 00020000
ConIn (01AFEE18) ConOut (01AFEE18) StdErr (01AFEE18)
Runtime Services     000000007ADB7E18
Boot Services        000000007A59D240
ACPI 2.0 Table       000000007AFFFF98
SMBIOS Table         00000000000F0480
```

- To display memory contents from `7adbf18` with size of 16 bytes:

```
Shell> mem 7ADB7F18 16
Memory Address 000000007ADB7F18 10 Bytes
7ADB7F18: 49 42 49 20 53 59 53 54-00 00 02 00 78 00 00 00  *IBI SYST....x...*
```

- To display memory mapped IO contents from `7adbf18` with size of 16 bytes:

```
Shell> mem 7ADB7F18 16 -MMIO
 Memory Address 000000007ADB7F18 10 Bytes
7ADB7F18: 49 42 49 20 53 59 53 54-00 00 02 00 78 00 00 00  *IBI SYST....x...*
```

10.1.33 memmap

Displays the memory map maintained by the EFI environment.

```
MEMMAP [-b]
```

-b Displays one screen at a time

1. The EFI environment keeps track of all the physical memory in the system and how it is currently being used.
2. Total memory is the physical memory size, MemMapIO and MemPortIO not included.
3. Refer to the EFI specification for memory type definitions.

> Example:

- To display the system memory map:

```
VX304x> memmap

Type       Start            End              # Pages          Attributes
BS_code    0000000000000000-0000000000007FFF  0000000000000008 000000000000000F
available  0000000000008000-000000000007EFFF  0000000000000077 000000000000000F
BS_data    000000000007F000-000000000007FFFF  0000000000000001 000000000000000F
BS_code    0000000000080000-000000000009FFFF  0000000000000020 000000000000000F
available  0000000000100000-0000000000FFFFFF  0000000000000F00 000000000000000F
BS_data    0000000001000000-00000000016DFFFF  00000000000006E0 000000000000000F
BS_code    00000000016E0000-00000000016E0FFF  0000000000000001 000000000000000F
BS_data    00000000016E1000-00000000016EAFFF  000000000000000A 000000000000000F
(...)
ACPI_NVS   000000007AF42000-000000007AF90FFF  000000000000004F 000000000000000F
available  000000007AF91000-000000007AF94FFF  0000000000000004 000000000000000F
ACPI_NVS   000000007AF95000-000000007AFE7FFF  0000000000000053 000000000000000F
available  000000007AFE8000-000000007AFFCFFF  0000000000000015 000000000000000F
ACPI_recl  000000007AFFD000-000000007AFFFFFF  0000000000000003 000000000000000F
available  0000000100000000-00000001005FFFFF  0000000000000600 000000000000000F
reserved   00000000000A0000-00000000000FFFFF  0000000000000060 8000000000000000
reserved   000000007B000000-000000007F9FFFFF  0000000000004A00 8000000000000000
MemMapIO   00000000F8000000-00000000FBFFFFFF  0000000000004000 8000000000000000
MemMapIO   00000000FEC00000-00000000FEC00FFF  0000000000000001 8000000000000000
MemMapIO   00000000FED10000-00000000FED13FFF  0000000000000004 8000000000000000
MemMapIO   00000000FED18000-00000000FED19FFF  0000000000000002 8000000000000000
MemMapIO   00000000FED1C000-00000000FED1FFFF  0000000000000004 8000000000000000
MemMapIO   00000000FEE00000-00000000FEE00FFF  0000000000000001 8000000000000000
MemMapIO   00000000FFA00000-00000000FFBFFFFF  0000000000000200 8000000000000000
MemMapIO   00000000FFE00000-00000000FFFFFFFF  0000000000000200 8000000000000000
```

```
    reserved  :  20,131 Pages (82,456,576)
    LoaderCode:     212 Pages (868,352)
    LoaderData:     282 Pages (1,155,072)
    BS_code   :   1,512 Pages (6,193,152)
    BS_data   :  69,966 Pages (286,580,736)
    RT_code   :      94 Pages (385,024)
    RT_data   :      23 Pages (94,208)
    available : 431,903 Pages (1,769,074,688)
    ACPI_recl :       3 Pages (12,288)
    ACPI_NVS  :     162 Pages (663,552)
    MemMapIO  :  17,420 Pages (71,352,320)
  T otal Memory: 1,969 MB (2,065,027,072) Bytes
```

10.1.34 mm

Displays or modifies MEM/MMIO/IO/PCI/PCIE address space.

```
MM Address [Value] [-w 1|2|4|8] [-MEM | -MMIO | -IO | -PCI | -PCIE] [-n]
```

Address	Starting address
Value	The value to write
-MEM	Memory Address type
-MMIO	Memory Mapped IO Address type
-IO	IO Address type
-PCI	PCI Configuration Space Address type:
	Address format: 0x000000ssbbddffrr
	ss Segment
	bb Bus
	dd Device
	ff Function
	rr Register
-PCIE	PCIE Configuration Space Address type:
	Address format: 0x00000ssbbddffrrr
	ss Segment
	bb Bus
	dd Device
	ff Function
	rrr Register
-w	Unit size accessed in bytes:
	1 1 byte
	2 2 bytes
	4 4 bytes
	8 8 bytes
-n	Non-interactive mode

1. If the address type parameter is not specified, address type defaults to the 'MEM' type.
2. If the 'Value' parameter is specified, the '-n' option will be used automatically. In this case, this command will write the value to the specified address in non-interactive mode. If the 'Value' parameter is not specified, only the current contents in the address are displayed.
3. If the '-w' option is not specified, unit size defaults to 1 byte.
4. If the PCI address type is specified, the 'Address' parameter should follow the PCI Configuration Space Address format above. The 'PCI' command can be used to determine the address for a specified device. It is listed in the PCI configuration space dump information, in the following format: "[EFI 0x000000ssbbddffxx]".
5. If the PCIE address type is specified, the 'Address' parameter should follow the PCIE Configuration Space Address format above.
6. In interactive mode, type a hex value to modify, 'q' or '.' to exit. If the '-n' option is specified, it will run in non-interactive mode which supports batch file operation without user intervention.
7. Not all PCI configuration register locations are writable.
8. MM will only write the specified value. Read-modify-write operations are not supported.
9. The 'Address' parameter should be aligned on a boundary of the specified width.
10. Not all addresses are safe to access. Access to any improper address can bring unexpected results.

> Examples:

- To display or modify memory:

```
Address 0x1b07288, default width=1 byte:
    fs0:\> mm 1b07288
    MEM  0x0000000001B07288 : 0x6D >
    MEM  0x0000000001B07289 : 0x6D >
    MEM  0x0000000001B0728A : 0x61 > 80
    MEM  0x0000000001B0728B : 0x70 > q
    fs0:\> mm 1b07288
    MEM  0x0000000001B07288 : 0x6D >
    MEM  0x0000000001B07289 : 0x6D >
    MEM  0x0000000001B0728A : 0x80 >        *Modified
    MEM  0x0000000001B0728B : 0x70 > q
```

- To modify memory:

```
    Address 0x1b07288, width = 2 bytes:
    Shell> mm 1b07288 -w 2
    MEM  0x0000000001B07288 : 0x6D6D >
    MEM  0x0000000001B0728A : 0x7061 > 55aa
    MEM  0x0000000001B0728C : 0x358C > q
    Shell> mm 1b07288 -w 2
    MEM  0x0000000001B07288 : 0x6D6D >
    MEM  0x0000000001B0728A : 0x55AA >      *Modified
    MEM  0x0000000001B0728C : 0x358C > q
```

- To display IO space:

```
    Address 80h, width = 4 bytes:
    Shell> mm 80 -w 4 -IO
    IO  0x0000000000000080 : 0x000000FE >
    IO  0x0000000000000084 : 0x00FF5E6D > q
```

- To modify IO space using non-interactive mode:

```
    Shell> mm 80 52 -w 1 -IO
    Shell> mm 80 -w 1 -IO
    IO  0x0000000000000080 : 0x52 > FE    *Modified
    IO  0x0000000000000081 : 0xFF >
    IO  0x0000000000000082 : 0x00 >
    IO  0x0000000000000083 : 0x00 >
    IO  0x0000000000000084 : 0x6D >
    IO  0x0000000000000085 : 0x5E >
    IO  0x0000000000000086 : 0xFF >
    IO  0x0000000000000087 : 0x00 > q
```

- To display PCI configuration space, ss=00, bb=00, dd=00, ff=00, rr=00:

```
Shell> mm 0000000000 -PCI
PCI  0x0000000000000000 : 0x86 >
PCI  0x0000000000000001 : 0x80 >
PCI  0x0000000000000002 : 0x30 >
PCI  0x0000000000000003 : 0x11 >
PCI  0x0000000000000004 : 0x06 >
PCI  0x0000000000000005 : 0x00 > q
```

These contents can also be displayed by 'PCI 00 00 00'.

- To display PCIE configuration space, ss=00, bb=06, dd=00, ff=00, rrr=000:

```
Shell> mm 00060000000 -PCIE
PCIE  0x0000000060000000 : 0xAB >
PCIE  0x0000000060000001 : 0x11 >
PCIE  0x0000000060000002 : 0x61 >
PCIE  0x0000000060000003 : 0x43 >
PCIE  0x0000000060000004 : 0x00 > q
```

10.1.35 pause

Prints a message and waits for keyboard input.

```
PAUSE [-q]
```

`-q` Does not display notification message

1. The PAUSE command is only available in batch script files.
2. The prompt message is "`Enter ´q´ to quit, any other key to continue`".

> Examples:

- To pause the system after displaying the date and time:

```
fs0:\> type pause.nsh
File: fs0:\pause.nsh, Size 204
#
#  Example script for ´pause´ command
#
echo pause.nsh begin..
date
time
pause
echo pause.nsh done.
```

- To execute the script with `echo on`:

```
+pause.nsh> echo pause.nsh begin..
pause.nsh begin..
+pause.nsh> date
06/19/2001
+pause.nsh> time
00:51:45
+pause.nsh> pause
Enter ´q´ to quit, any other key to continue:
+pause.nsh> echo pause.nsh done.
pause.nsh done.
fs0:\> pause.nsh
```

- To execute the script with `echo off`:

```
fs0:\> echo -off
fs0:\> pause.nsh
pause.nsh begin..
06/19/2001
00:52:50
Enter ´q´ to quit, any other key to continue: q
fs0:\>
```

10.1.36 pci

Displays PCI device list or PCI function configuration space.

```
PCI [Bus Dev [Func] [-s Seg] [-i]]
```

Bus	Bus number
Dev	Device number
Func	Function number
-s	Optional segment number specified
Seg	Segment number
-i	Information interpreted

1. If no parameters are specified all PCI devices will be listed.
2. If the Bus and Device number parameters are specified while the Function or Segment parameters are not, Function or Segment will be set as default value 0.
3. The '-i' option can be used to display verbose information for the specified PCI device. The PCI configuration space for the specified device will be dumped with a detailed interpretation.

> Examples on VX304x:

- To display all PCI devices in the system:

```
VX304x> pci
   Seg  Bus  Dev  Func
   ---  ---  ---  ----
    00   00   00    00 ==> Bridge Device - Host/PCI bridge
              Vendor 8086 Device 0154 Prog Interface 0
    00   00   01    00 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 0151 Prog Interface 0
    00   00   01    01 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 0155 Prog Interface 0
    00   00   02    00 ==> Display Controller - VGA/8514 controller
              Vendor 8086 Device 0166 Prog Interface 0
    00   00   14    00 ==> Serial Bus Controllers - USB
              Vendor 8086 Device 1E31 Prog Interface 30
    00   00   19    00 ==> Network Controller - Ethernet controller
              Vendor 8086 Device 1502 Prog Interface 0
    00   00   1A    00 ==> Serial Bus Controllers - USB
              Vendor 8086 Device 1E2D Prog Interface 20
    00   00   1D    00 ==> Serial Bus Controllers - USB
              Vendor 8086 Device 1E26 Prog Interface 20
    00   00   1F    00 ==> Bridge Device - PCI/ISA bridge
              Vendor 8086 Device 1E55 Prog Interface 0
    00   00   1F    02 ==> Mass Storage Controller - UNDEFINED
              Vendor 8086 Device 1E03 Prog Interface 1
    00   00   1F    03 ==> Serial Bus Controllers - System Management Bus
              Vendor 8086 Device 1E22 Prog Interface 0
```

```
    00   00   1F    06 ==> Data Acquisition & Signal Processing Controllers - 0t
             Vendor 8086 Device 1E24 Prog Interface 0
    00   01   00    00 ==> Network Controller - Ethernet controller
             Vendor 8086 Device 10FC Prog Interface 0
    00   01   00    01 ==> Network Controller - Ethernet controller
             Vendor 8086 Device 10FC Prog Interface 0
    00   02   00    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8725 Prog Interface 0
    00   02   00    01 ==> Base System Peripherals - Other system peripheral
             Vendor 10B5 Device 87D0 Prog Interface 0
    00   02   00    02 ==> Base System Peripherals - Other system peripheral
             Vendor 10B5 Device 87D0 Prog Interface 0
    00   02   00    03 ==> Base System Peripherals - Other system peripheral
             Vendor 10B5 Device 87D0 Prog Interface 0
    00   02   00    04 ==> Base System Peripherals - Other system peripheral
             Vendor 10B5 Device 87D0 Prog Interface 0
    00   03   00    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8725 Prog Interface 0
    00   03   08    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8725 Prog Interface 0
    00   03   09    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8725 Prog Interface 0
```

- To display the configuration space of Bus 3, Device 9, Function 0:

```
VX304x> pci 3 9 0 -i
  PCI Segment 00 Bus 03 Device 09 Func 00 [EFI 8000000000000000003DA4F000000090000]
  00000000: B5 10 25 87 07 00 10 00-BA 00 04 06 10 00 01 00  *..%.............*
  00000010: 00 00 00 00 00 00 00 00-03 06 06 00 F1 01 00 00  *................*
  00000020: F0 FF 00 00 F1 FF 01 00-00 00 00 00 00 00 00 00  *................*
  00000030: FF 00 00 00 40 00 00 00-00 00 00 00 05 01 10 00  *....@...........*

  00000040: 01 48 03 C8 08 00 00 00-05 68 86 01 00 00 00 00  *.H.......h......*
  00000050: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000060: 00 00 00 00 00 00 00 00-10 A4 62 01 01 80 00 00  *..........b.....*
  00000070: 10 08 09 00 83 60 79 09-00 00 01 00 DF 0C 48 01  *.....`y.......H.*
  00000080: 40 05 70 00 00 00 00 00-00 00 00 00 60 08 04 00  *@.p.........`...*
  00000090: 00 00 00 00 0E 01 00 00-03 00 01 00 00 00 00 00  *................*
  000000A0: 00 00 00 00 0D 00 00 00-B5 10 25 87 00 00 00 00  *..........%.....*
  000000B0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000C0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000D0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000E0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000F0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*

Vendor ID(0): 10B5                    Device ID(2): 8725
```

```
Command(4): 0007
  (00)I/O space access enabled:       1  (01)Memory space access enabled:    1
  (02)Behave as bus master:           1  (03)Monitor special cycle enabled:  0
  (04)Mem Write & Invalidate enabled: 0  (05)Palette snooping is enabled:    0
  (06)Assert PERR# when parity error: 0  (07)Do address/data stepping:       0
  (08)SERR# driver enabled:           0  (09)Fast back-to-back transact...:  0

Status(6): 0010
  (04)New Capabilities linked list:   1  (05)66MHz Capable:                  0
  (07)Fast Back-to-Back Capable:      0  (08)Master Data Parity Error:       0
  (09)DEVSEL timing:               Fast  (11)Signaled Target Abort:          0
  (12)Received Target Abort:          0  (13)Received Master Abort:          0
  (14)Signaled System Error:          0  (15)Detected Parity Error:          0

Revision ID(8):     BA                  BIST(0F):  Incapable
Cache Line Size(C): 10                  Latency Timer(D): 00
Header Type(0E):    01, Single function, P2P bridge
Class: Bridge Device - PCI/PCI bridge -
Base Address Registers(10):  (None)
No Expansion ROM(38)

(Bus Numbers)  Primary(18)     Secondary(19)   Subordinate(1A)
               ------------------------------------------------------
               03               06               06
Secondary Latency Timer(1B):       00

Secondary Status(1E):    0
  (04)New Capabilities linked list:   0  (05)66MHz Capable:                  0
  (07)Fast Back-to-Back Capable:      0  (08)Master Data Parity Error:       0
  (09)DEVSEL timing:               Fast  (11)Signaled Target Abort:          0
  (12)Received Target Abort:          0  (13)Received Master Abort:          0
  (14)Received System Error:          0  (15)Detected Parity Error:          0

Resource Type                          Base                    Limit
-------------------------------------------------------------------------
I/O(1C)                            00FFF000                00000FFF
Memory(20)                         FFF00000                000FFFFF
Prefetchable Memory(24)    00000000FFF00000         00000000000FFFFF

Capabilities Ptr(34):   40

Bridge Control(3E)     0010
  (00)Parity Error Response:          0  (01)SERR# Enable:                   0
  (02)ISA Enable:                     0  (03)VGA Enable:                    0
  (05)Master Abort Mode:              0  (06)Secondary Bus Reset:           0
  (07)Fast Back-to-Back Enable:       0  (08)Primary Discard Timer:       2^15
  (09)Secondary Discard Timer:     2^15  (10)Discard Timer Status:          0
  (11)Discard Timer SERR# Enable:     0

Interrupt Line(3C)      05             Interrupt Pin(3D):       01

Pci Express device capability structure:
CapID( 0):          10            NextCap Ptr( 1):    A4
```

```
Cap Register( 2):             0162
  Capability Version(3:0):          0x0002
  Device/PortType(7:4):             Downstream Port of PCI Express Switch
  Slot Implemented(8):              1
  Interrupt Message Number(13:9):   0x00000
Device Capabilities( 4):      00008001
  Max_Payload_Size Supported(2:0):          256 bytes
  Phantom Functions Supported(4:3):         0
  Extended Tag Field Supported(5):          5-bit Tag field supported
  Role-based Error Reporting(15):           1
Device Control( 8):           0810
  Correctable Error Reporting Enable(0):    0
  Non-Fatal Error Reporting Enable(1):      0
  Fatal Error Reporting Enable(2):          0
  Unsupported Request Reporting Enable(3):  0
  Enable Relaxed Ordering(4):               1
  Max_Payload_Size(7:5):                    128 bytes
  Extended Tag Field Enable(8):             0
  Phantom Functions Enable(9):              0
  Auxiliary (AUX) Power PM Enable(10):      0
  Enable No Snoop(11):                      1
  Max_Read_Request_Size(14:12):             128 bytes
Device Status( A):            0009
  Correctable Error Detected(0):            1
  Non-Fatal Error Detected(1):              0
  Fatal Error Detected(2):                  0
  Unsupported Request Detected(3):          1
  AUX Power Detected(4):                    0
  Transactions Pending(5):                  0
Link Capabilities( C):        09796083
  Supported Link Speeds(3:0):                   8.0 GT/s, 5.0 GT/s and 2.5 GT/s supported
  Maximum Link Width(9:4):                      x8
  Active State Power Management Support(11:10):       No ASPM Supported
  LOs Exit Latency(14:12):                      2us-4us
  L1 Exit Latency(17:15):                       32us-64us
  Clock Power Management(18):                   0
  Surprise Down Error Reporting Capable(19):          1
  Data Link Layer Link Active Reporting Capable(20):  1
  Link Bandwidth Notification Capability(21):         1
  Port Number(31:24):                           0x09
Link Control(10):             0000
  Active State Power Management Control(1:0):         Disabled
  Link Disable(4):                              0
  Common Clock Configuration(6):                0
  Extended Synch(7):                            0
  Enable Clock Power Management(8):             0
  Hardware Autonomous Width Disable(9):         0
```

```
  Link Bandwidth Management Interrupt Enable(10):     0
  Link Autonomous Bandwidth Interrupt Enable(11):     0
Link Status(12):              0001
  Current Link Speed(3:0):                            2.5 GT/s
  Negotiated Link Width(9:4):                         x0
  Link Training(11):                                  0
  Slot Clock Configuration(12):                       0
  Data Link Layer Link Active(13):                    0
  Link Bandwidth Management Status(14):               0
  Link Autonomous Bandwidth Status(15):               0
Slot Capabilities(14):        01480CDF
  Attention Button Present(0):                        1
  Power Controller Present(1):                        1
  MRL Sensor Present(2):                              1
  Attention Indicator Present(3):                     1
  Power Indicator Present(4):                         1
  Hot-Plug Surprise(5):                               0
  Hot-Plug Capable(6):                                1
  Slot Power Limit Value(14:7):                       0x19
  Slot Power Limit Scale(16:15):                      1.0x
  Electromechanical Interlock Present(17):            0
  No Command Completed Support(18):                   0
  Physical Slot Number(31:19):                        41
Slot Control(18):             0540
  Attention Button Pressed Enable(0):                 0
  Power Fault Detected Enable(1):                     0
  MRL Sensor Changed Enable(2):                       0
  Presence Detect Changed Enable(3):                  0
  Command Completed Interrupt Enable(4):              0
  Hot-Plug Interrupt Enable(5):                       0
  Attention Indicator Control(7:6):                   On
  Power Indicator Control(9:8):                       On
  Power Controller Control(10):                       Power Off
  Electromechanical Interlock Control(11):            0
  Data Link Layer State Changed Enable(12):           0
Slot Status(1A):              0070
  Attention Button Pressed(0):           0
  Power Fault Detected(1):               0
  MRL Sensor Changed(2):                 0
  Presence Detect Changed(3):            0
  Command Completed(4):                  1
  MRL Sensor State(5):                   MRL  Opened
  Presence Detect State(6):              Card Present in slot
       Electromechanical Interlock Status(7): Electromechanical Interlock Disengaged
  Data Link Layer State Changed(8):      0
Root Control(1C):             0000
Root Capabilities(1E):        0000
Root Status(20):              00000000
```

```
Start dumping PCIex extended configuration space (0x100 - 0xFFF).

  00000100: 03 00 41 FB 00 0E DF B5-10 00 87 BA 19 00 81 14  *..A.............*
  00000110: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000120: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000130: 00 00 00 00 00 00 00 00-04 00 C1 10 00 00 00 00  *................*
  00000140: 00 00 00 00 01 00 00 00-02 00 01 E0 11 0C 00 00  *................*
(...)
```

10.1.37 reconnect

Reserved - Not To Be Used

10.1.38 reset

Resets the system.

```
RESET [-w [string]]
RESET [-s [string]]
```

`-w`	Performs a warm reset
`-s`	Performs a shutdown
`string`	String to be passed to reset service

1. Reset will be guaranteed to reset the chipset as well as the processor when cold reset is called.
2. This command does not support output redirection.

10.1.39 set

Displays, creates, changes, or deletes EFI Shell environment variables.

```
SET [-v] [sname [value]]
SET [-d <sname>]
```

`-d`	Deletes the environment variable
`-v`	Volatile variable
`sname`	Environment variable name
`value`	Environment variable value

1. SET values are stored in EFI NVRAM and will be retained between boots unless the option `-v` is specified.

> Examples:

- To add an environment variable:

```
Shell> set DiagnosticPath fs0:\efi\diag;fs1:\efi\diag
```

- To display all environment variables:

```
Shell> set
               * path           : .
               diagnosticPath : fs0:\efi1.1\diag;fs1:\efi1.1\diag
```

- To delete an environment variable:

```
Shell> set -d diagnosticpath
Shell> set
               * path           : .
```

- To change an environment variable:

```
fs0:\> set src efi
fs0:\> set
               * path : .;fs0:\efi\tools;fs0:\efi\boot;fs0:\
                 src  : efi
fs0:\> set src efi1.1
fs0:\> set
               * path : .;fs0:\efi\tools;fs0:\efi\boot;fs0:\
                 src  : efi1.1
```

- To append an environment variable:

```
Shell> set
               * path           : .
Shell> set path %path%;fs0:\efi\tools;fs0:\efi\boot;fs0:\
Shell> set
               * path           : .;fs0:\efi\tools;fs0:\efi\boot;fs0:\
```

- To set a volatile variable that will disappear at the next boot:

```
Shell> set -v EFI_SOURCE c:\project\EFI1.1
Shell> set
               * path       : .;fs0:\efi\tools;fs0:\efi\boot;fs0:\
               * EFI_SOURCE : c:\project\EFI1.1
```

10.1.40 shift

Shifts batch file input parameter positions.

```
SHIFT
```

1. The SHIFT command is only available in batch script files.
2. Each time the SHIFT command is executed the parameters are shifted one position higher, giving you access to more than ten parameters.

> Examples:

- To execute a batch file named `MySript.nsh`:

```
fs0:\> MyScript.nsh X1 X2 X3 X4 X5 X6 X7 X8 X9 X10
```

The parameters available when `MyScript.nsh` initially begins execution will be set as follows:

%1 = X1
%2 = X2
%3 = X3
%4 = X4
%5 = X5
%6 = X6
%7 = X7
%8 = X8
%9 = X9

- To shift the parameters one position inside the batch file:

```
shift
```

The parameters available in `MyScript.nsh` are changed as follows:

%1 = X2
%2 = X3
%3 = X4
%4 = X5
%5 = X6
%6 = X7
%7 = X8
%8 = X9
%9 = X10

10.1.41 smbiosview

Displays SMBIOS information.

```
SMBIOSVIEW [-t SmbiosType]|[-h SmbiosHandle]|[-s]|[-a]
```

`-t`	Displays all structures of SmbiosType
`SmbiosType`	SMBIOS structure type
`-h`	Displays structure of SmbiosHandle
`SmbiosHandle`	SMBIOS structure unique 16-bit handle
`-s`	Displays statistics table
`-a`	Displays all information

1. The SmbiosType parameter supports the following types:
   - 0 - BIOS Information
   - 1 - System Information
   - 2 - Base Board Information
   - 4 - Processor Information
   - 7 - Cache Information
   - 11 - OEM Strings
   - 16 - Physical Memory Array
   - 17 - Memory Device
   - 18 - 32-bit Memory Error Information
   - 19 - Memory Array Mapped Address
   - 20 - Memory Device Mapped Address
   - 21 - Built-in Pointing Device
   - 22 - Portable Battery
   - 26 - Voltage Probe
   - 27 - Cooling Device
   - 28 - Temperature Probe
   - 29 - Electrical Current Probe
   - 32 - System Boot Information
   - 34 - Management Device
   - 35 - Management Device Component
   - 36 - Management Device Threshold Data
   - 39 - System Power Supply
2. The SmbiosHandle parameter can be specified in either decimal or hexadecimal format. Use the '0x' prefix format for hexadecimal values.

10.1.42 smbutil

Reserved - Not To Be Used

10.1.43 time

Displays or changes the current system time.

```
time [hh:mm[:ss]]
```

hh	Hour of time to set, range: 0 - 23
mm	Minute of time to set, range: 0 - 59
ss	Second of time to set, range: 0 - 59

1. Hour and minute are required to set the time.
2. If second is not specified, 0 will be used as default.

10.1.44 timezone

Displays or sets time zone information.

```
TIMEZONE [-s hh:mm | -l] [-b] [-f]
```

-s hh:mm	Sets time zone associated with hh:mm offset from GMT
-l	Displays list of all time zones
-b	Displays one screen at a time
-f	Displays full information for specified timezone

> Example:

```
VX304x> timezone -s +1:00

VX304x> timezone -f
GMT+01:00, Amsterdam, Berlin, Bern, Rome, Paris, West Central Africa
```

The current time is not modified by this command; it is only an information about the time zone displayed with the command time.

10.2 Environment Variables

EFI shell allows user to set environment variables.

Three environment variables are available on VX304x board to control the behavior of EFI shell as described hereafter.

10.2.1 Bootcmd

The environment variable "**bootcmd**" allows the end user to run automatically an EFI command at startup of the EFI shell without typing any command on the keyboard.

> Examples:

1. To set **bootcmd** to run the "**pci**" command on EFI shell:

```
VX304x> set bootcmd "pci"
```

2. To check if the **bootcmd** variable is set on EFI shell:

```
VX304x> set
bootcmd: pci
```

3. To clear the **bootcmd** variable on EFI shell:

```
VX304x> set -d bootcmd
```

10.2.2 StartupAuto

The environment variable "**StartupAuto**" allows user to run the EFI shell script file "**startup.nsh**" present for example on a USB Flash drive plugged on the board.

> Examples:

1. To set **StartupAuto** variable on EFI shell:

```
VX304x> set StartupAuto 1
```

2. To clear **StartupAuto** variable on EFI shell:

```
VX304x> set -d StartupAuto
```

10.2.3 StartupDelay

The environment variable "`StartupDelay`" allows user to set a timeout delay before running the EFI shell script file "startup.nsh" present for example on a USB Flash drive plugged on the board.

The value of "`StartupDelay`" is a number that represents a delay in seconds.

> Examples:

1. To set a 2 seconds delay in `StartupDelay` variable on EFI shell:

```
VX304x> set StartupDelay 2
```

2. To clear `StartupDelay` variable on EFI shell:

```
VX304x> set -d StartupDelay
```

By default, the startup delay before running the EFI shell script `startup.nsh` is equal to 5 seconds.

Chapter 11 - BIOS Versions Description

11.1 Recommendations and Known Limitations

1. Reserved Setup settings

All the settings that are not described in this documentation are reserved and should not be changed. Changing any of these settings may cause system dysfunction or failure.

2. After BIOS upgrades

   It is recommended to turn the system off and do a power-on after upgrading the BIOS with the EFI shell "`kflash`" command or another utility.

3. Display Port hot plug

   The BIOS does not support hot plug for Display Port. The user has to plug the Display Port device before switching the board on.

4. ACPI warnings under Linux OS

   Some ACPI warnings are logged under the Linux Fedora operating system using the "`dmesg`" utility. Those messages are not errors and should be ignored.

5. HEST and ERST ACPI tables are not supported

   Currently, the BIOS does not implement the Hardware Error Source Table (HEST) and the Error Record Serialization Table (ERST) in the ACPI atbles. So, the operating system cannot retrieve error information as the PCI-Express Advanced Error reporting (AER).

6. "`kflash`" command limitation

   The "`-i`", "`-v`" and "`-sp`" options of the "`kflash`" command are not operational.

7. Intel® vPro™

   Intel® vPro™ technology is a set of security and manageability capabilities built into the 3rd generation Intel® Core vPro™ processor family like Intel "Ivy Bridge".

   Currently, the BIOS supports only the Intel Virtualization feature as part of the Intel® vPro™ technology and this feature is disabled by default into the setup.

11.2 Known Problems Table

The following table lists the BIOS relative known problems.

11.2.1 How to use the table:

1. Get the BIOS ID associated to your board. Refer to Chapter 3 “Main Menu” page 4 of this document.
2. Check for a specific item in the table rows:
   2.1. A “x" (cross) in the BIOS ID column indicates this item applies to this BIOS release (problem is not solved).
   2.2. No “x" (cross) in the BIOS ID column indicates this item does not apply to this release (problem is fixed).
3. A full description associated to a specific problem is available in the next section.

Item	CRP	Description	BIOS ID					
			12355	13148	13205	13287	13346	
1	4099	Internal Graphic Display cannot be disabled and causes memory hole in MAIN memory area	X	X	X			
2	4194	Screen issue at start-up and loss of serial console with some VGA monitors				X		

11.2.2 Detailed description of the problems

Item # 1 Internal Graphic Display cannot be disabled and causes memory hole in MAIN memory area

Description: For vxWorks BSP VX3042 & VX3044, it is necessary to disable interne (IGFX) to avoid the following memory mapping holes (extract of memmap):

```
available 000000002100000-000000001FFFFFFF 000000000001DF00 000000000000000F
reserved  0000000020000000-00000000201FFFF  0000000000000200 000000000000000F
available 0000000020200000-00000000040003FFF0000000000001FE04 000000000000000F
reserved  0000000040004000-00000000040004FFF0000000000000001 000000000000000F
```

The issue is that, when IGD is disabled under Setup, the BIOS does not allow to boot Legacy devices like Ethernet network by PXE or SATA HDD.

For example with Ethernet PXE, the serial console displays following messages:

```
ERROR: Type:2; Severity:90; Class:3; Subclass:5; Operation: 1006
ERROR: Type:2; Severity:90; Class:3; Subclass:5; Operation: 1006
```

Workaround: None

Item # 2 Screen issue at start-up and loss of serial console with some monitors

Description: With BIOS ID13287, a screen issue at start-up with a loss of the serial console redirection occurs with some VGA monitors that are not "MultiSync".

Root Cause: When no EDID exist on VGA monitors, the BIOS selects three possible resolutions, i.e. 1024x768, 800x600 or 640x480 but in the new BIOS release, a 4th resolution has been added for graphical Text Mode resolution that is not compliant with some VGA monitors and serial redirection.

Workaround: Use a MultiSync monitor with both DisplayPort and VGA output.

11.3 BIOS ID12355 Release Notes

The identified problems relative to the BIOS release ID12355 are described in the section 11.2 “Known Problems Table” above.

Here are some of the Kontron specific features implemented in the release.

These following are accessible by setup:

- Serial Port Console Redirection on COM0 and/or COM1 - Section 5.8 page 34
- Ethernet LAN Configuration - Section 5.2 page 24
- VPX Configuration - Section 5.6 page 28
- USB keyboard configuration - Section 5.3 page 25
- UUID Configuration - Section 5.4 page 26
- Watchdog timer implementation at OS boot time - Section 5.9 page 35
- Vital Product Data display - Section 5.5 page 27

These following are accessible by Kontron EFI commands. Refer to chapter 10 page 55 for details:

- `kdiag`, Board diagnostics (feature available only if ordered, the version included in BIOS ID12104 is not fully implemented/tested)
- `kflash`, SPI flasher.
- `kmac`, GbeLan MAC address management.
- `kpld`, CPLD register and I2C device access
- `ktemp`, Board temperature display
- `kvpd`, Vital Product Data information
- `kvpx`, VPX configurator

11.4 BIOS ID13148 Release Notes

For BIOS upgrade from release less than ID13148, use the following procedure:

1. Before flashing, retrieve the MAC addresses of the Intel 82579 by using:

```
kmac -r
MAC Address LAN ETH2 (Intel 82579) = 00:00:DE:XX:XX:XX
```

2. It is mandatory to kflash this BIOS with the "`-e`" option in order to overwrite the Gigabit Ethernet EEPROM area in the system flash:

```
kflash -p -r -e VX3040_ID13148.bin
```

3. After flashing, restore the MAC address for the Intel 82579 with:

```
kmac -wf 0x0000DExxxxxx
```

The following lists the evolutions or enhancements relative to this BIOS release:

- Fix PLX EEPROM configuration according to user settings.
- Fix incompatibility between BIOS supporting PLX EEPROM programmation thru I2C and board without the hardware implementation.
- Fix PCI Interrupt routing for Legacy PMC behind PCIe Switch.
- Fix SMBIOS table reference to LM78 device that do not exist.
  Change LM78 device by Nuvoton device NCT7802.
- Fix cTDP configurable using microswitch SW3[2:1].
- Add new feature to select SATA speed for each port in IDE/AHCI mode.
- Add new version of the VPX EEPROM management using I2C bus instead of PCI access for VPX configuration in case of EEPROM corruption.
- Add new "`kvpx -plx_eeprom conf`" command to display current VPX configuration.
- Add new Intel NVM 0.F4 image file for Gigabit Ethernet Intel 82579 device.
- Fix SW2.2 microswitch function for BIOS_FailSafe feature.
- Fix in Setup default value for PEG Swing Control to REDUCED.
- Fix Issue on SYSCON not detected when forced in Setup for PCI alignment for VxFabric™.
- Change word "`DIMM`" by "`CHANNEL`" in EFI tool `SmbUtil` to avoid confusion.
- Fix a bug in displaying IGFx Frequency in Setup menu System Agent Configuration -> Graphics Configuration (always display 0 MHz).
- Add new feature for POST MEMORY Test depending on board microswitches:

SPD Debug Mode	Factory Test Mode	Action
OFF	OFF	No PBITs run OFF
OFF	ON	PBITs run on Channel A & B
ON	OFF	PBITs run on Channel A
ON	ON	PBITs run on Channel B

1. The PBITs Memory test run on Channel A & B ONLY if board supports Dual Channel mode.
2. The PBITs Memory test run on Channel A ONLY if board supports Dual Channel mode.
3. If board supports ONLY Single Channel Mode, then the PBITs Memory test run on Channel B.

- Fix bug on boot if WRITE-PROTECT is set on SPI System Flash and if VX3042 & VX3044 is switched between SYSCON and PERIPHERAL.
- Fix bug on kmac display XMC-401 MAC ADDRESS instead of onboard 82599 MAC ADDRESS.
- Add new Microcode Update for IVB mask E-1 / L-1 / K-0.
- Do not spend time in "ME Ready To Boot Event" in HECI driver since ME is disabled by BIOS.

This release also includes the PBIT software(*) V1.5 ID13003 implementing the following evolutions:

- Test Ethernet: Correction for loop mode test. The test indicates: "`NOT EXECUTED ANYMORE`" after 100 loops
- Test Memory: error detected in loop mode corrected
- Correction for command `kdiag cfg + flag`
- System test: Link Training on PCIe bridge may cause PBIT system test failure
- Add GEN3 support for Link Speed

(*) PBIT - Power on Built In Test - is a software developed by Kontron. It is an optional product. For more information, please contact your field representative.

11.5 BIOS ID13205 Release Notes

The following lists the evolutions or enhancements relative to this BIOS release:

- Add BIOS version in banner at startup
- Disable AES instructions in CPU for Export Control compliance
- Fix Bugzilla #6527: double refresh rate for DDR3 Memory from 64 ms(1X) to 32 ms(2X) for VX304x WA/RA/RC class board.
  This feature is configurable in setup using either ASR mode (default) or SRT mode.
  Program SPD Byte 31 with the value 0x05 (ASR enabled) by default for all DDR3 memories configuration using "`smbutil`" EFI shell utility.

Because BIOS programs DDR3 registers for each DIMM to set ASR or SRT mode, the boot time is increased of about 1 sec.

- Fix Bugzilla #6716: add CPU frequency selection in setup Kontron CPU Configuration menu for VX3044 board
- Fix processor current speed and current voltage on SMBIOS table 4
- Fix Bugzilla #6894: take into account Variant in VPD EEPROM for PLX Silicon Revision instead of PCI RID (08h) because it can be corrupted if EEPROM is not correctly set.
- Fix Bugzilla #6900: SW3 switch inversion for cTDP and Backplane PCI-E port size.
  Update of document CA.DT.A98 User's Guide for VX3042 & VX3044 boards is planified and current BIOS release is compliant to the correct documentation.
- Fix Bugzilla #6903: Max payload size for PLX Silicon Rev. BA must be equal to 256 bytes when updates are made in VPX EEPROM for changes in VPX configuration
- Update `kvpx` command for ports A,B and C PCIE switch in case of port size = 2x4 or 4x2
- Fix number of bytes (16 instead of 256) in EFI shell command smbutil /rtc
- Add "****** Low battery ******" message on serial console for missing or weak battery

the BIOS displays this message once after a battery is fitted because Battery Low detector flag in RTC must be cleared by BIOS.

This release also includes the PBIT software(*) V1.6 ID13177 implementing the following evolutions:

- Fix status mode NT or Transparent for PLX PEX8725 device
- Fix Bugzilla #6851 (enhancement): add more messages in memory init and early DRAM test (POST) in case of failures
- Fix Bugzilla #6897: extend test NvRAM to full size 1Mb FMRAM
- Add Personality Module option conform to VX304x conf 1.11
- Add test skeletons etherPM_loop and etherPM_link for MOD-1BT module
- test core_dmi: add DMI link status control (x4, gen2).

(*) PBIT - Power on Built In Test - is a software developed by Kontron. It is an optional product.
For more information, please contact your field representative.

11.6 BIOS ID13287 Release Notes

The following lists the evolutions or enhancements relative to this BIOS release:

- First release based on new AMI-BIOS source code V27: one work-around remains to avoid infinite loop on RTC polling
- Work-around for Bugzillas:
  - #6556: patch EEPROM NVM Intel Gigabit LAN v0.F4 for i82579 chipset with 0xFF at offset range [0x80-0x583] to avoid RX-ERR on Ethernet test between 2 VX304x boards
  - #6667: on VX304x board PCB Revision A & B, COM2 signals RTS2/CTS2 are not correctly wired on hardware; this work-around do not apply for VX304x board PCB Revision C & D
  - #6939: in CPLD if WatchDog_At_Startup enabled, the watchdog must be stopped if not enabled in BIOS setup
- Fixed Bugzillas:
  - #6832: no boot on PXE available if IGFX is disabled
  - #6907: "`kvpx -plx_eeprom conf`" command hangs if the PLX EEPROM is programmed in 1x8, no auto-update enabled in Setup but SW3[3:4] configured in 2x4 or 4x2
  - #6923: `kmac` error with `-lan` option
  - #6929: PLX Status Mode is not correct in PCI algorithm
  - #6940: COM2 initialization error for RTS2/CTS2 polarity
  - #6943: disabling PCIe Switch in Setup do not work
  - #6946: SMBIOS table is not correctly updated for CPU speed
  - #6948: CRC bytes inverted in SPD internal table and EEPROM
  - #6952: PLX EEPROM updated failed and crash the board
- Enhancements:
  - WatchDog_At_Startup: increase default watchdog to 21 sec in BIOS, add add refresh in BIOS POST to avoid watchdog trig; also, let the default action to Power Cycle instead of Reset
  - Add SSD_Reset feature in Setup to maintain the SSD on-board device in reset
  - Suppress half/full duplex menu in Serial Configuration for RS-485 mode because not implemented in hardware
  - Upgrade BoardID field in PLX scratchpad for VxFabrix™ according to the ProductName field in SMBIOS table Type 1
  - Add HotPlug support on PCIe Switch on ports 9,10 and 11 by programming the PLX EEPROM
  - Add I2C multi-master support on I2C buses connected to CPLD
  - Add check for valid temperature in MSR to avoid out of spec shutdown under OS at very low temperature (less than -5 C)
  - Add `kflash -ver` to display current BIOS version
  - Add option in "`kmac -prog`" for new EEPROM image for 10G
  - Add option "`-c|noc`" in "`krcconfig`" for disabling C-STATES

This release also includes the PBIT software(*) V1.7 ID13274 implementing the following evolutions:

- PBIT system management enhanced with new system-edit menu and error reporting improved.

(*) PBIT - Power on Built In Test - is a software developed by Kontron. It is an optional product. For more information, please contact your field representative.

11.7 BIOS ID13346 Release Notes

The following lists the evolutions or enhancements relative to this BIOS release:

- Improves memory initialization by reading DDR3 SPD only one time when MRC Fast Boot parameter is enabled in setup.
- Fixes a memory leak issue when using binary file as parameter in "`kvpx`" and "`kmac`" EFI shell commands.
- `kflash` command: fixes "`-ver`" issue
- `ktemp` command: adds power information.
- "PowerOnWait" feature added: new option in Kontron Board Misc Configuration menu to control board power on/off.
- `kvpx` command in `conf` option: adds NT mode status of the PCI-E switch for VPX.
- `ksata` command: new command for FDM-SATA and SSD flash devices to program "Early Power Down" or "Write-Protect" features.
- Adds `timezone` command under EFI shell.
- Fixes Bugzilla #6958: watchdog at startup is set with a time-out value of 511 sec instead of 21 sec.
- Fixes Bugzilla #6987: suppress delay for VPX propagation of SYSRESET#.
- Fixes CRP#4194/Bugzilla #6989: screen issue at start-up and loss of serial redirection with some monitors that are not "MultiSync".
- Fixes Bugzilla #6994: E.C. Level Hardware 20004XX on VX3042 board is not compliant with BIOS code to patch VPX EEPROM of PCI-E switch.

This release also includes the PBIT software(*) V1.8 ID13310 implementing the following evolutions:

- PBIT version 1.8 ID13310

  - add therm test to display PKG, CPU, PCH temperatures and powers (for production use only).

(*) PBIT - Power on Built In Test - is a software developed by Kontron. It is an optional product. For more information, please contact your field representative.

Chapter 12 - Use Cases

This chapter gives some advise for following pratical cases:

> DEPLOY : How to deploy VX304x - BIOS, section 12.1 page 117

> DEVEL: How to develop applications with VX304x - BIOS, section 12.2 page 118

> EVAL: How to benchmark VX304x - BIOS, section 12.3 page 118

> TROUBLESHOOT: How to troubleshoot VX304x - BIOS, section 12.4 page 118

12.1 DEPLOY: How to deploy VX304x - BIOS

Deploying with VX304x boards usually requires to handle the following tasks:

> Cloning a board,

> Managing a pool of deployed boards.

12.1.1 Cloning a board:

To be able to replace a VX304x with another one in a system, cloning allows to duplicate VX304x settings in the new board prior to replacement. This is how to proceed with VX304x:

> **On Original VX304x**

Duplicate the hardware settings. (see VX304x User's Guide: chapter Configuration)

Duplicating BIOS settings:

BIOS and BIOS settings are stored in the BIOS FLASH device itself. See Annex A.3 page 120 of this document to know how to save a BIOS ROM image.

> **New VX304x**

Check the Board EC level to insure the BIOS + Settings you are going to install are compatible with the hardware evolution.

See Annex A.1 page 119 on how to program the new BIOS + settings.

Boot the board and set the Date Time to the correct date/time.

Now the new board is a functional clone ot the initial VX304x.

Once the system has been qualified, it may be a good idea to save the image of the BIOS + Settings for later use.

In the case of removable storage like USB or SATA FLASH mezzanine, refer to VX304x User's Guide (CA.DT.A95) for details of removal and fitting operations.

For large programs, Kontron can contribute with high level software to automate this cloning task. Contact support-kom-sa@kontron.com for details.

12.1.2 Managing a pool of VX304x:

To manage a pool of boards, the main task is to identify and track board using serial number, E.C. Level, BIOS version, MAC addresses, etc... possibly without having to take the system apart to look at its labels.

See chapter 2.2 of VX304x User's Guide about the board identifcation labels.

See section 5.5 page 27 on VPD of this document to retrieve the board serial number and E.C. level.

See VPD Tool in the Linux BSP document to know how to get this information from a Linux OS running on the board.

The BIOS information is also transmitted from the BIOS to the OS using a software table in memory, use the `dmidecode` command to retrieve this information from Linux.

Kontron maintains a database of all the boards sold to customers. This includes customer, program, system and any information you may wish to have maintained by us, allowing to retrieve an exact board pool status, whenever needed.

Contact support-kom-sa@kontron.com for details.

12.2 DEVEL: How to develop applications with VX304x - BIOS

TBD

12.3 EVAL: How to benchmark VX304x - BIOS

TBD

12.4 TROUBLESHOOT: How to troubleshoot VX304x - BIOS

» SETUP not accessible

If setup is not accessible, make sure the board IS operational in rescue mode (see VX304x User'sGuide for Boot from the Rescue SPI Flash).

» SETUP accessible but OS not booting

Enter setup by pressing the <F2> key as indicated at BIOS boot time and check if the boot device is visible in the boot device list. See chapter 7 page 40 “Boot Method and Priority” of this document

Eventually restore the default manufacturing setup configuration. See chapter 9 page 53 “Save and Exit Menu” to restore setup.

Appendix A - How to Update and Restore the BIOS

A.1 Update BIOS from UEFI Shell using USB device

This section details the update of the AMI BIOS Firmware on a VX304x board. An USB key with the BIOS image to flash will be used.

Operating Mode

> Copy the BIOS image under the USB device

> Boot VX304x on UEFI shell. If necessary enter the BIOS SETUP pressing <F2> during the boot sequence. Then navigate to Save & Exit Menu and select UEFI shell in Boot override menu and boot under UEFI shell. Plug the USB device on the concerned USB interface

> Enter command

```
map -r
```

> fs0: file system must become visible, then Enter

```
fs0:
```

> Eventually use cd command to reach a directory where the Bios image is stored. Use ls to display file list

If BIOS image is named `VX304x_IDYYXXX.bin` then flash the BIOS entering command

```
VX304x > kflash -p -r  VX304x_IDYYXXX.bin
```

Do not turn off nor reset the board until the end of the command. This prevent the system to boot at next power on.

> Wait about 1 minutes and 30 seconds and check if message "`image are equal`" is displayed. If not, do again the flash update. When upgrade is finished without any errors, then turn off the system and do a fresh cold start in order to boot with the new BIOS.

The serial console displays a toolbar `[======]` during Flash process to show the progression of the Flash update while the graphical screen not.

A.2 Restore or Update BIOS from Rescue BIOS

A rescue BIOS is available on any VX304x CPU. It is possible to boot on rescue BIOS and update the main BIOS with the rescue BIOS.

When board is powered off, set micro switch SW2 function 1 to ON. Then Boot on Rescue BIOS and EFI-Shell. If necessary enter BIOS SETUP with F2 in boot sequence and then navigate to Save & Exit Menu and select UEFI shell in Boot override menu. Check if EFI-Shell prompt is VX304x-RESCUE.

> Enter command:

```
VX304x-RESCUE> kflash -c
```

Do not power down the board during update process. This behavior will prevent the board to boot.

> Wait about 1 minutes and 30 seconds the command end.

The BIOS is restored. Power off the board, set micro switch SW2 function 1 to Off then boot on Main BIOS.

A.3 Record BIOS image ROM and setting from UEFI Shell using USB device

This section details the record of the AMI BIOS Firmware and its setting of a VX304x board. An USB key will be used to store the BIOS image

» Operating Mode

> Boot VX304x on UEFI shell. If necessary enter BIOS SETUP with F2 in boot sequence. Then navigate to Save & Exit Menu and select UEFI shell in Boot override menu and boot under UEFI shell. Plug the USB device on the concerned USB interface

> Enter command

```
map -r
```

> fs0: file system must become visible, then Enter

```
fs0:
```

> Eventually use cd command to reach a directory where the Bios image is stored. Use ls to display file list

If BIOS image is named `VX304x_CLONE.bin` then copy the BIOS image entering command

```
VX304x> kflash -s  VX304x_CLONE.bin
```

> Wait 20 seconds. When finished without error then the BIOS ROM image is stored onto the USB device.

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-MAR-Toulon@kontron.com
Support: GSS-MAR-Toulon@kontron.com

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