USER REFERENCE MANUAL

VX3035 AMI BIOS

SD.DT.F97-9e - May 2016

POSSIBILITIES START HERE kontron

VX3035 AMI BIOS User Reference Manual

Lise-Meitner-Str. 3-5

86156 Augsburg

Germany

www.kontron.com

REVISION HISTORY

PUBLICATION TITLE:	VX3035 AMI BIOS User Reference Manual	
DOC. ID:	SD.DT.G97-9e	
Revision	Brief Description of Changes	Date of Issue
9e	New BIOS ID15300 based on BIOS ID13245	05-2016
8e	New BIOS ID15084 based on BIOS ID13127	04-2015
7e	New BIOS ID13245	09-2013
6e	New BIOS ID13127	05-2013
5e	Update of chapter 11 - BIOS Version Description New BIOS ID12347 & ID13010	01-2013
4e	Update of chapter 11 - BIOS Version Description	11-2012
3e	Update of chapter 11 - BIOS Version Description	10-2012
2e	Update of: - Section 10-1-24 - kmac command - Chapter 11 - BIOS Version Description	09-2012
1e	Update of - Chapter 10 - EFI Shell - Chapter 11 - BIOS Versions Description	07-2012
0e	Initial Version	04-2012

SYMBOLS

The following symbols may be used in this manual:

DANGER DANGER indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING WARNING indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION CAUTION indicates a hazardous situation which, if not avoided, may result in minor or moderate injury.

NOTICE NOTICE indicates a property damage message.

Electric Shock!
This symbol and title warn of hazards due to electrical shocks (> 60V) when touching products or parts of them. Failure to observe the precautions indicated and/or prescribed by the law may endanger your life/health and/or result in damage to your material. Please refer also to the "High-Voltage Safety Instructions" portion below in this section.

ESD Sensitive Device!
This symbol and title inform that the electronic boards and their components are sensitive to static electricity. Care must therefore be taken during all handling operations and inspections of this product in order to ensure product integrity at all times.

This symbol indicates general information about the product and the user manual.
This symbol also indicates detail information about the specific product configuration.

This symbol precedes helpful hints and tips for daily use.

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

As a precaution, in case of danger, the power connector is the product's main disconnect device and must be easily accessible.

⚠CAUTION

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.
Earth ground connection to vehicle's chassis or a central grounding point shall remain connected. The earth ground cable shall be the last disconnected or the first connected during operations of cabling.

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.

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

TRADEMARKS

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

Table Of Contents

1 / Overview

This manual introduces the SETUP, EFI-SHELL of the AMI BIOS firmware available on Kontron VX3035 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
- USB routing
- Serial Port, Terminal Type, Console redirection
- CPU Frequency
- Boot method and priority
- Security password

This manual applies to the release ID15084 of the AMI BIOS*
BIOS ID15084 is based on BIOS13127 and doesn't integrate the modification of the BIOS13245.

* Enter SETUP/MAIN menu to get BIOS ID

1.1 Structure

- Chapter 2 - Accessing SETUP Menu
- Chapter 3 to Chapter 9 -Sampling of menu items
- Chapter 10 -EFI-SHELL
- Chapter 11 - Known Limitations
- Chapter 12 - Use Cases
- Appendix A - How To Update the BIOS

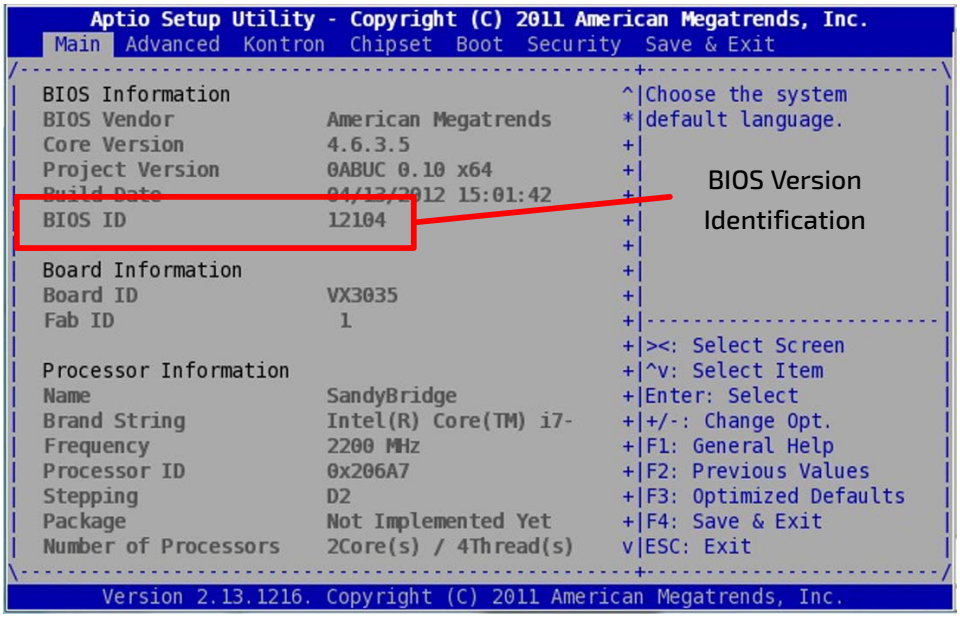
1.2 Related Documents

- **VX3035 Hardware**
  - VX3035 Hardware Release Notes CA.DT.A96
  - VX3035 User's Guide CA.DT.A95
- **VX3035 Software**
  - VX3035 - Release Notes for BSP Fedora 14 . . . . . SD.DT.F82

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 VX3035 board.

The following chapter details the items that are available on Kontron VX3035. 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:

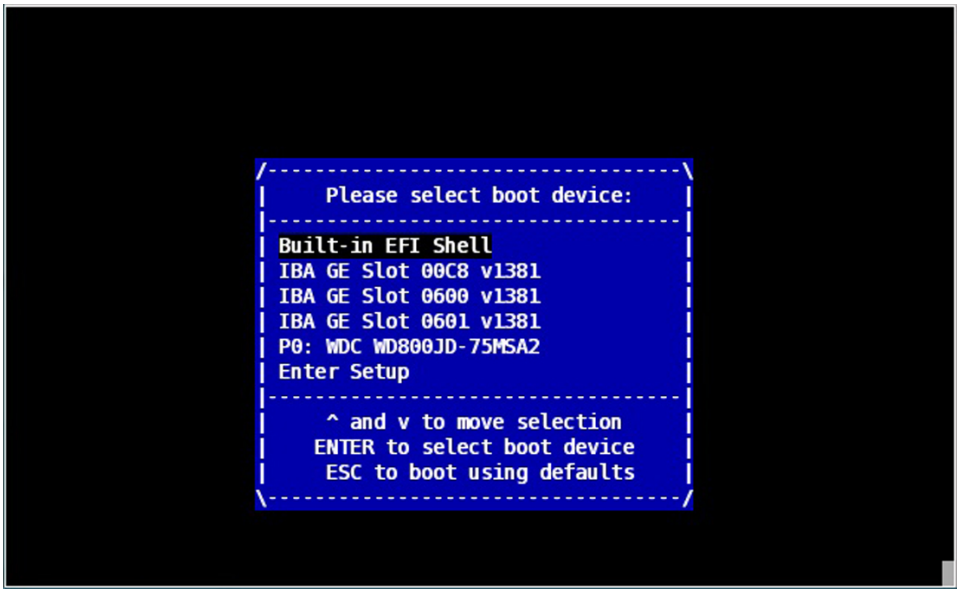
- A pop-up window prompts users to enable/disable the selected item.
- A window appears with a list of options to choose from.
- A window appears prompting the user to supply input.
- Links to the submenu.

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.



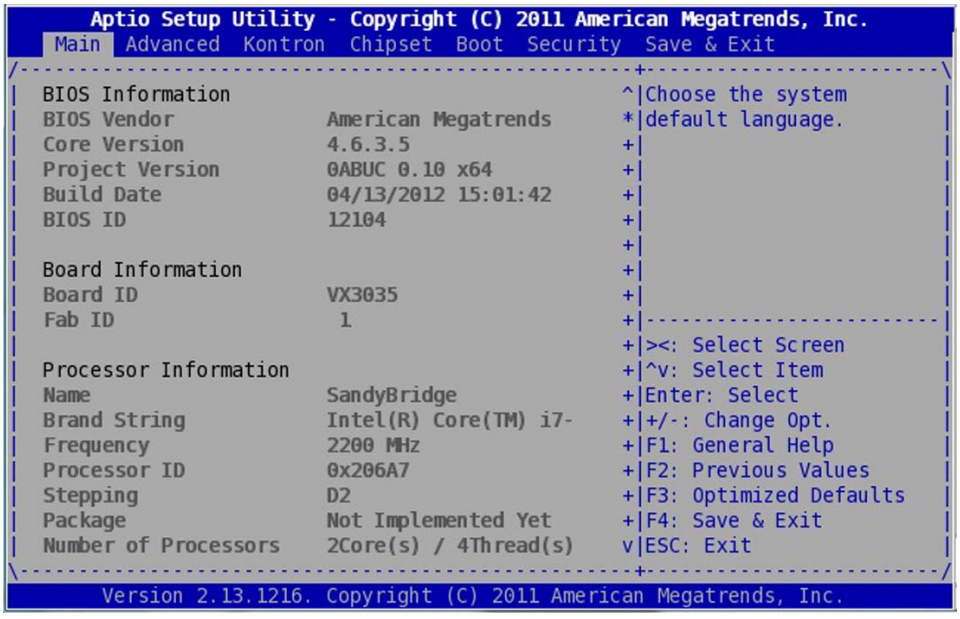
- Select a device from the list (Use the <↑> or <↓> to highlight the desired item)
- Press < ENTER > to boot the selected device or enter setup

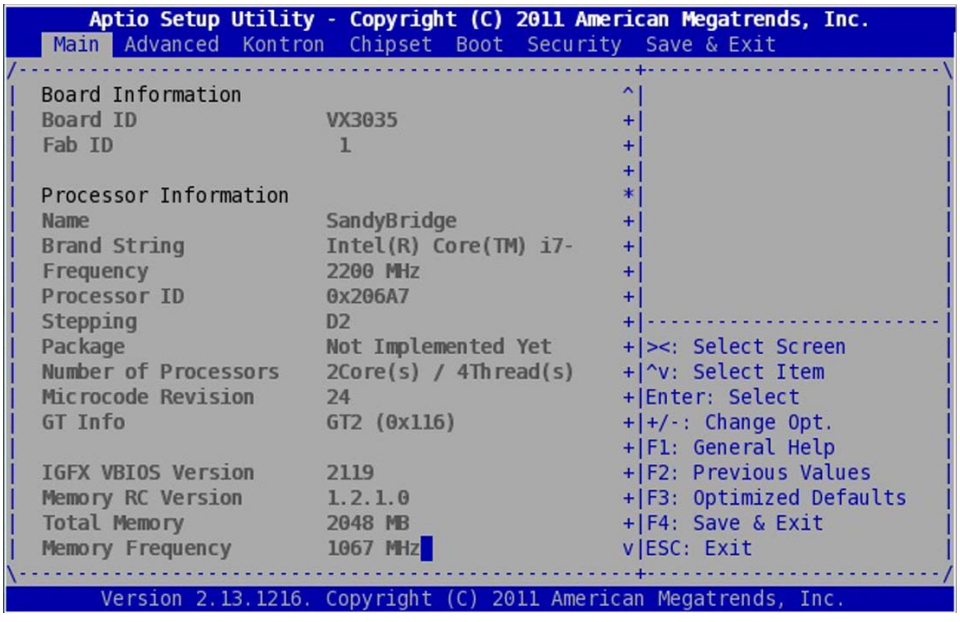
3 / Main Menu

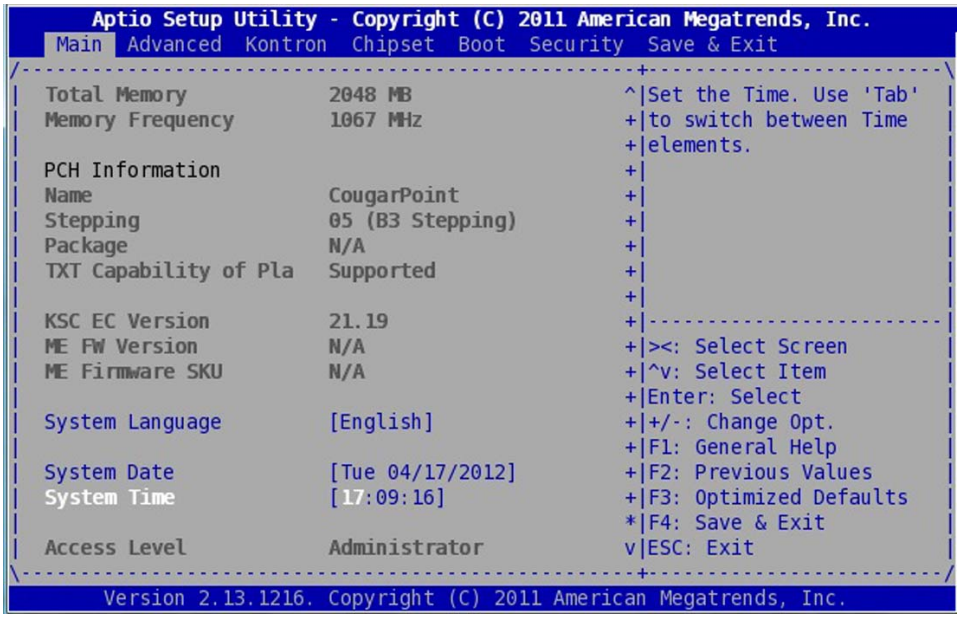
The Main Menu provides general system information and is the first accessible menu page.

Six parts or settings are available in the main menu:

- BIOS Information
- Board Information
- Processor Information
- PCH Information
- System Language
- System Date Time







The parts named "**Information**" display:

- BIOS ID and build date
- Board identity
- Processor name, frequency, stepping, number of cores and threads, graphic information, total memory size and frequency
- PCH (Platform Controller Hub) name, stepping

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.

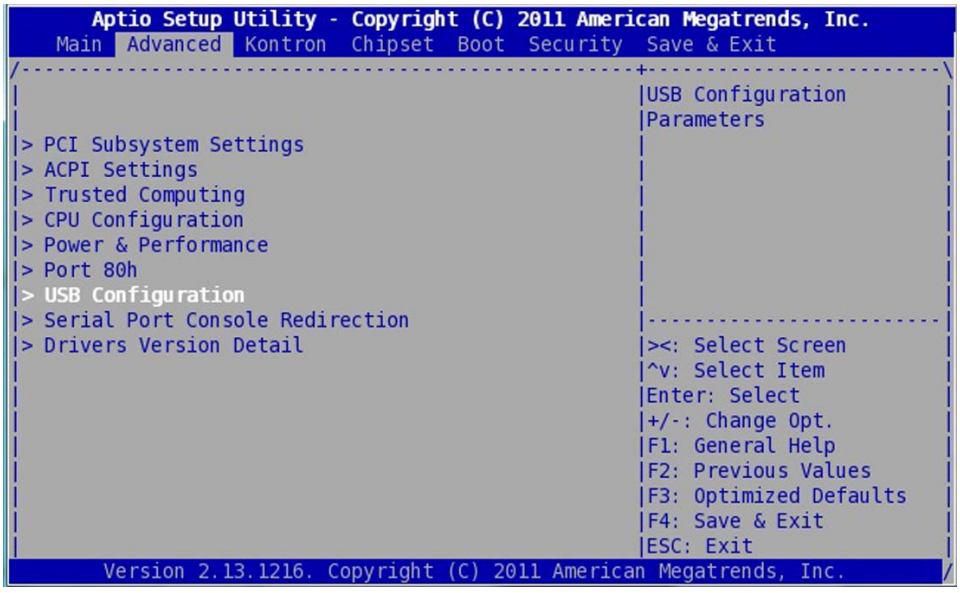
To update the Time, use the <+> or <-> keys to select the Hour (<+> to increase / <-> to decrease the hour), and press the <Enter> key to validate your choice. Proceed in the same way for the minutes and finally for the seconds.

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 VX3035 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.

4 / Advanced Menu



The Advanced Menu provides system-level controls to configure device settings:

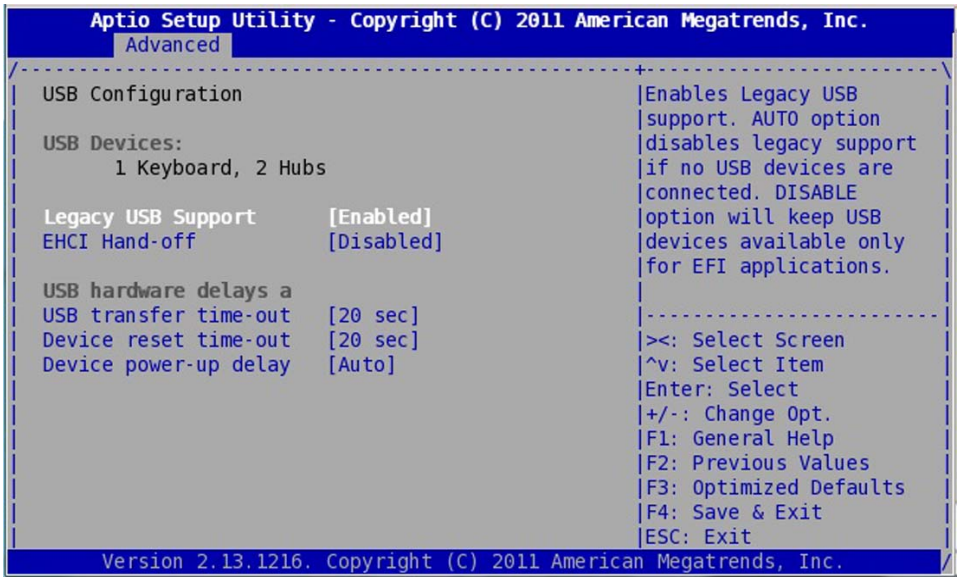
- **`USB Configuration`** (for Legacy support) Section 4.1 page 7
- **`Serial Port Console Redirection`** Section 4.2 page 8

The other following submenus are Reserved and Not intended to be used:

- PCI Subsystem Settings
- ACPI Settings
- Trusted Computing
- CPU Configuration
- Power & Performance
- Port 80h
- Drivers Version Detail

4.1 USB Configuration

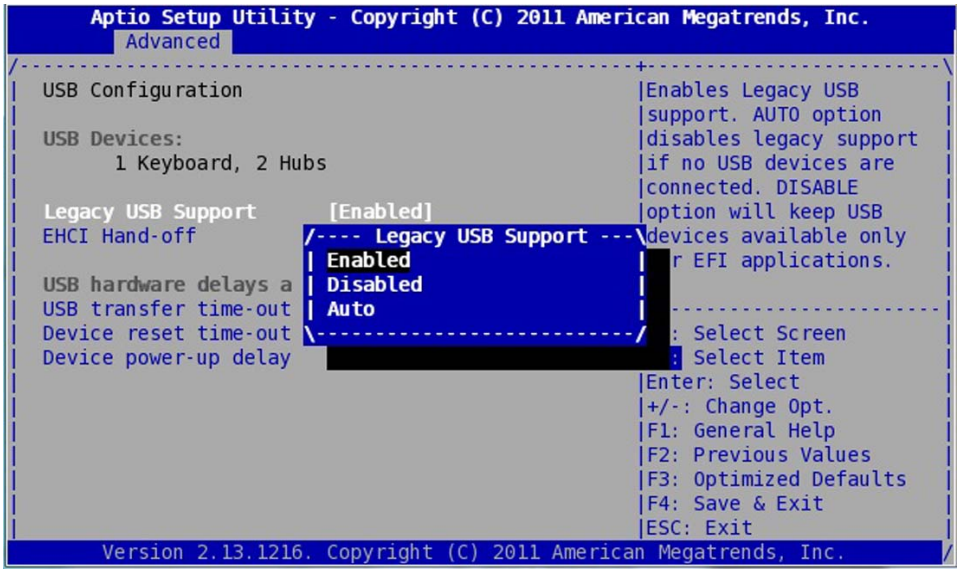
This menu can be used to enable/disable the **Legacy USB Support** (such as DOS legacy environment) . It can be used to avoid booting on an USB device when an USB device is connected. This is the only option that is not reserved in the menu.



Other following options are Reserved and Not to be used:

- EHCI Hand-off
- USB transfer time-out [20 sec]
- Device reset time-out [20 sec]
- Device power-up delay [Auto]

4.1.1 Legacy USB Support



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

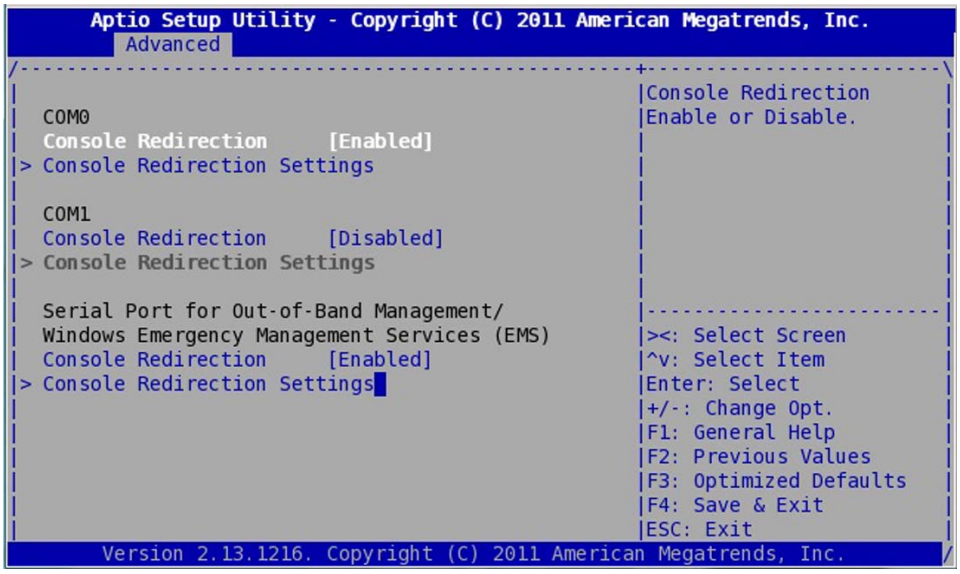
- Enabled
- Disabled
- Auto

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

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

4.2 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:

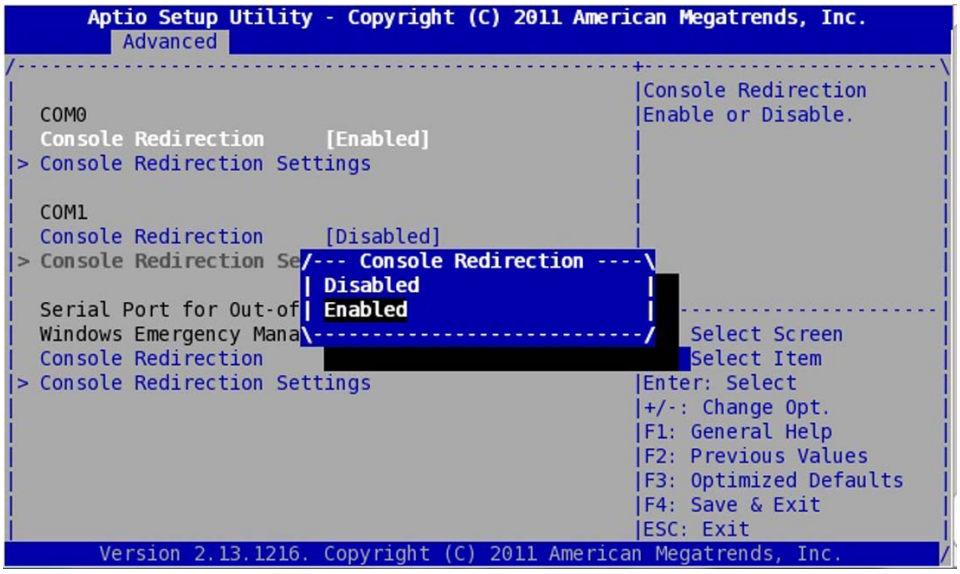


Other following options are Reserved and Not to be Used:

- Serial Port for Out-of-Band Management/
  Windows Emergency Management Services (EMS)
- Console Redirection

4.2.1 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 VX3035 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.



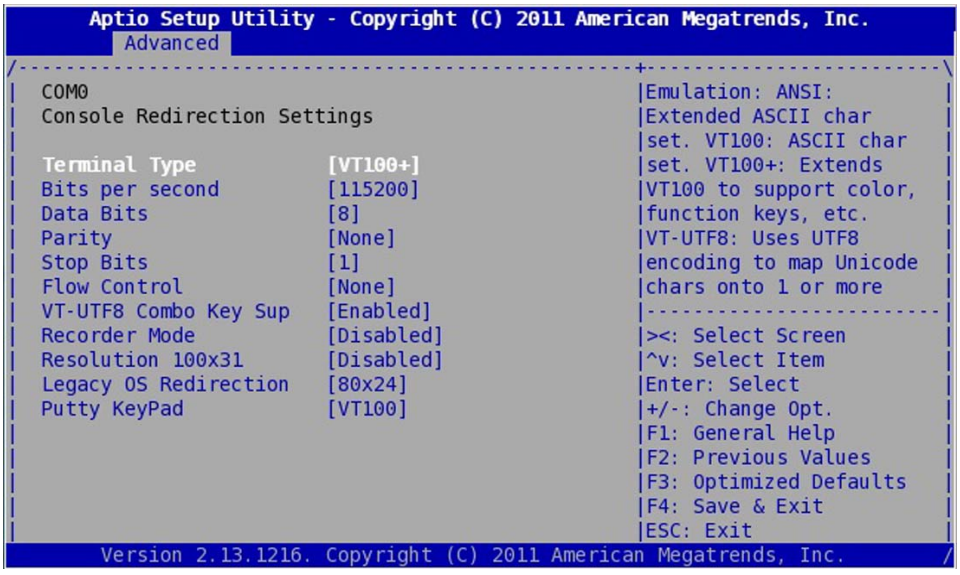
NOTICE

In case of 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.

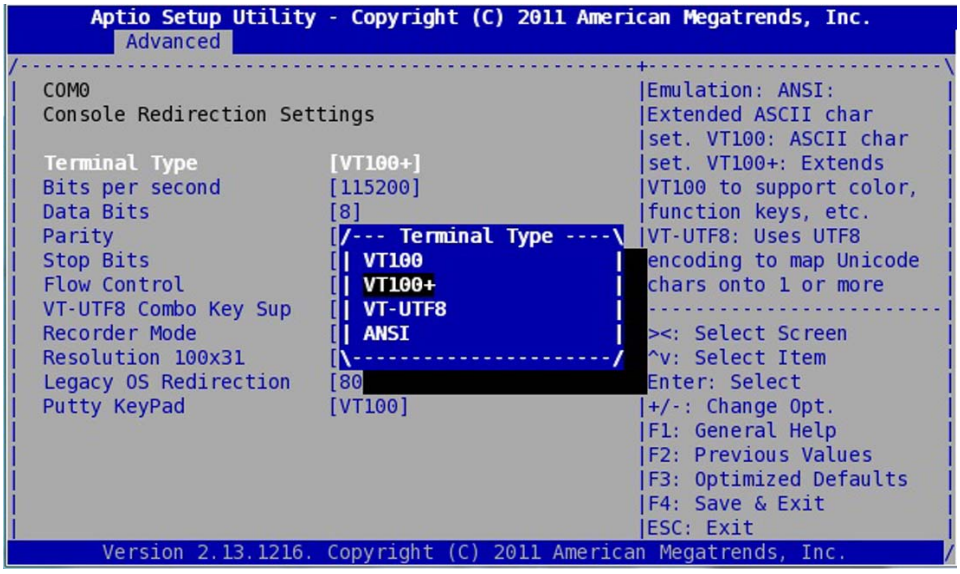
4.2.2 COM0/COM1 Console Redirection Settings

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



4.2.2.1 Terminal Type

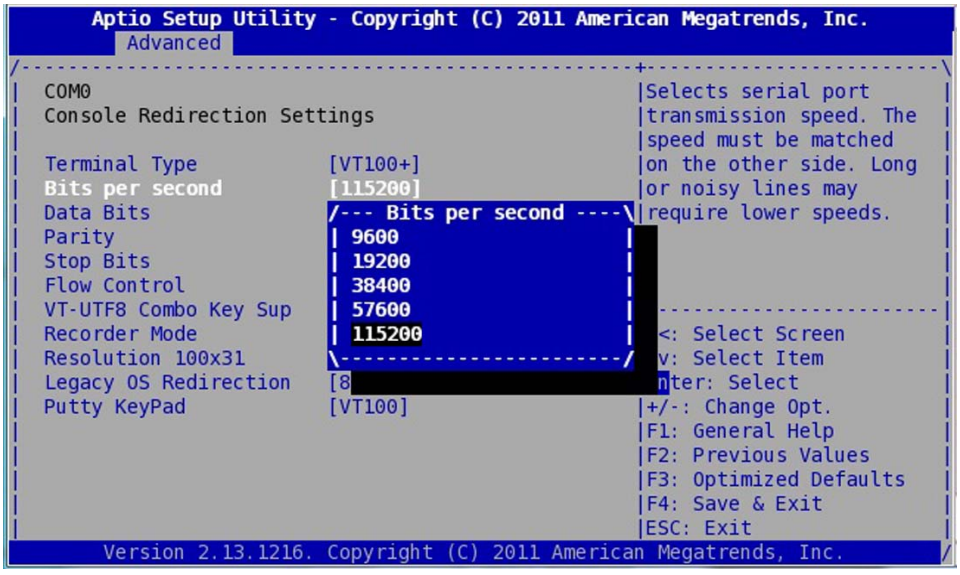


Set Terminal Type:

- **VT100** ASCII Char set
- **VT100+** Extends VT100 to support colours, functions keys
- **VT-UTF8** Uses UTF8 encoding to map Unicode onto 1 or more
- **ASCII** Extended ASCII char set

Default is **VT100+**

4.2.2.2 Bits per second

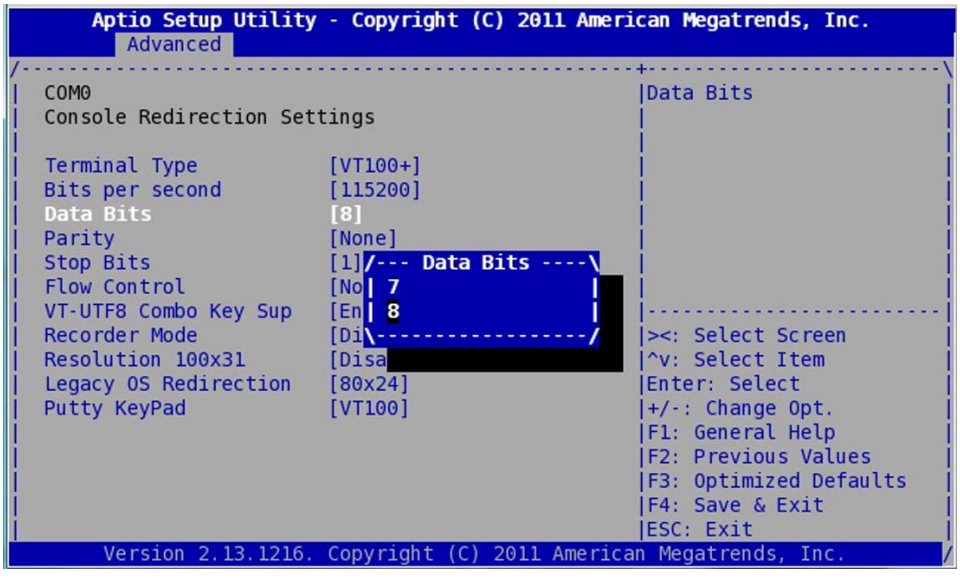


Set bits per second

- 9600
- 19200
- 57600
- 115200

Default and recommended value is **115200** bits per second for serial line baud rate on COM0 and COM1

4.2.2.3 Data Bits

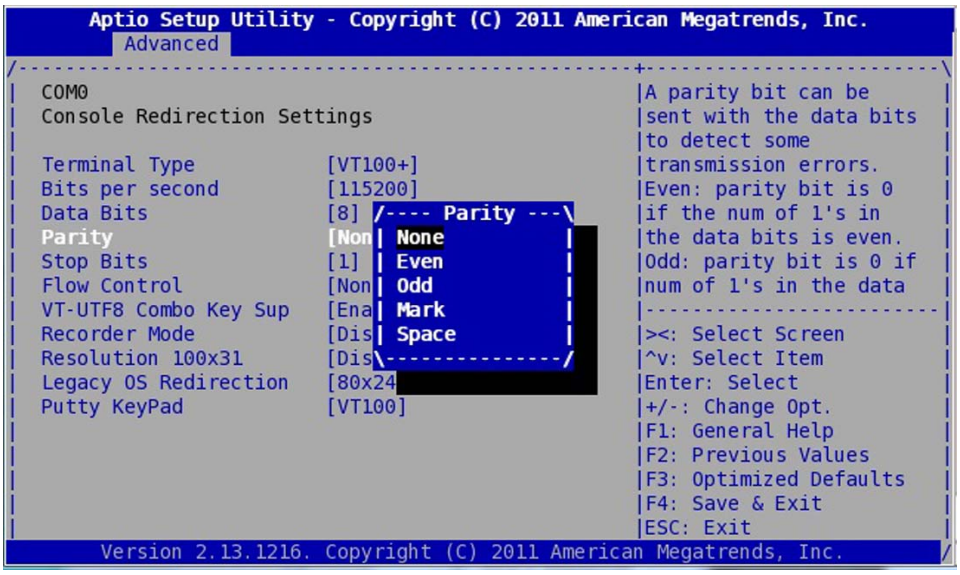


Set Data bit number for serial line COM0 or COM1

- 7
- 8

Default value is **8**.

4.2.2.4 Parity

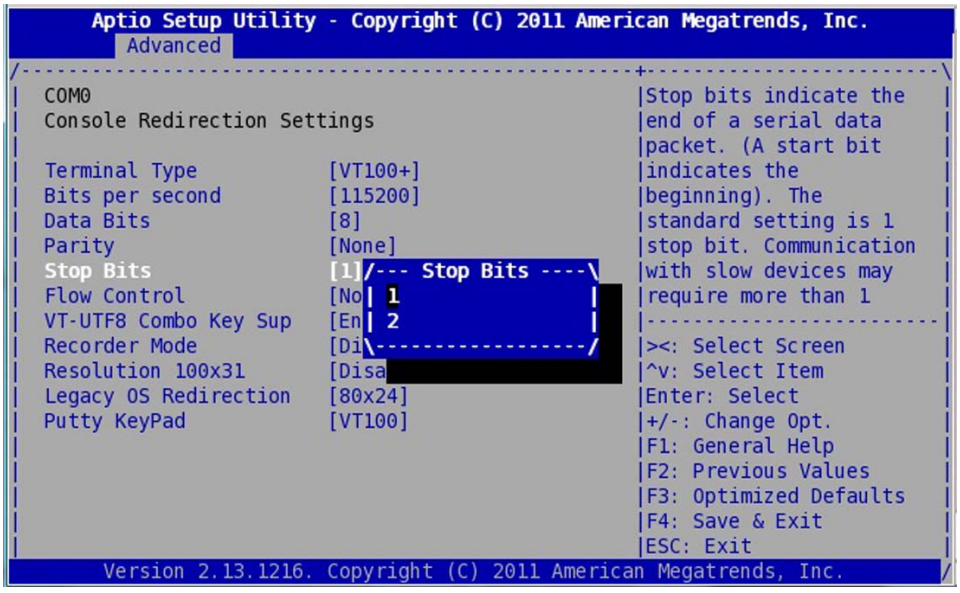


Set Parity bit

- **None**
- **Even**
- **Odd**
- **Mark**
- **Space**

Default for parity bit is **None**

4.2.2.5 Stop Bits

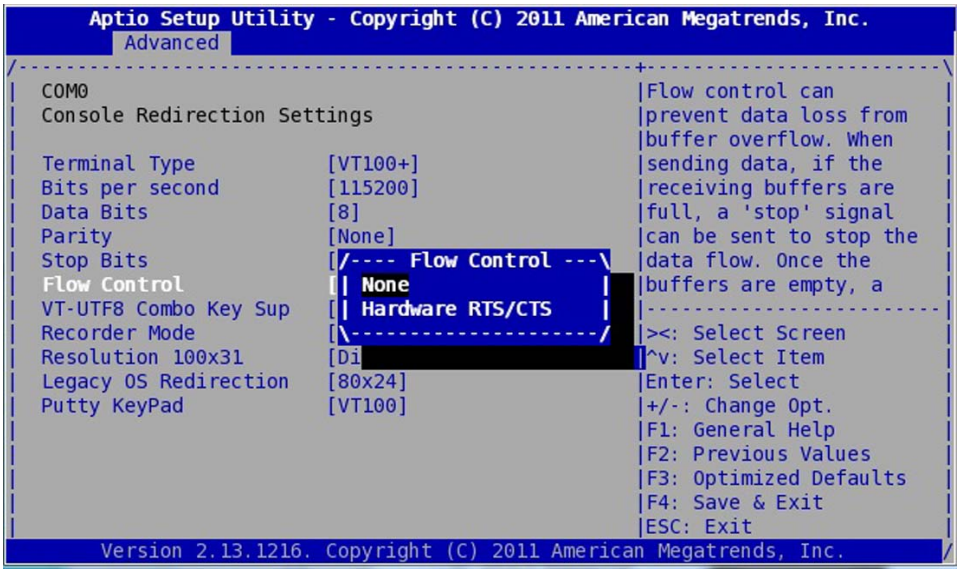


Set Parity bit

- **1**
- **2**

Default for stop bit is **1**

4.2.2.6 Flow Control



Set Flow Control or modem signals

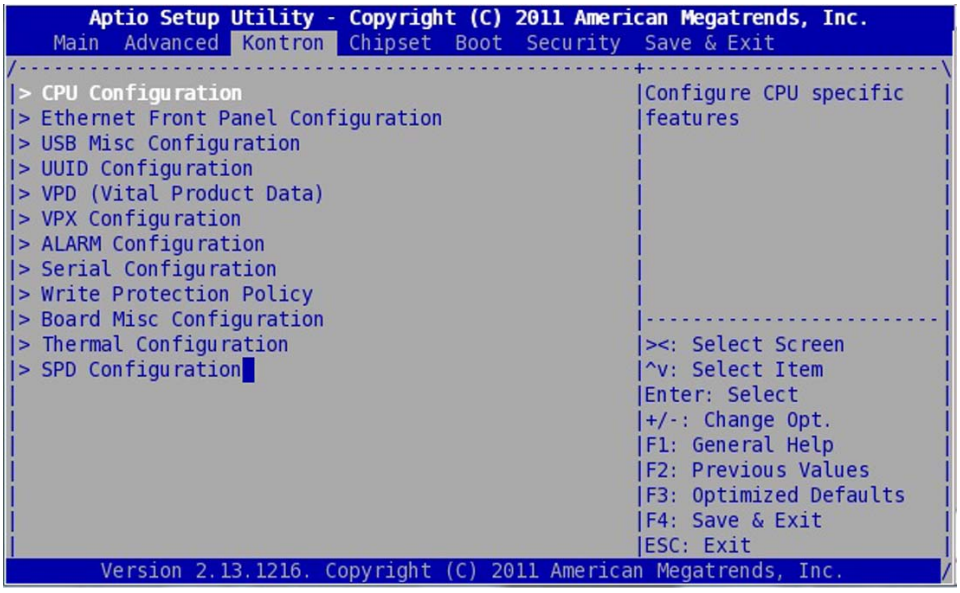
- **None**
- **Hardware RTS/CTS**

Default for Flow Control setting is **None**

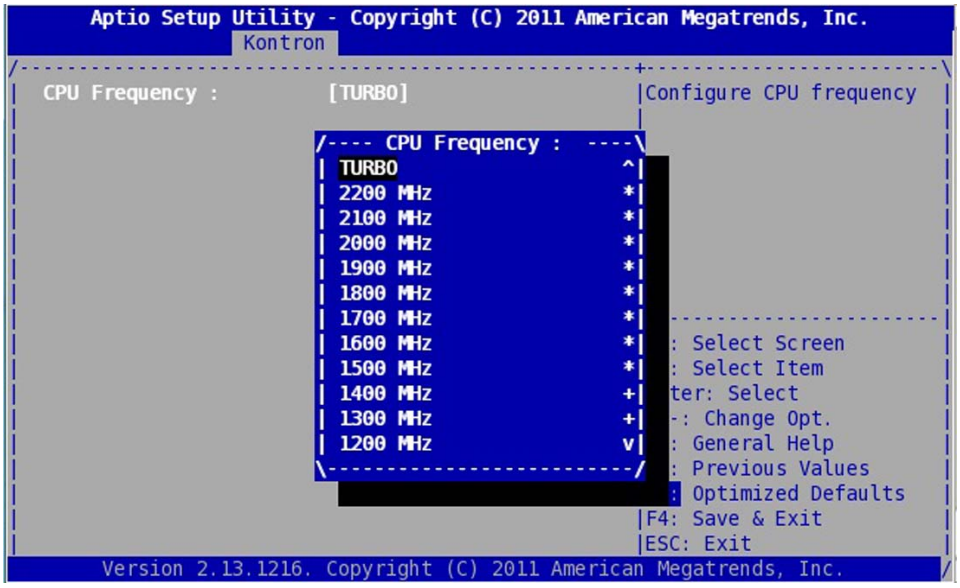
5 / Kontron Menu

The Kontron Menu provides system-level controls to configure specific VX3035 hardware design.

The different parameters are described in the following sections:



5.1 CPU Configuration



This option allows to set the CPU frequency.

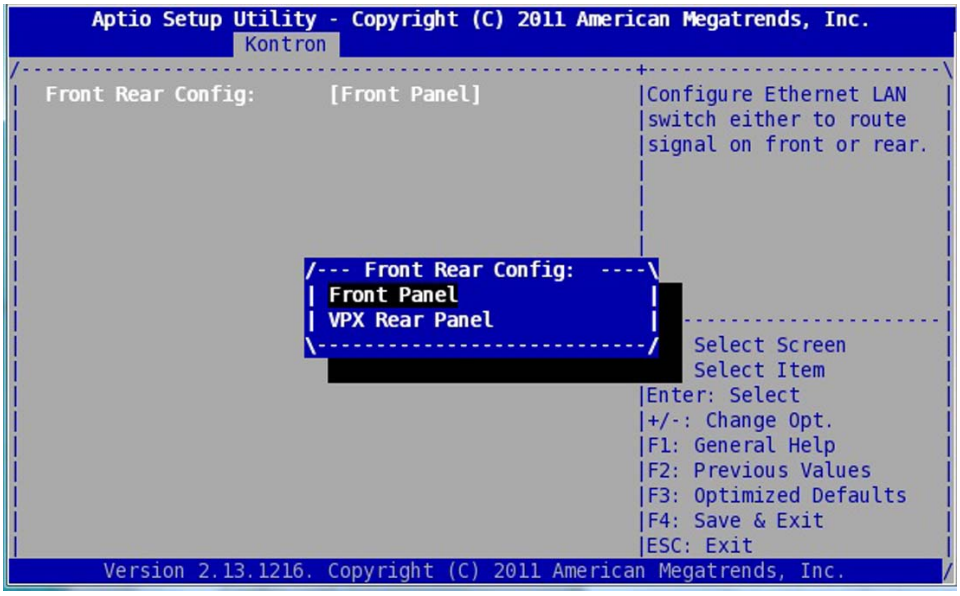
The setting named **TURBO** allows the CPU to boost its frequency above 2200 MHz according to the CPU load and temperature. But, to do so the Turbo mode option must be also enabled in the Advanced / Power & Performance / CPU – Power Management Control menu otherwise the maximum frequency 2200 MHz will be set.

The other settings will be used to force the CPU frequency to the indicated value. To guarantee the chosen frequency the Turbo mode option must be disabled in the Advanced / Power & Performance / CPU – Power Management Control menu.

The CPU frequencies can be set from 800 MHz to 2200 MHz with a 100 MHz stepping.

Default setting is **TURBO**

5.2 Ethernet Front Panel Configuration



Set LAN switch routing:

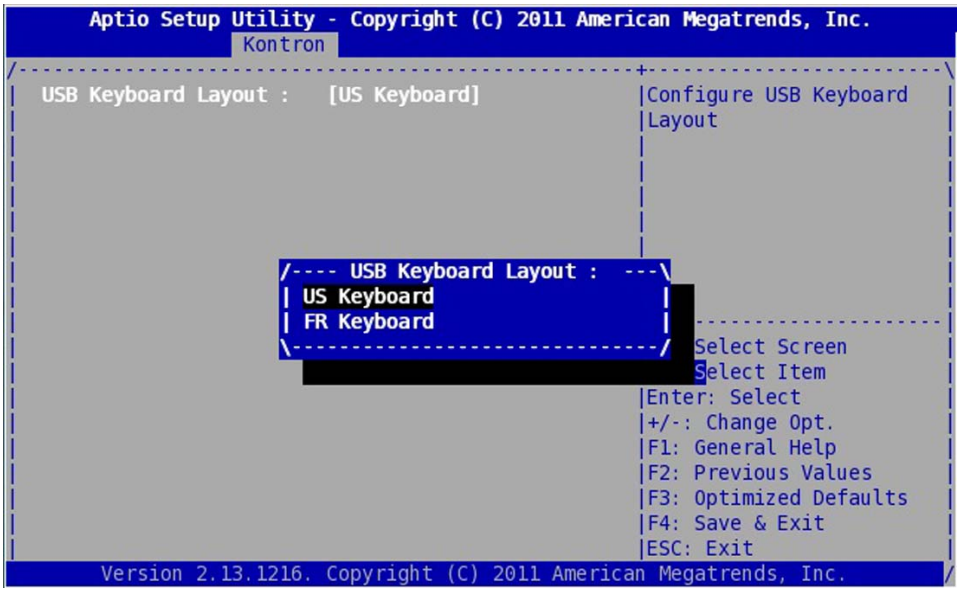
- Front Panel
- VPX Rear Panel

Default is **`Front Panel`**

This menu allows user to change LAN Ethernet Switch to route signal either on front panel or VPX Rear Panel using RTM (Rear Transition Module).

5.3 USB Misc Configuration

The following option is displayed :



Set the USB Keyboard Layout:

- US Keyboard
- FR Keyboard

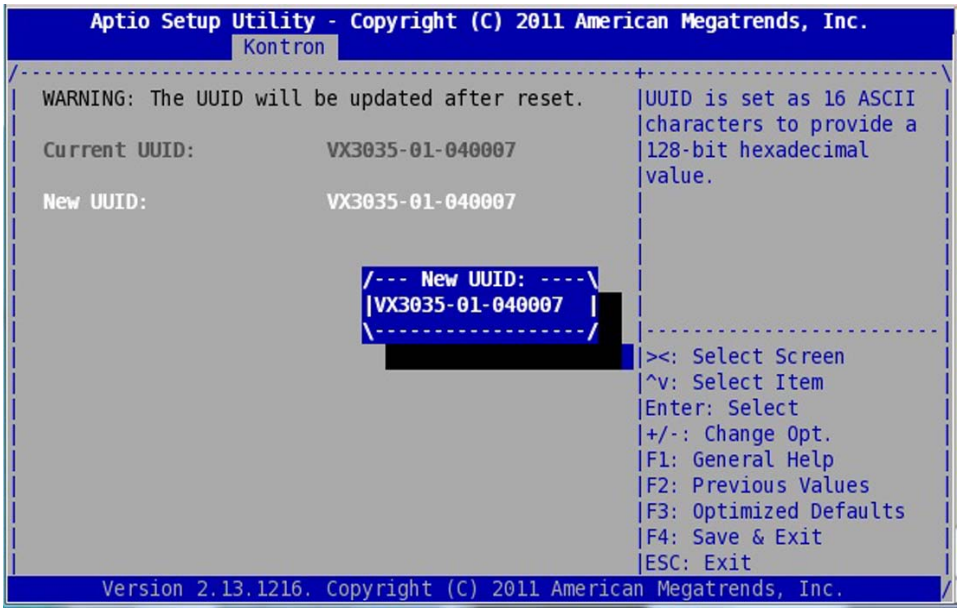
Default is **US Keyboard**.

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

NOTICE

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

5.4 UUID Configuration



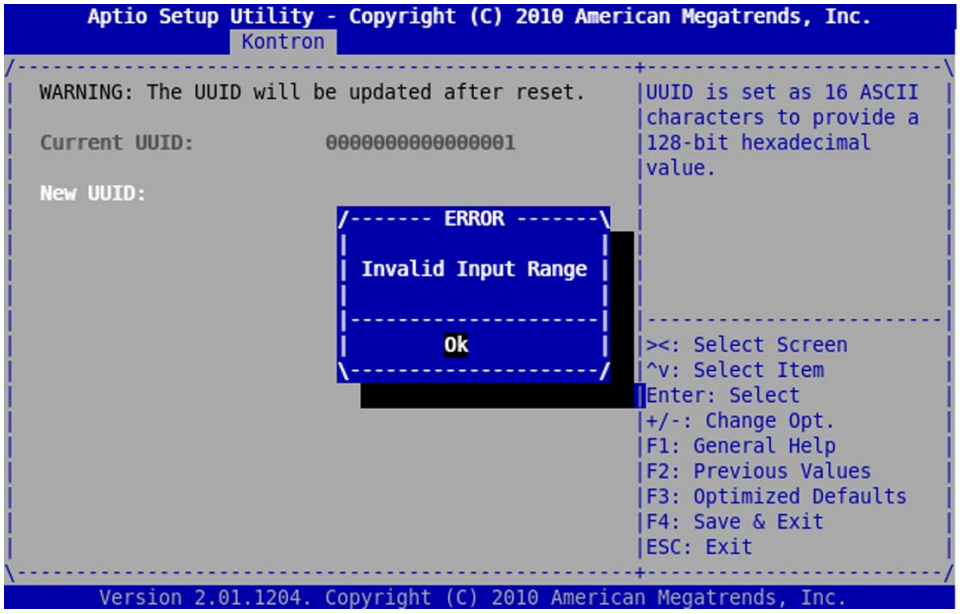
UUID stands for Universally Unique IDentifier also known as GUIDs (Globally Unique IDentifier). A UUID is 128 bits long, and can guarantee uniqueness across space and time. Please refer to RFC4122 documentation for more details about UUID.

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.

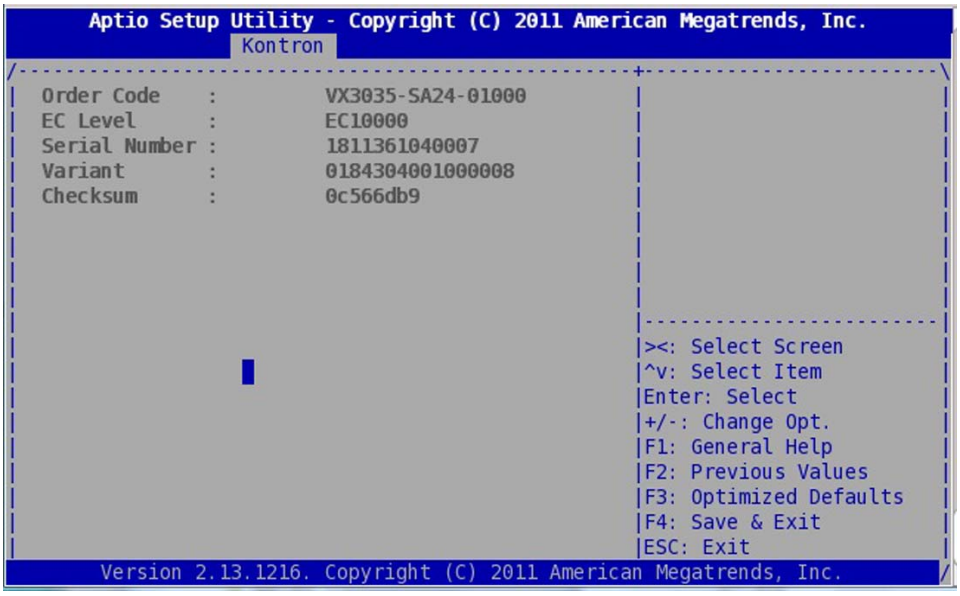
This submenu provides ability to user to modify the default value of the UUID (see picture above).

⚠CAUTION

Once the UUID is modified, it must be equal to exactly sixteen ASCII characters.
If not, an Error pop-up message appears on the screen (see picture below).
To cancel entering of a new UUID, type ESC key then enter key to close the pop-up message.



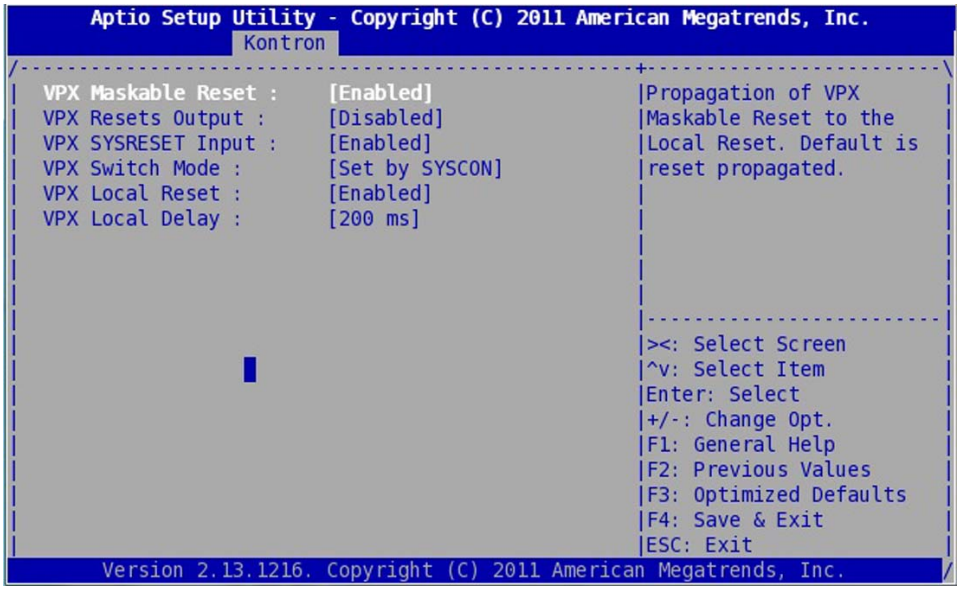
5.5 VPD – VITAL PRODUCT DATA



This menu only displays the Vital Product Data (VPD) information for VX3035. VPD are stored in VX3035 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

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.

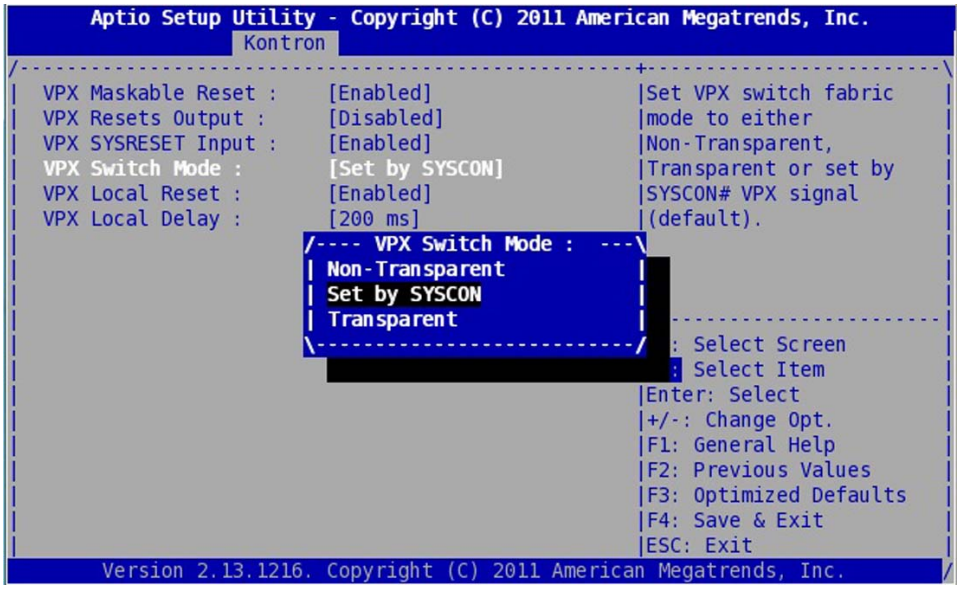
5.6.3 VPX SYSRESET Input

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 Mode



The **VPX Switch Mode** allows to set the VPX switch fabric device in a forced mode (**Transparent** or **NonTransparent** mode) disregarding the state of the **VPX SYSCON#** signal.

By default the VPX system controller board uses the **Transparent** mode for the VPX switch fabric device.

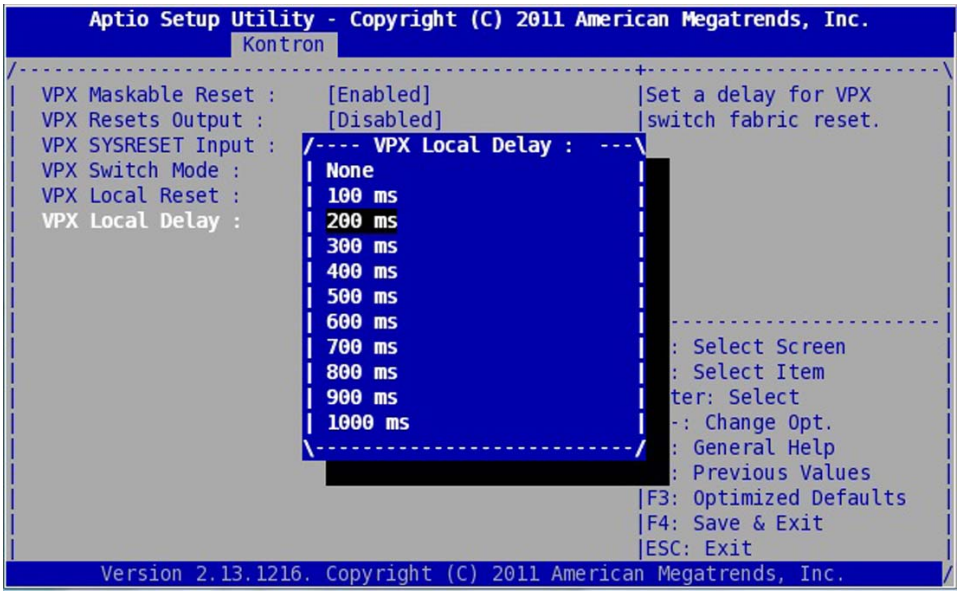
In **Transparent** mode, the EEPROM connected to the VPX switch device is not accessible from the BIOS.

Setting this parameter in **NonTransparent** mode allows the BIOS to access the EEPROM device even if the board is system controller of a VPX multi-boards system.

5.6.5 VPX Local Reset

The **VPX Local Reset** parameter allows VPX switch fabric devices to be enabled for VPX fabric connection. If this parameter is set to disabled, no VPX fabric connections are possible and the board will not appear in VPX discovery mecanism under OS.

5.6.6 VPX Board Delay



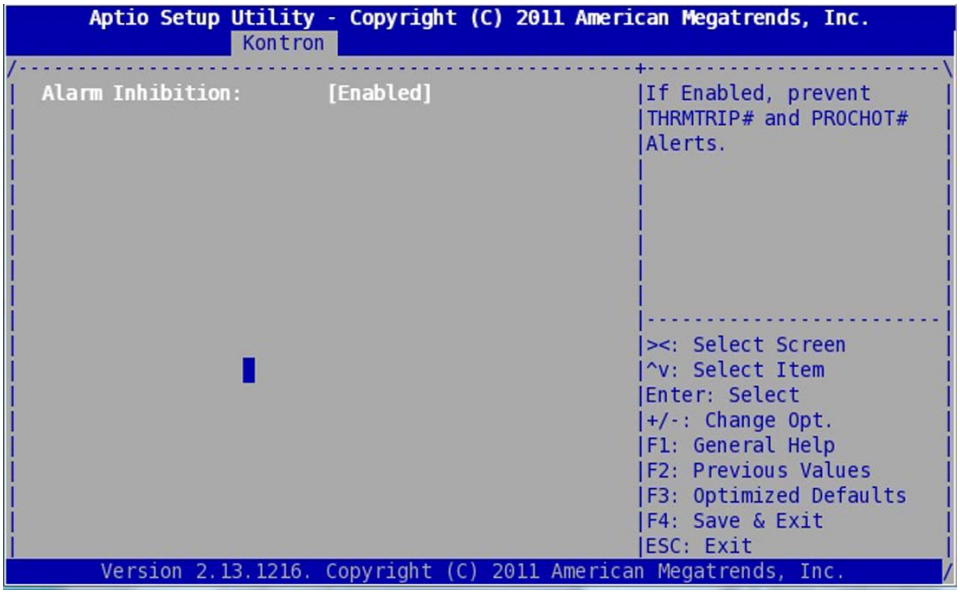
Set VPX Board Delay

- Value are:
  none
  100 ms
  200 ms
  ...
  1000 ms

Default is **200 ms**.

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

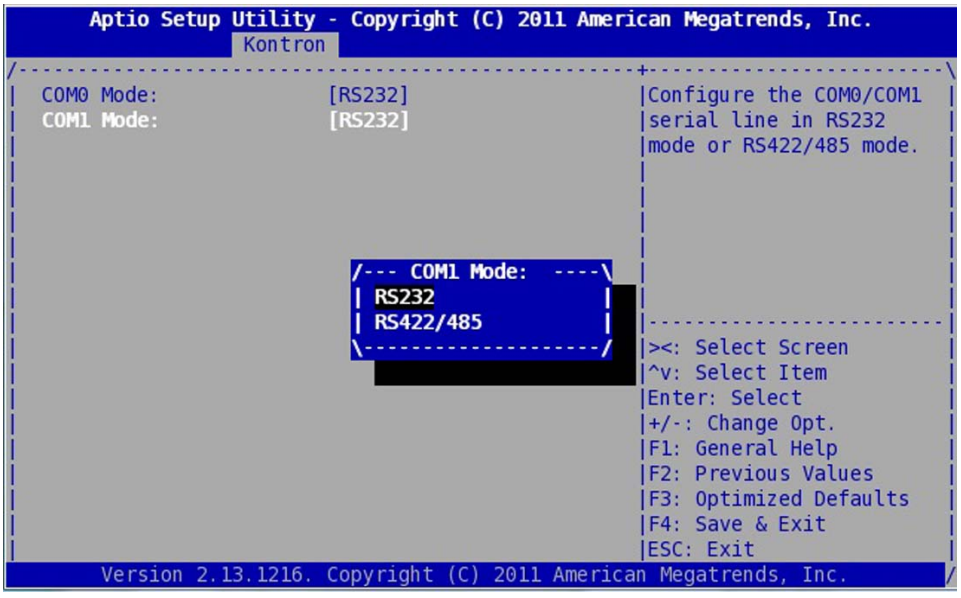
5.7 ALARM Configuration



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

⚠CAUTION It is strongly recommended not to disable this parameter for normal use. This parameter must be used with caution.

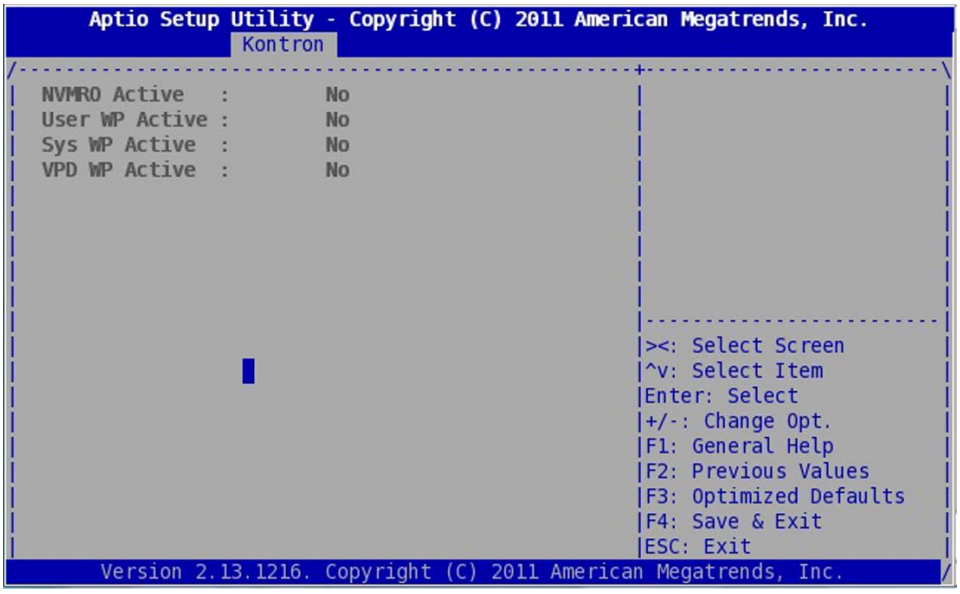
5.8 Serial Configuration



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.

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

5.9 Write Protection Policy



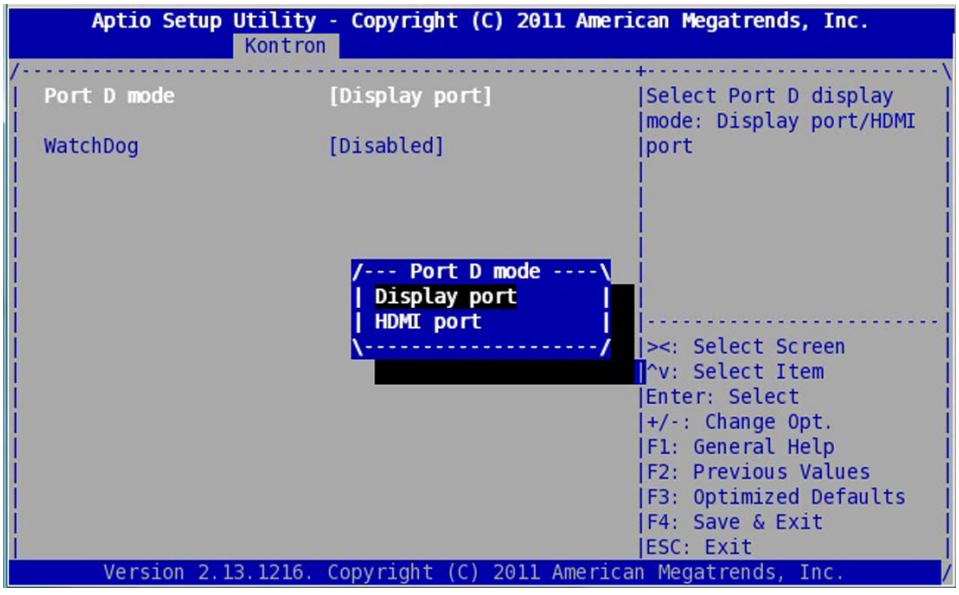
This menu displays the NVMRO status and the configuration of the VPD EEPROM, System EEPROM and FRAM write protection switches on the SW1 microswitch of the VX3035.

5.10 Board Misc Configuration

This menu is used to select the graphic port D mode:

- Display Port,
- HDMI Port

Default is **`Display Port`**.



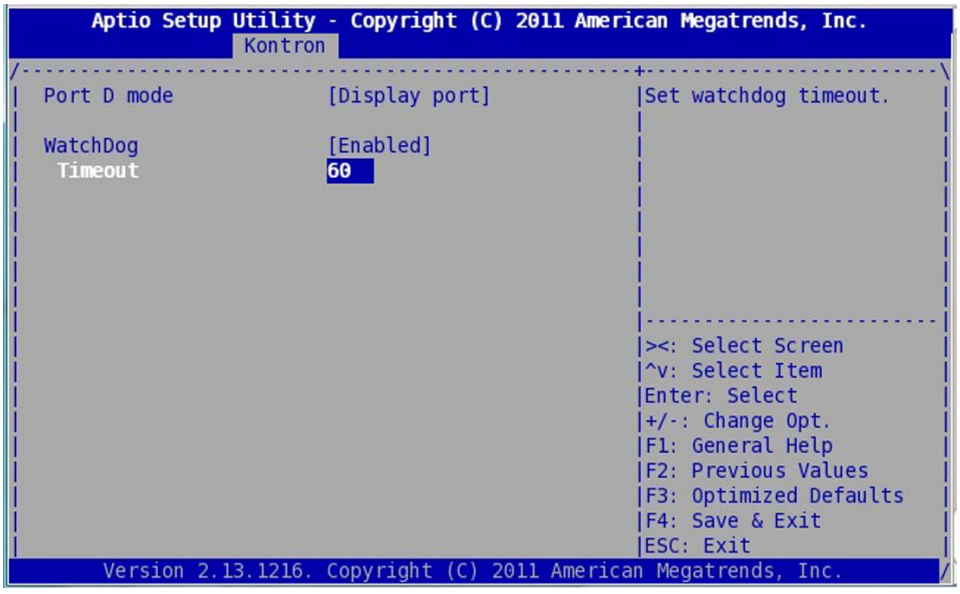
The WatchDog option allows to disable (default setting) or enable the CPLD Watchdog Timer and to define the timeout value.

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 Reset mode is handled.

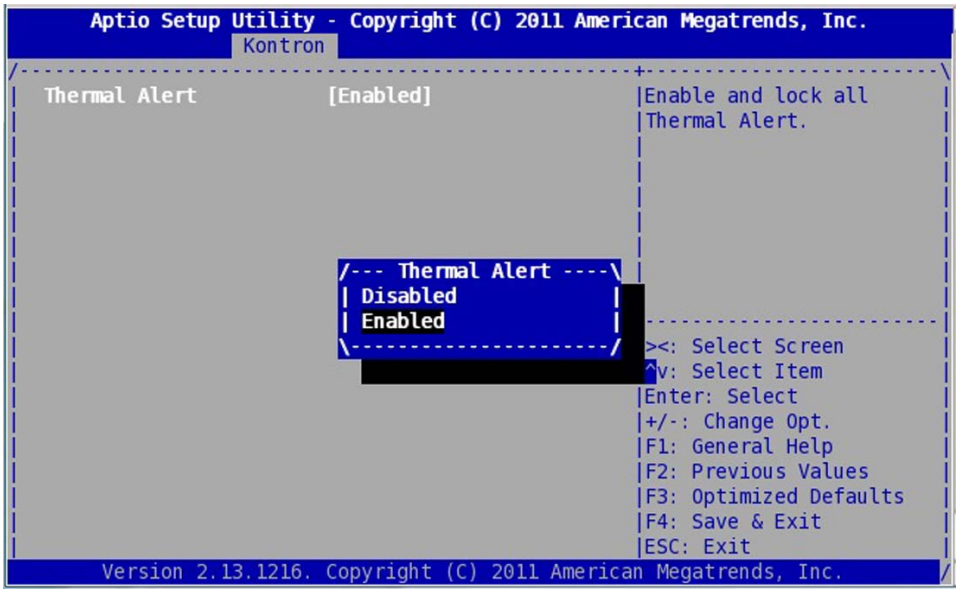
⚠CAUTION The WatchDog setting is kept even after a timeout has occurred.



Since BIOS ID15300 a new feature "**WatchDog at Startup**" appears into the menu and allows to disable (default setting) or enable the CPLD Watchdog Timer at Power-on.

The default timeout is **9 sec** and is not configurable by setup. When set the **Watchdog at Startup** will start only at next Power-On of the board. Only the **Power Mode** mode is handled.

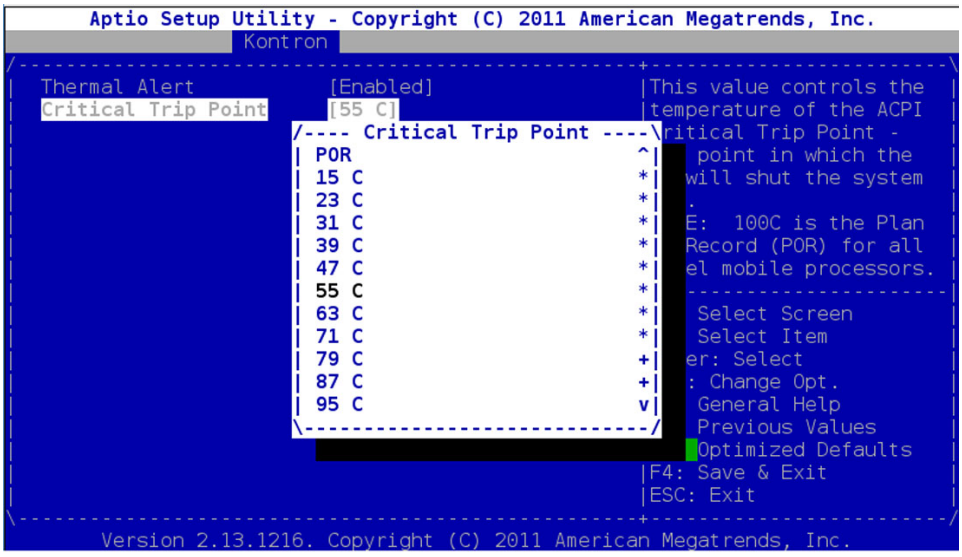
5.11 Thermal Configuration



This menu can be used to enable (default setting) or disable the PCH alert. If enabled, the PCH will signal if its temperature is outside the temperature limits.

It is recommended to keep this option **Enabled**.

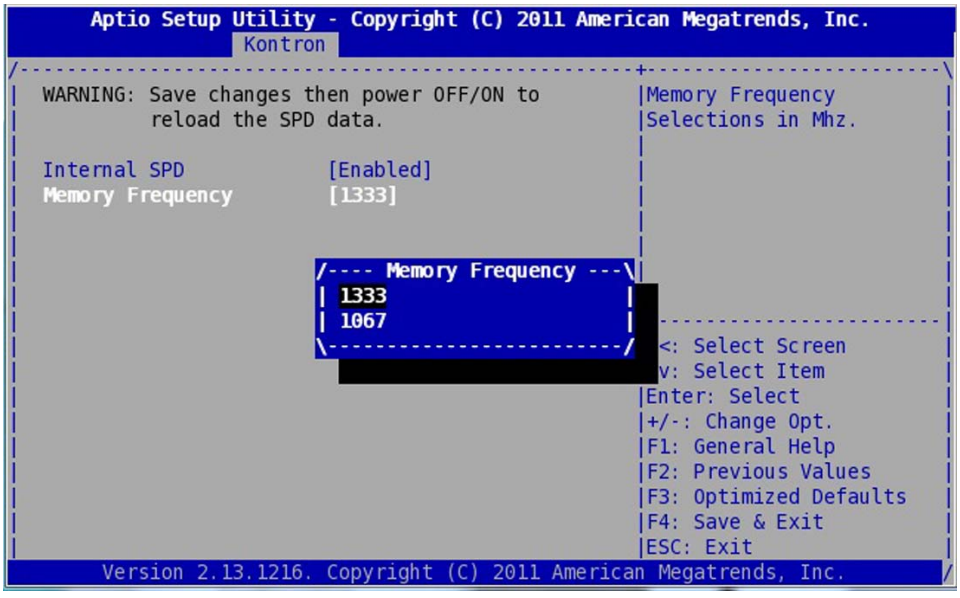
Since BIOS ID15300 it is also possible to select the Thermal Trip Point:



The value controls the temperature of the ACPI Critical Trip POint in which the OS will shutdown the system. Possible values come from **15 C** to **115 C**.

The default value is **POR = 100 C**

5.12 SPD Configuration



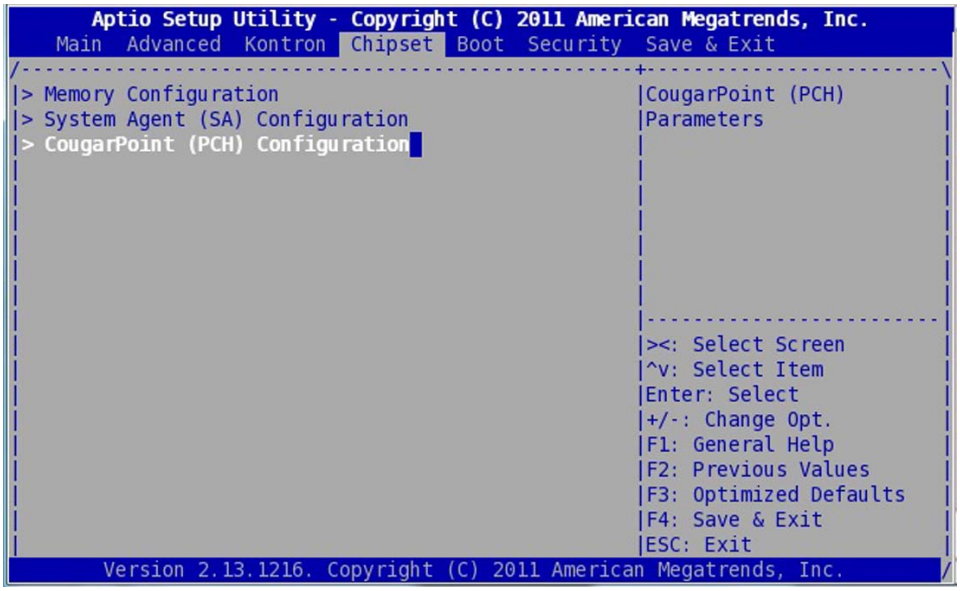
The SPD Configuration menu allows to select either the SPD data from BIOS internal tables or from the SPD EEPROM accessible thru the PCH SMBus (default setting).

The BIOS internal tables are based on an hardware configuration and on the VPD (Vital Product Data).

It is possible also to select the **Memory Frequency**: **1067** or **1333** MHz.

This feature allows to bypass SMBus access on PCH in order to speed up and secure the boot process in case of reset during I2C EEPROM access

6 / Chipset Menu



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

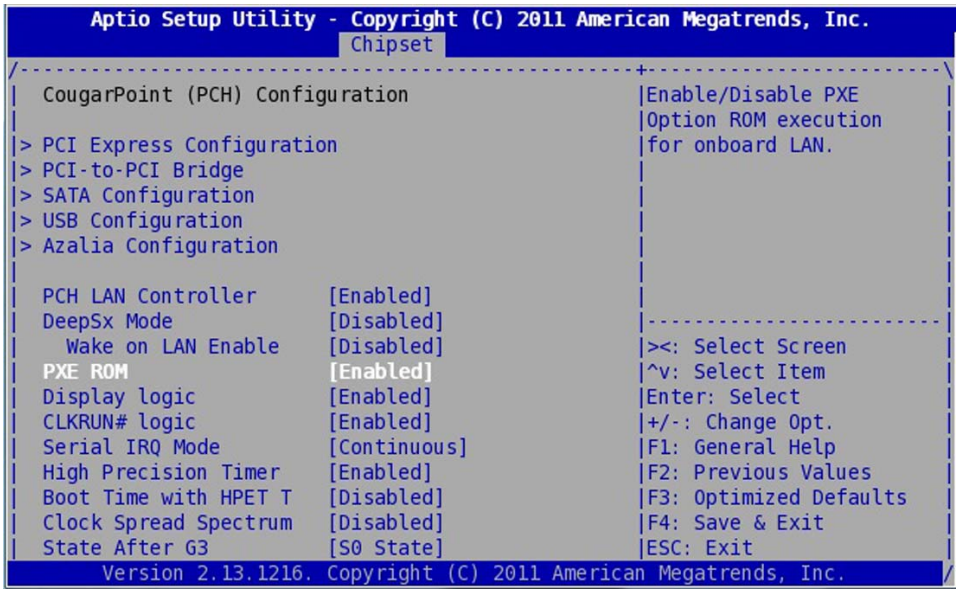
In particular the **CougarPoint (PCH) Configuration** menu will be used to enable the Pre-boot Execution Environment (PXE) ROM and also to manage the SATA Configuration.

6.1 PXE ROM Configuration

Enter the **CougarPoint (PCH) Configuration** menu and select the **PXE ROM** option.

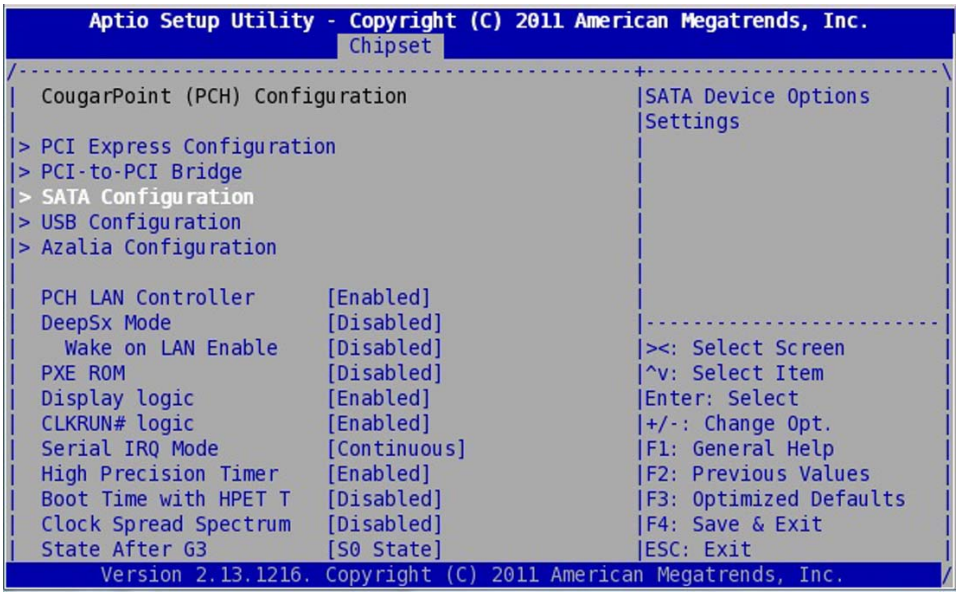
This option allows to enable/disable the PXE ROM.

The other settings are reserved and not intended to be used.



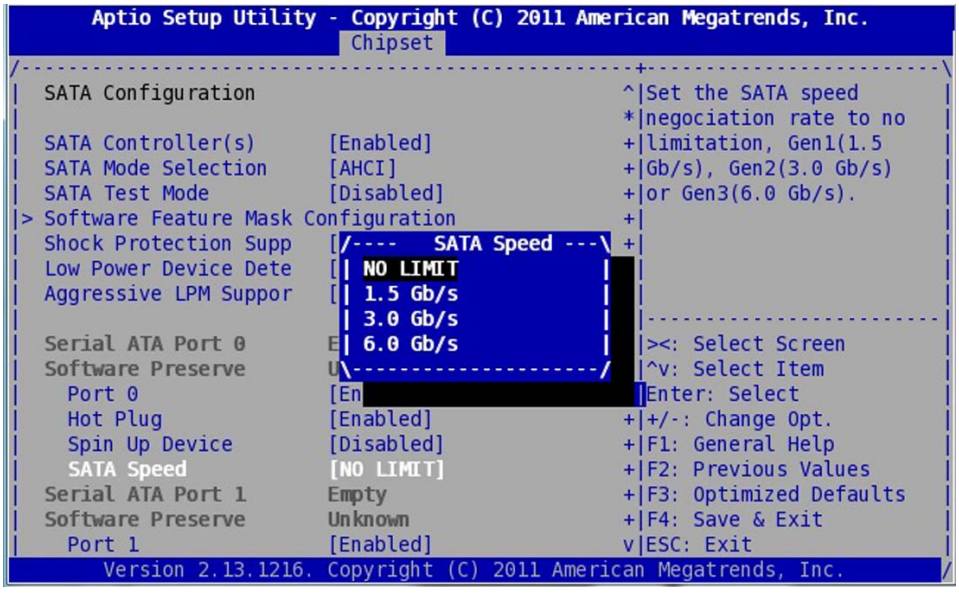
6.2 SATA Configuration

Enter the **CougarPoint (PCH) Configuration** menu and select the **SATA Configuration** menu



By default, the SATA controllers are enabled and the SATA mode is **AHCI**.

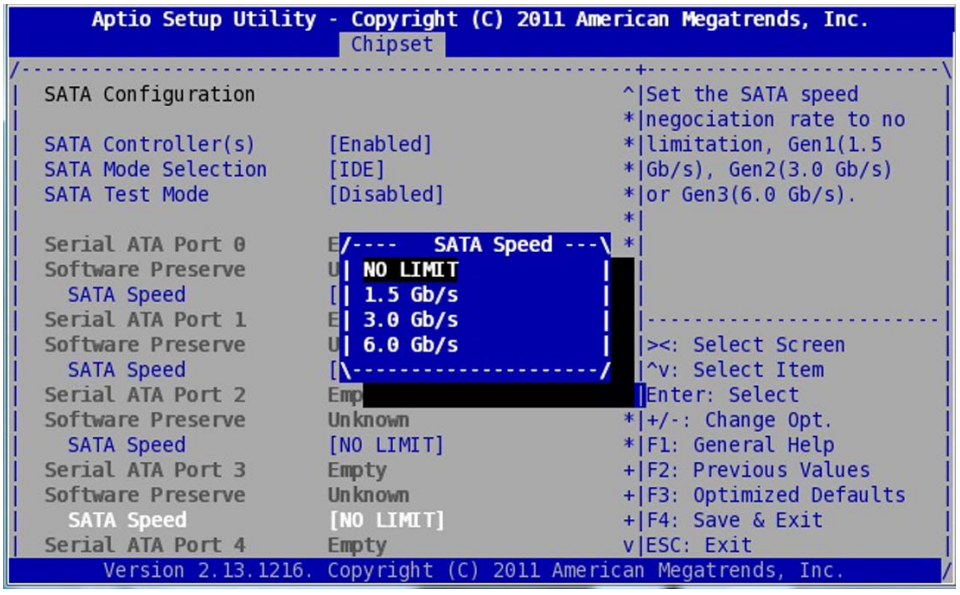
By default, no speed limitation is selected (**NO LIMIT**) for each supported port excepted for SATA port 4 where the SATA flash may be connected. For SATA port 4 the default speed is set to 1.5 Gb/s (Gen1). If **NO LIMIT** is selected, the communication speed is negotiated based on the max speed capability (6 Gb/s).



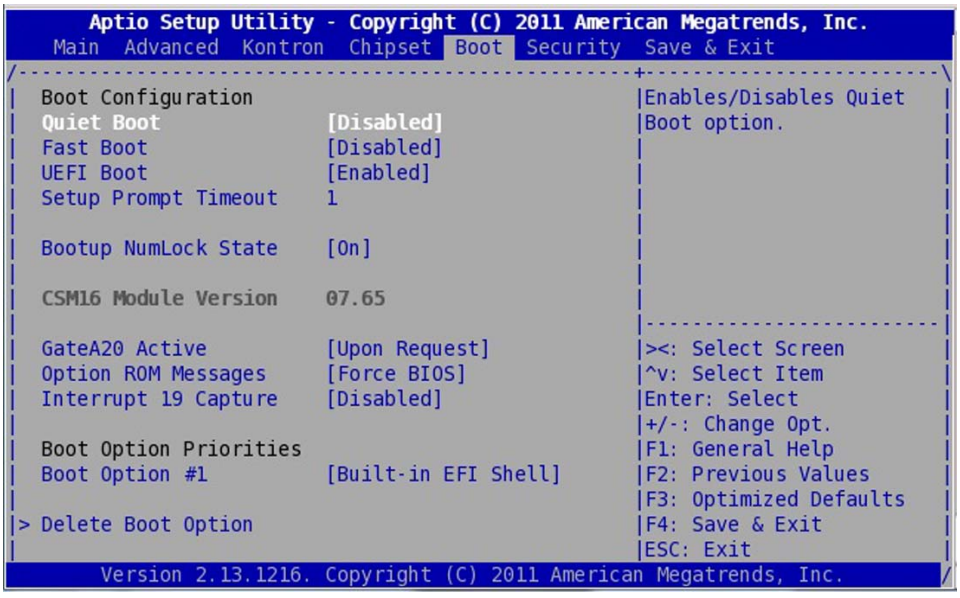
NOTICE

In AHCI mode, the operating system may re-negotiate the SATA speed based on the capabilities registers. When booting Linux, it is possible to force the SATA speed by using the **libata.force** option in the kernel command line.

The following describes the **SATA Configuration** menu when the selected SATA mode is IDE.



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 30
- **UEFI Boot**: Section 7.2 page 30
- **Setup Prompt Timeout**: Section 7.3 page 30
- **Bootup NumLock State**: Section 7.4 page 30
- **Boot Option Priorities**: Section 7.5 page 31
- **Network Device BBS Priorities**: Section 7.6 page 32
- **Hard Drive BBS Priorities**: Section 7.7 page 34
- **Delete Boot Option**: Section 7.8 page 35

Other following submenus are Reserved and Not to be used !

- GateA20 Active
- Option ROM Messages
- Interrupt 19 Capture
- Add New Boot Option

NOTICE

The VX3035 boot time is about 4s after a reset and 7s after a power on, assuming boot time end is when the EFI shell prompt appears.

7.1 Quiet boot

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

```
Version 2.13.1216. Copyright (C) 2011 American Megatrends, Inc.
   BIOS Date: 04/13/2012 15:01:42 Ver: 0ABUC0010
```

Press <DEL> or <F2> to enter setup. Press <F7> for BBS POPUP Menu.

Set **Quiet Boot**

- Disabled
- Enabled

Default is **Disabled**

7.2 UEFI boot

UEFI Boot setting allows to enable or disable UEFI boot from disk

Set **UEFI Boot**

- Disabled
- Enabled

Default is **Enabled**

7.3 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.

7.4 Bootup Numlock State

This menu selects the keyboard numlock state

Set **Bootup NumLock State**

- On
- Off

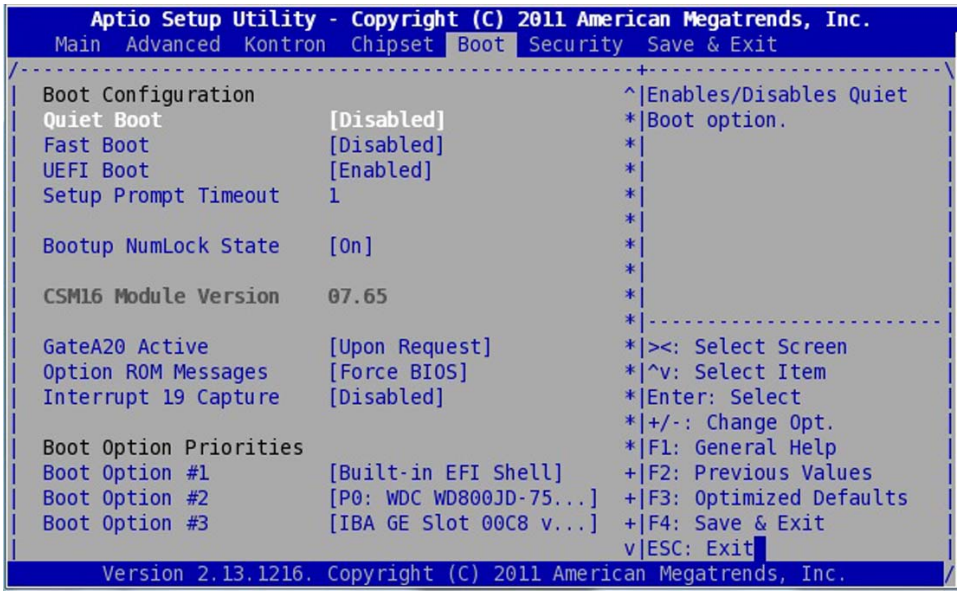
Default is **On**

7.5 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

NOTICE

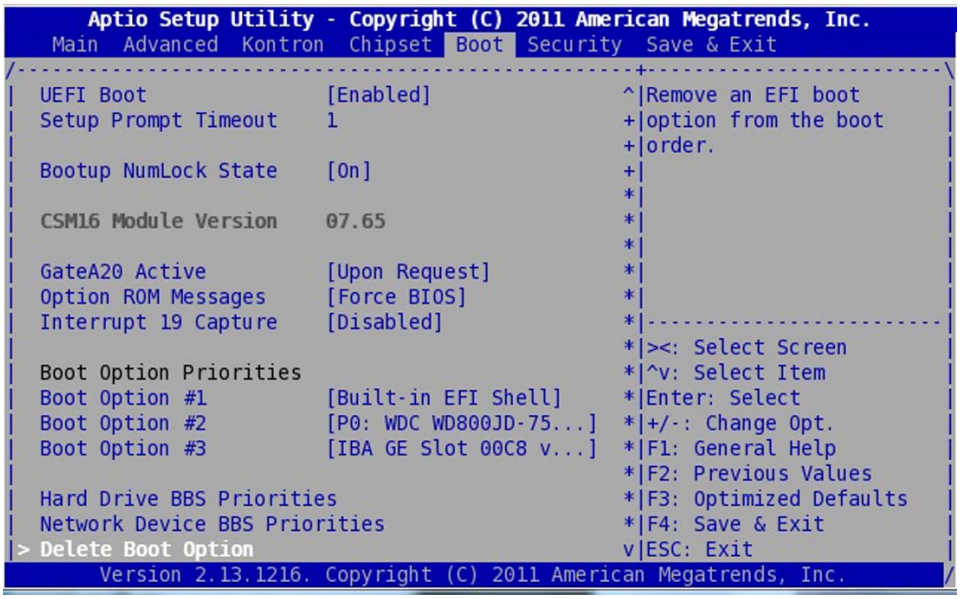
The possible family boot device can be SATA, USB or Gigabit Ethernet (Gbe). In the boot device item list only one item per family will appear. If more than one device is available for booting (for example 2 SATA disk or 3 Ethernets for PXE) then 2 new submenus can appear below the item list. So it can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

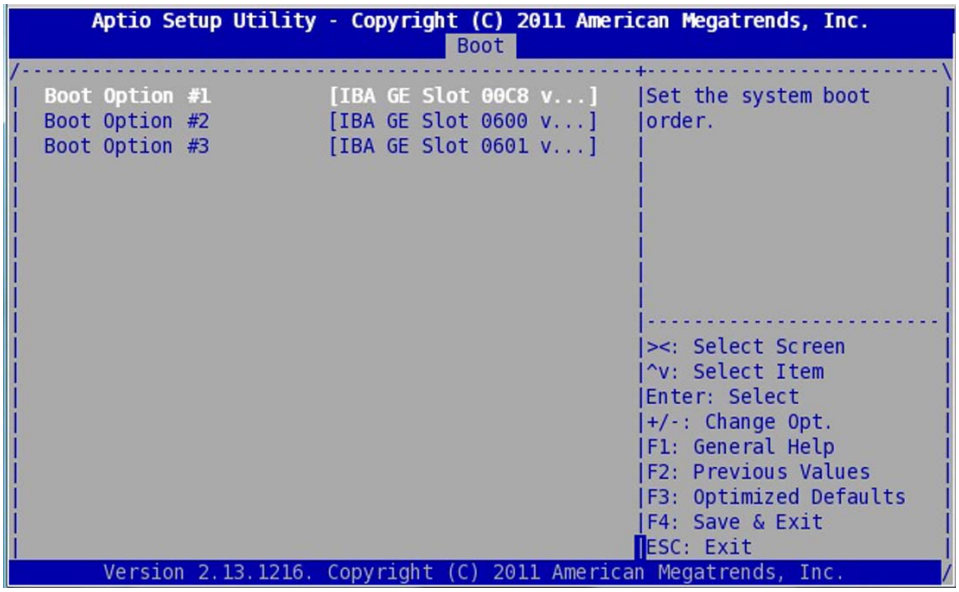
7.6 Network Device BBS Priorities (when PXE ROM Enabled)

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 first Network Device "**IBA GE Slot 00C8**" is related to the Ethernet Interface of the Intel(R) 82579 device.

The Network Devices "**IBA GE Slot 0600**" and "**IBA GE Slot 0601**" are related to the Ethernet Interfaces of the Intel(R) 82580 Dual Port device.

NOTICE

The numeric values "**0600**" and "**0601**" may change for these interfaces depending on the PCI-Express devices connected to the board and so the numbering of the PCI-Express busses.

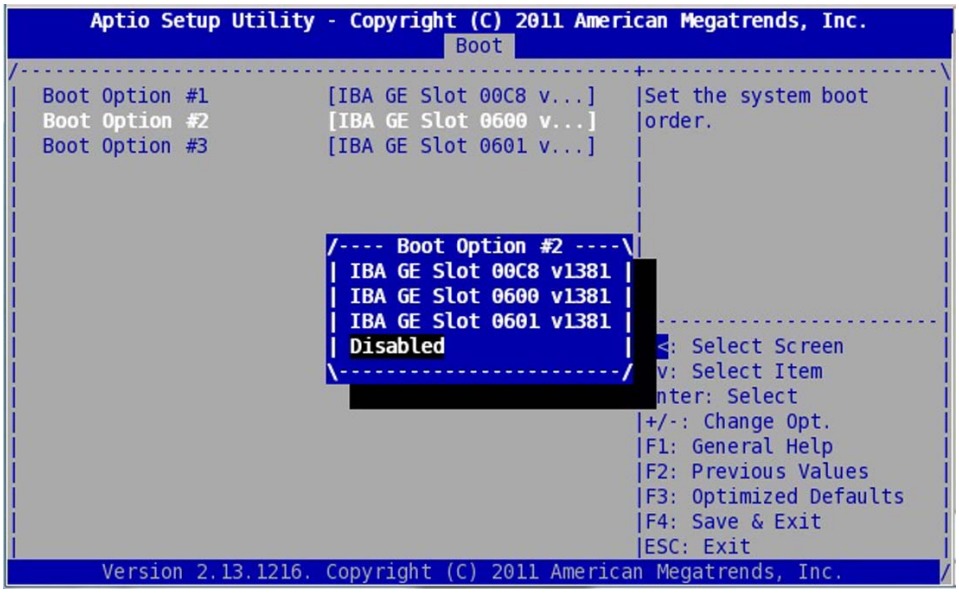
To change the PXE boot device ordering

- 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



NOTICE

When a PXE boot device is disabled this does not disable the PXE OpROm loading for the corresponding boot device. So the following message will appear 3 times in any case when PXE ROM is enabled for South Bridge:

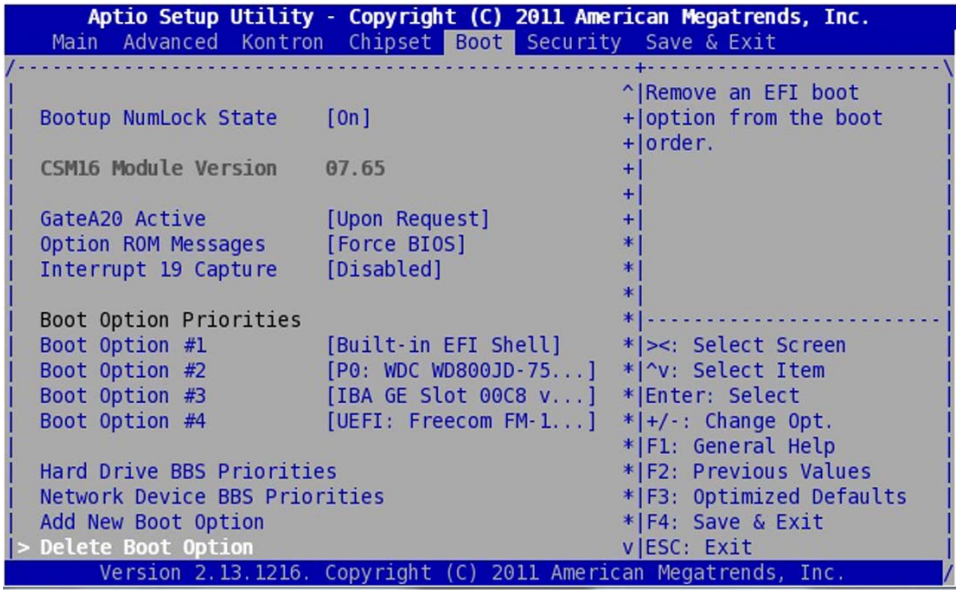
```
Initializing Intel(R) Boot Agent GE v1.3.81
PXE 2.1 Build 091 (WfM 2.0)
```

Press <Ctrl>+<S> to enter the Setup Menu..

7.7 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 available SATA & USB boot device and be able to disable it or to reorganize the boot sequence.

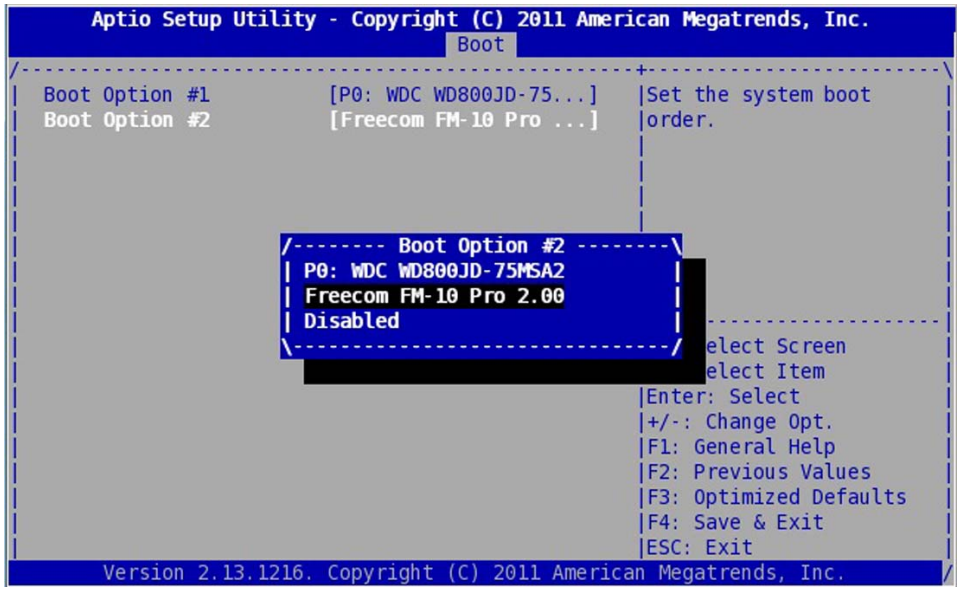
To change the boot device ordering

- 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 in the list

To disable one of the boot device

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

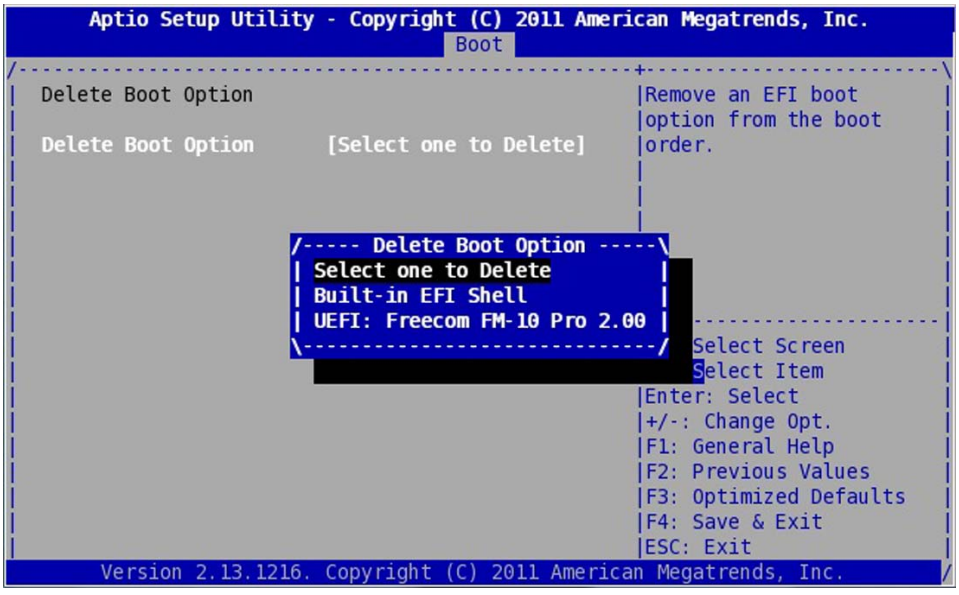
A new submenu appears (see image) , select **Disabled** to disable the SATA or USB device



7.8 Delete Boot Option

The setting allows to delete a boot device from the available boot device list.

In particular Built-In EFI shell can be deleted.



To delete a boot device like EFI Shell

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

8 / Security Menu

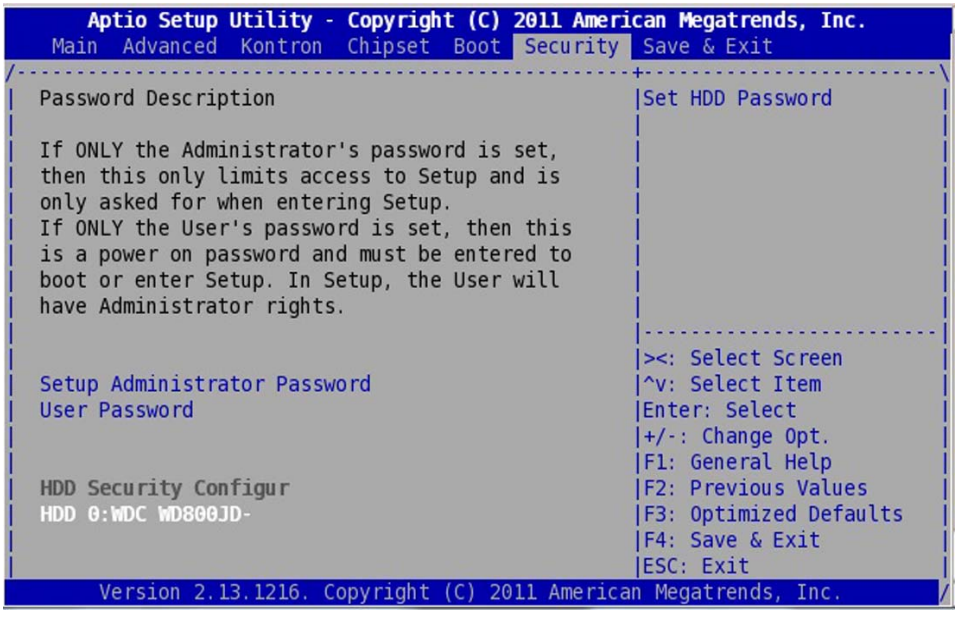
The security Menu allows the user to set a password for SETUP or boot access.

NOTICE

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 appears when a SATA disk is connected.

This submenu is Reserved and Not To Be Used



8.1 Enter Administrator or user password

To enter password:

- Select the administrator or user password item
- A pop-up window appears and proposes to you to create a new password
- Enter a password from 1 to 20 characters
- Confirm password
- Then the new password will be recorded if save change is launch in Save & Exit Menu.

At next reboot if <F2> key is pressed then entering password is mandatory to enter SETUP

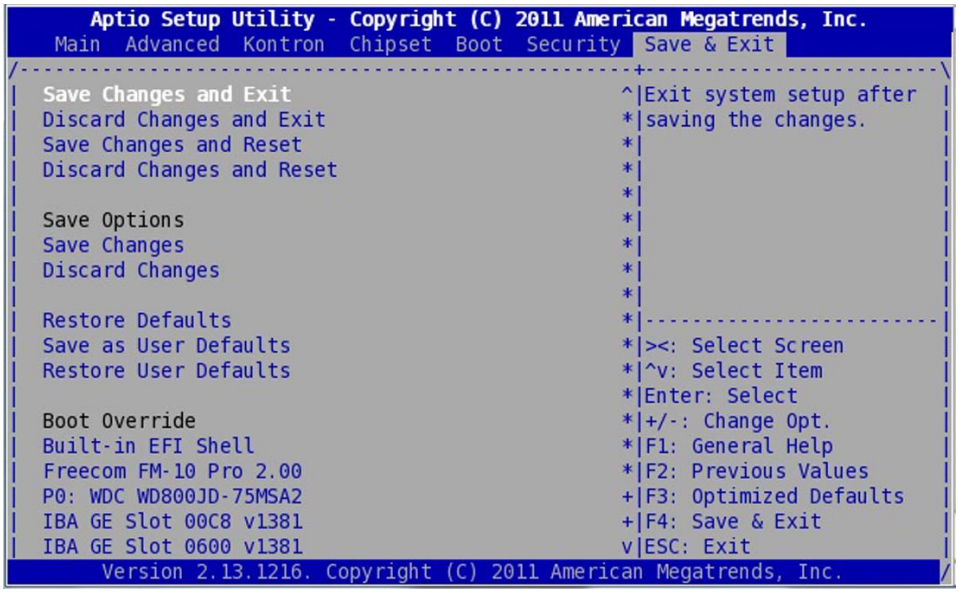
When User password has been set the password will be required to entering SETUP and to to execute the BIOS boot device selection .

To suppress password

- Select the administrator or user password item
- A pop-up window appears and proposes to you to enter a password
- Enter previous password
- A pop-up window appears and proposes to you to enter a new password
- Then type an empty password
- Confirm empty password
- Password will be deleted if save change is launch in Save & Exit Menu.

NOTICE If password is lost the solution to unlock it will be to flash the BIOS or to flash the SETUP BIOS part.

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 39
- **Discard Changes and Exit**: section 9.1 page 39
- **Save Changes and Reset**: section 9.1 page 39
- **Discard Changes and Reset**: section 9.1 page 39
- **Save Changes**: section 9.2 page 40
- **Discard Changes**: section 9.2 page 40
- **Restore Defaults**: section 9.2 page 40
- **Save as User Defaults**: section 9.3 page 40
- **Restore User Defaults**: section 9.3 page 40
- **Boot Override**: section 9.4 page 40

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

Current SETUP configuration can be saved as user configuration and can be restored the same way the default configuration.

Select desired item and <Enter>

- Save as User Defaults
- Restore User Defaults

9.4 Boot Override

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

10 / EFI SHELL

EFI Shell is a boot shell available on the VX3035 that is accessible in the boot device list. EFI Shell is launched automatically if no other boot device is connected to the VX3035. 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 **VX3035> help help**) to have the detail of a command syntax

- VX3035> help

COMMAND NAME	DESCRIPTION	SEE SECTION
?	Displays the EFI Shell command list or verbose command help	10.1.20 page 58
alias	Displays, creates, or deletes EFI Shell aliases	10.1.1 page 43
amlview	AML view utility	10.1.2 page 44
bcfg	Boot configuration utility	10.1.3 page 45
cd	Displays or changes the current directory	10.1.4 page 46
cls	Clears standard output and optionally changes background color	10.1.5 page 47
connect	Connects one or more EFI drivers to a device	10.1.6 page 47
cpuutil	CPU information utility	10.1.7 page 47
date	Displays or changes the current system date	10.1.8 page 48
devices	Displays the list of devices managed by EFI drivers	10.1.9 page 48
dh	Displays EFI handle information	10.1.10 page 49
disconnect	Disconnects one or more EFI drivers from a device	10.1.11 page 51
drvcfg	Invokes the Driver Configuration Protocol	10.1.12 page 51
drivers	Displays the EFI driver list	10.1.13 page 53
dumpacpi	Print ACPI Tables	10.1.14 page 54
dumpaml	Print AML dump	10.1.15 page 54
echo	Controls batch file command echoing or displays a message	10.1.16 page 55
exit	Exits the EFI Shell environment	10.1.17 page 55
for	Executes commands for each item in a set of items	10.1.18 page 56
goto	Forces batch file execution to jump to specified location	10.1.19 page 57
help	Displays the EFI Shell command list or verbose command help	10.1.20 page 58
if	Executes commands in specified conditions	10.1.21 page 59
kdiag	Perform board diagnostics - Available ONLY if ordered.	10.1.22 page 59
kflash	Kontron SPI flasher	10.1.23 page 60
kmac	Kontron MAC Address viewer	10.1.24 page 60
kpld	Kontron PLD Commands	10.1.25 page 61
ktemp	Kontron Board Temperature	10.1.26 page 61
kuuid	Kontron UUID Configurator	10.1.27 page 62
kvpd	Kontron VPD Information	10.1.28 page 62
kvpx	Kontron VPX Configurator	10.1.29 page 63

COMMAND NAME	DESCRIPTION	SEE SECTION
ls	Displays a list of files and subdirectories in a directory	10.1.30 page 64
map	Displays or defines mappings	10.1.31 page 66
mem	Displays the contents of memory	10.1.32 page 69
memmap	Displays the memory map	10.1.33 page 71
mm	Displays or modifies MEM/MMIO/IO/PCI/PCIE address space	10.1.34 page 72
mv	Moves one or more files or directories to another location	10.1.35 page 75
pause	Prints a message and waits for keyboard input	10.1.36 page 76
pci	Displays PCI device list or PCI function configuration space	10.1.37 page 77
reconnect	Reconnects one or more EFI drivers to a device	10.1.38 page 79
reset	Resets the system	10.1.39 page 79
set	Displays or modifies EFI Shell environment variables	10.1.40 page 80
shift	Shifts batch file input parameter positions	10.1.41 page 81
smbiosview	Displays SMBIOS information	10.1.42 page 82
smbutil	SMBus utility	10.1.43 page 83
time	Displays or changes the current system time	10.1.44 page 83

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

NOTICE

1. '**sname**' should not be an internal EFI Shell command.
2. '**value**' can be an internal EFI Shell command, a script, or an EFI application. However, any other values are also acceptable.
3. **ALIAS** values are stored in EFI NVRAM and will be retained between boots unless the '**-v**' option is specified.
4. **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

```
fs0:\> AmlView
Welcome to AmlView on EFI Shell (Version 0.01)
DefinitionBlock ("Dsdt.aml", "DSDT", 1, "ALASKA", "SNB-CPT", 0)
```

```
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     add ´file´ with ´desc´ at position #
  addp    add ´file´ with ´desc´ at position #.Use hard drive path
  addh    add ´handle´ with ´desc´ at position #.Use Handle
  rm      remove #
  mv      move # 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)/C57AD6B7-0515-40A8-9D21-
551652854E37 "Built-in EFI Shell"
02. BBS-Net() "Network Card" OPT
03. Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(2, 0)/HD(Part1,SigBB2FF4E4) "UEFI: SMART eUSB 874D"
04. BBS-Harddrive() "Hard Drive" OPT
05. Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig00A94D6E) "UEFI: CHIPSBNKv3.3.8.8 5.00"
06. Not Found

Shell> bcfg boot mv 4 2
bcfg: boot option 4 moved to 2

Shell> bcfg boot dump
The boot option list is:
01.VenMedia(5023B95C-DB26-429B-A648-BD47664C8012)/C57AD6B7-0515-40A8-9D21-
551652854E37 "Built-in EFI Shell"
02. BBS-Harddrive() "Hard Drive" OPT
03. BBS-Net() "Network Card" OPT
04. Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(2, 0)/HD(Part1,SigBB2FF4E4) "UEFI: SMART eUSB 874D"
05. Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig00A94D6E) "UEFI: CHIPSBNKv3.3.8.8 5.00"
06. Not Found
```

10.1.4 cd

Displays or changes the current directory.

```
CD [path]
```

path	The relative or absolute directory path

NOTICE

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

NOTICE

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

NOTICE

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

NOTICE

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 (MemMap(11:FFF60000-FFFFFFFF))
4: DevPath (MemMap(11:7A626004-7A9E6003))
5:
6: Decompress
7:
8:
9: UnicodeCollation
A:
B: DriverBinding ComponentName
C:
D: Image(Runtime)
E:
F:
10: Image()
11:
12:
13:
14:
15:
16: Image(SaDxePolicyInit)
17: Image(SmmAccessWrap)
(…)

- To display detailed information for handle 0x30:

```
Shell> dh 17
```

Handle 17 (016E1C18)
  Image (16ECE40) File:SmmAccessWrap
    ParentHandle..: 7A9F5F18
    SystemTable...: 7ADB7F18
    DeviceHandle..: 1003418
    FilePath.....: 1323C999-DAD5-4126-A54B-7A05FBF41515
    PdbFileName...: C:\Aptio\Project\REF_VX3035\Build\SmmAccessWrap.pdb
    ImageBase.....: 16EB000 - 16EBD40
    ImageSize.....: D40
    CodeType.....: BS_code
    DataType.....: BS_data

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

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

Handle dump by protocol 'Diskio'
  E7: DiskIo BlkIo UsbIo DevPath (..ci(1D|0)/Usb(1, 0)/Usb(1, 0))
  FB: Fs DiskIo BlkIo DevPath (..(1, 0)/HD(Part1,Sig000B9400))
  EE: DiskIo BlkIo DevPath (Acpi(PNP0A03,0)/Pci(1F|2)/?)
  FC: DiskIo BlkIo DevPath (..F|2)/?/HD(Part1,SigED32B4EF))
  FD: DiskIo BlkIo DevPath (..F|2)/?/HD(Part2,SigED32B4EF))

- 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)
   D: Image(Runtime)
  10: Image()
  16: Image(SaDxePolicyInit)
  17: Image(SmmAccessWrap)
  18: Image(SBRun)
  1A: Image(PciHotPlug)
  1C: Image(SBIDE)
  1E: Image(ActiveBios)
  20: Image(PchReset)
  23: Image(PchSerialGpio)
  25: Image(SmmControl)
  26: Image(WdtDxe)
  27: Image(EcPs2Kbd)
  2A: Image(IdeSMART)
  2C: Image(DtsDxePolicyInit)
  2D: Image(PlatformInfoDxe)
  2E: Image(SmbiosGetFlashData64)
  2F: Image(TcgDxe)
  30: Image(CpuDxe)
  32: Image(SaveMemoryConfigSrc)
  33: Image(PciHostBridge)
  36: Image(ACPIS3Save)

Press <ENTER> to continue, <q> to exit:

10.1.11 disconnect

Reserved - Not To Be Used

10.1.12 drvcfg

Invokes the Driver Configuration Protocol.

```
DRVCFG [-l XXX] [-c] [-f Type|-v|-s]
       [DriverHandle [DeviceHandle [ChildHandle]]]
```

-l	Configures using the specified ISO 639-2 language
-c	Configures all child devices
-f	Forces defaults
-v	Validates options
-s	Sets options
Type	The type of default configuration options to force on the controller specified by ControllerHandle and ChildHandle: **0** - Safe Defaults. **1** - Manufacturing Defaults. **2** - Custom Defaults. **3** - Performance Defaults.
DriverHandle	Handle of the driver to configure
DeviceHandle	Handle of a device that DriverHandle is managing
ChildHandle	Handle of a device that is a child of DeviceHandle

NOTICE

Default Type.

1. Safe Defaults. Places a controller in a safe configuration with the greatest probability of functioning correctly in a platform.
2. Manufacturing Defaults. Optional type that places the controller in a configuration suitable for a manufacturing and test environment.
3. Custom Defaults. Optional type that places the controller in a custom configuration.
4. Performance Defaults. Optional type that places the controller in a configuration that maximizes the controller's performance in a platform.

Other Value - Depends on the driver's implementation.

- **Examples:**
  - To display the list of devices available for configuration:

```
Shell> drvcfg
```

  - To display the list of devices and child devices available for configuration:

```
Shell> drvcfg -c
```

  - To force defaults on all devices:

```
Shell> drvcfg -f 0
```

  - To force defaults on all devices managed by driver 0x17:

```
Shell> drvcfg -f 0 17
```

- To force defaults on device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -f 0 17 28
```

- To force defaults on all child devices of device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -f 0 17 28 -c
```

- To force defaults on child device 0x30 of device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -f 0 17 28 30
```

- To validate options on all devices:

```
Shell> drvcfg -v
```

- To validate options on all devices managed by driver 0x17:

```
Shell> drvcfg -v 17
```

- To validate options on device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -v 17 28
```

- To validate options on all child devices of device 0x28 which are managed by driver 0x17:

```
Shell> drvcfg -v 17 28 -c
```

- To validate options on child device 0x30 of device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -v 17 28 30
```

- To set options on device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -s 17 28
```

- To set options on child device 0x30 of device 0x28 which is managed by driver 0x17:

```
Shell> drvcfg -s 17 28 30
```

- To set options on device 0x28 which is managed by driver 0x17, in English:

```
Shell> drvcfg -s 17 28 -l eng
```

- To set options on device 0x28 which is managed by driver 0x17, in Spanish:

```
Shell> drvcfg -s 17 28 -l spa
```

10.1.13 drivers

Displays the EFI driver 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

- **Examples:**
  - 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
== ======== = = = == == ================================== ==================
0B 00000010 B - -  1  2 AMI Generic LPC Super I/O Driver    CORE_DXE
75 00010000 D - -  1  - AMI File System Driver              FileSystem
77 00020200 B - -  1 22 <UNKNOWN>                           PciBus
88 00000010 D - -  1  - PCH Serial ATA Controller Initializ SataController
8A 00000001 B - -  1  1 AMI AHCI BUS Driver                 AHCI
8C 00000010 ? - -  -  - <UNKNOWN>                           BIOSBLKIO
8D 00000024 B - -  1  1 BIOS[INT10] Video Driver            CsmVideo
8E 00000010 ? - -  -  - <UNKNOWN>                           <UNKNOWN>
9D 00000010 B - -  2  2 AMI Serial I/O Driver               Terminal
9E 00000010 B - -  1  1 AMI Terminal Driver                 Terminal
9F 0000008A D - -  2  - AMI USB Driver                      UHCD
A1 0000008A B - -  2  3 USB bus                             UHCD
A2 00000001 ? - -  -  - USB Keyboard driver                 UHCD
A3 00000002 ? - -  -  - USB Mouse driver                    UHCD
A4 00000001 D - -  1  - USB Mass Storage driver             UHCD
A5 04080100 B X X  3  3 Intel(R) PRO/1000 4.8.01 PCI-E      IntelGigabitLanx64
C4 00000010 D - -  5  - <UNKNOWN>                           CORE_DXE
C5 00000010 D - -  1  - <UNKNOWN>                           CORE_DXE
C6 00000010 B - -  3  3 <UNKNOWN>                           CORE_DXE
C8 00000010 B - -  2  3 <UNKNOWN>                           CORE_DXE
C9 00000010 ? - -  -  - AMI PS/2 Driver                     CORE_DXE
CA 00000010 ? - -  -  - AMI Floppy Driver                   CORE_DXE
CB 00000001 ? - -  -  - AMI IDE BUS Driver                  CORE_DXE
```

10.1.14 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.

10.1.15 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)

10.1.16 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

NOTICE

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
```

10.1.17 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.

10.1.18 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

NOTICE

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
```

10.1.19 goto

Forces batch file execution to unconditionally jump to specified location.

```
GOTO label
```

`label`	Specifies a location in batch file

NOTICE

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.

- Examples:

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

10.1.20 help

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

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

cmd	Shell command name
pattern	Wildmatch pattern

NOTICE

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
```

10.1.21 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

NOTICE

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.

- Examples:

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

10.1.22 kdiag

Performs board diagnostics. Available ONLY if ordered.

10.1.23 kflash

Kontron SPI flasher

Usage:

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

- Operation mode
  - `-p` Program 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
  - `-cs` Program CRC into Current Bios Flash. CRC will be checked at boot time if Flash Write Protect or NVMRO is set.

10.1.24 kmac

Kontron MAC Address utility

Usage:

```
kmac [-h|-r|-dump] [-w value] [-save|-load [filename]] [-prog]
```

- Operation mode
  - `-h` Shows this help
  - `-r | --read` Shows MAC Addresses (82579 and 82580 chipsets)
  - `-w | --write value` Updates MAC Address for 82579 chipset and set adjacent addresses for 82580 chipset
    value format = 0x0000DEaabbcc
  - `-prog` Programs the 82580 EEPROM with a predefined image (only dual link supported yet)
  - `-dump` Dumps the first 1024 words of the 82580 EEPROM
  - `-save filename` Saves the 82580 EEPROM contents to <filename>
  - `-load filename` Loads the 82580 EEPROM with the contents of <filename>
- Example

```
Shell> kmac -r

Quad link Gbe 82580 forced in dual mode configuration
MAC Address of Intel 82579 = 00:00:DE:40:41:4D
MAC Address of Intel 82580 LAN0 = 00:00:DE:40:41:4E
MAC Address of Intel 82580 LAN1 = 00:00:DE:40:41:4F
```

10.1.25 kpld

Kontron PLD Commands: this command allows basics accesses to internal PLD registers and I2C device (EEPROM, Thermal sensors)

Usage:

```
kpld [ -h|-? ]
```

- Operation mode
  - **-h** Shows this help
  - **-v** Shows cpld revision
  - **-m** Memory information protection -r : Read 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
  - **-i2cw** Writes Access to I2C bus
    -> kpld -i2cw busNum Add Offset Type Data

10.1.26 ktemp

Usage:

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

- Operation mode
  - **-h** Shows this help
  - **-p** Prints PCH temperature
- **Example:**

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

+-------------------------+-------------+
| CPU Temperature  |    55 C     |
+-------------------------+-------------+
| PKG Temperature  |    57 C     |
+-------------------------+-------------+
| PCH Temperature  |    68 C     |
+-------------------------+-------------+
```

10.1.27 kuuid

Kontron UUID configurator: this command allows user to change the default UUID value of the board and overcome the value set on the setup (See section 5.4 page 17).

Usage:

```
kuuid [ -a|-r|-p|-h ]
```

- Operation mode

  `-a | --ascii` Stores UUID in ASCII format
  `-r | --raw` Stores UUID in RAW format
  `-p | --print` Prints UUID
  `-h | --help` Shows this help

- **Example:**

```
VX3035> kuuid -r
Enter UUID[15-8]:0000000000000000
Enter UUID[7-0]:0000000000000000
Current UUID: 0000000000000001
New UUID: 00000000000000000000000000000000
Is this correct ?
[n] No (re-enter UUID)
[y] Yes
[q] Exit no change
y
VX3035> kuuid -p
Current UUID (RAW) : 00000000-0000-0000-0000000000000000
VX3035> reset
```

NOTICE It is mandatory to perform a reset at the end of the process to update UUID in SMBIOS table.

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


Current configuration:
              Order Code     : VX3035-SA24-01000
              EC Level       : EC10000
              Serial Number  : 1811361040007
              Variant        : 0184304001000008
              Check Sum      : 0C566DB9
```

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: Manage PCIe switch PEX8609 Serial EEPROM

Parameter list:

prog: Program PCIe switch serial EEPROM

dump: Dump PCIe switch serial EEPROM

Options:

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

- **Example:**

```
Shell> kvpx -plx_eeprom prog
Warning: EEPROM contents not loaded by PEX, may be currently in transparent mode
Writing Backplane PEX8609 serial EEPROM OK
Shell> kvpx -plx_eeprom dump
Warning: EEPROM contents not loaded by PEX, may be currently in transparent mode
@0x0000 = 0x5A008400
@0x0004 = 0x77008000
@0x0008 = 0x34317700
@0x000C = 0x00003431
@0x0010 = 0x8F00809C
@0x0014 = 0x14003504
(...)
```

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)

NOTICE

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 ':'

NOTICE

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 hd22b0b0b blk0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk0 :Removable HardDisk - Alias hd22b0b0b fs0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk1 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part1,SigED32B4EF)
  blk2 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part2,SigED32B4EF)
  blk3 :BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?
  blk4 :Removable BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)
  hd22b0b0b :Removable HardDisk - Alias fs0 blk0
             Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To display all mappings in the device mapping table:

```
Shell> map
Device mapping table
  fs0  :Removable HardDisk - Alias hd22b0b0b blk0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk0 :Removable HardDisk - Alias hd22b0b0b fs0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk1 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part1,SigED32B4EF)
  blk2 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part2,SigED32B4EF)
  blk3 :BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?
  blk4 :Removable BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)
  hd22b0b0b :Removable HardDisk - Alias fs0 blk0
             Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To display verbose mapping table information:

```
Shell> map -v
Device mapping table
  fs0   Consistent Name hd22b0b0b
        Other Name    blk0
        Handle        F4: Fs DiskIo BlkIo
        Media Type    HardDisk
        Removable     YES
        Current Dir   \
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk0  Consistent Name hd22b0b0b
        Other Name    fs0
        Handle        F4: Fs DiskIo BlkIo
        Media Type    HardDisk
        Removable     YES
        Current Dir   \
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk1  Consistent Name (null)
        Other Name    (null)
        Handle        F5: DiskIo BlkIo
        Media Type    HardDisk
        Removable     NO
        Current Dir   \
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part1,SigED32B4EF)
  blk2  Consistent Name (null)
        Other Name    (null)
        Handle        F6: DiskIo BlkIo
        Media Type    HardDisk
        Removable     NO
        Current Dir   \
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part2,SigED32B4EF) blk3  Consistent Name (null)
        Other Name    (null)
        Handle        EE: DiskIo BlkIo
        Media Type    BlockDevice
        Removable     NO
        Current Dir   \
        Acpi(PNP0A03,0)/Pci(1F|2)/?
```

```
blk4  Consistent Name (null)
      Other Name   (null)
      Handle       E7: DiskIo BlkIo UsbIo
      Media Type   BlockDevice
      Removable   YES
      Current Dir  \
      Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)
hd22b0b0b  Consistent Name hd22b0b0b
           Other Name   fs0 blk0
           Handle       F4: Fs DiskIo BlkIo
           Media Type   HardDisk
           Removable   YES
           Current Dir  \
           Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To assign fs0 another name:

```
Shell> map floppy fs0:
Device mapping table
  floppy :Removable HardDisk - Alias hd22b0b0b fs0 blk0
          Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To display information about the mapped name:

```
Shell> map floppy
Device mapping table
  floppy :Removable HardDisk - Alias hd22b0b0b fs0 blk0
          Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To operate with the mapped name:

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

- To delete a mapped name:

```
Shell> map -d  floppy
Shell> map
Device mapping table
  fs0  :Removable HardDisk - Alias hd22b0b0b blk0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk0 :Removable HardDisk - Alias hd22b0b0b fs0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk1 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part1,SigED32B4EF)
  blk2 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part2,SigED32B4EF)
  blk3 :BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?
  blk4 :Removable BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)
  hd22b0b0b :Removable HardDisk - Alias fs0 blk0
             Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
```

- To display all the mapped names starting with **'b'**:

```
Shell> map b*
Device mapping table
  blk0 :Removable HardDisk - Alias hd22b0b0b fs0
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)/HD(Part1,Sig000B9400)
  blk1 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part1,SigED32B4EF)
  blk2 :HardDisk - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?/HD(Part2,SigED32B4EF)
  blk3 :BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1F|2)/?
  blk4 :Removable BlockDevice - Alias (null)
        Acpi(PNP0A03,0)/Pci(1D|0)/Usb(1, 0)/Usb(1, 0)
```

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

NOTICE

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

NOTICE

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

- Examples:
  - To display the system memory map:

```
VX3035> 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  0000000100000000-000000001005FFFFF  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

NOTICE

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 mv

Moves one or more files or directories to another location.

```
MV src [src...] [dst]
```

src	Source file/directory name (wildcards are permitted)
dst	Destination file/directory name (wildcards not permitted)

NOTICE

1. If the '**dst**' parameter is not specified, the current directory is assumed to be the destination.
2. If there is more than one argument in the command line, the last one will be taken as '**dst**' unconditionally. If there is more than one source file or directory to move, the '**dst**' should be an existing directory.
3. Attempting to move a read-only file or directory is not allowed.
4. Moving a directory that contains read-only file(s) is allowed.
5. You cannot move a directory into itself or its subdirectories.
6. You cannot move a directory if the current directory is itself or its subdirectory.
7. Redirecting output to a file under a directory to be moved is not allowed.
8. If an error occurs, the remaining files or directories will still be moved.

- Examples:
  - To rename a file:

```
fs0:\> mv IsaBus.efi Bus.efi
moving fs0:\IsaBus.efi -> \Bus.efi
 - [ok]
```

  - To move a directory to the current directory:

```
fs0:\> mkdir test1\temp
fs0:\> mv test1\temp
moving fs0:\test1\temp -> \.\temp
 - [ok]
```

  - To rename a directory:

```
fs0:\> mv efi efi1.1
moving fs0:\efi -> \efi1.1
 - [ok]
```

  - To move multiple directories at a time:

```
fs0:\> mv test1 test2 test
moving fs0:\test1 -> \test\test1
 - [ok]
moving fs0:\test2 -> \test\test2
 - [ok]
```

  - Moving a read-only directory will result a failure:

```
fs0:\test> attrib +r temp1
DA  R fs0:\test\temp1
fs0:\test> mv temp1 temp2
moving fs0:\test\temp1 -> \test\temp2
 - [error] - Write Protected
```

10.1.36 pause

Prints a message and waits for keyboard input.

```
PAUSE [-q]
```

`-q` Does not display notification message

NOTICE

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:\>
```

10.1.37 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

NOTICE

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 VX3035:
  - To display all PCI devices in the system:

```
VX3035> pci
   Seg  Bus  Dev  Func
   ---  ---  ---  ----
    00   00   00    00 ==> Bridge Device - Host/PCI bridge
              Vendor 8086 Device 0104 Prog Interface 0
    00   00   01    00 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 0101 Prog Interface 0
    00   00   01    01 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 0105 Prog Interface 0
    00   00   02    00 ==> Display Controller - VGA/8514 controller
              Vendor 8086 Device 0116 Prog Interface 0
    00   00   16    00 ==> Simple Communications Controllers - Other communicati
              Vendor 8086 Device 1C3A Prog Interface 0
    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 1C2D Prog Interface 20
    00   00   1C    00 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 1C10 Prog Interface 0
    00   00   1C    04 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 1C18 Prog Interface 0
    00   00   1C    07 ==> Bridge Device - PCI/PCI bridge
              Vendor 8086 Device 1C1E Prog Interface 0
    00   00   1D    00 ==> Serial Bus Controllers - USB
              Vendor 8086 Device 1C26 Prog Interface 20
    00   00   1F    00 ==> Bridge Device - PCI/ISA bridge
              Vendor 8086 Device 1C4F Prog Interface 0
    00   00   1F    02 ==> Mass Storage Controller - UNDEFINED
              Vendor 8086 Device 1C03 Prog Interface 1
```

```
   00   00   1F    03 ==> Serial Bus Controllers - System Management Bus
             Vendor 8086 Device 1C22 Prog Interface 0
    00   00   1F    06 ==> Data Acquisition & Signal Processing Controllers - Ot
             Vendor 8086 Device 1C24 Prog Interface 0
    00   02   00    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8609 Prog Interface 0
    00   02   00    01 ==> Base System Peripherals - Other system peripheral
             Vendor 10B5 Device 8609 Prog Interface 0
    00   03   01    00 ==> Bridge Device - PCI/PCI bridge
             Vendor 10B5 Device 8609 Prog Interface 0
    00   06   00    00 ==> Network Controller - Ethernet controller
             Vendor 8086 Device 1510 Prog Interface 0
    00   06   00    01 ==> Network Controller - Ethernet controller
             Vendor 8086 Device 1510 Prog Interface 0
```

- To display the configuration space of Bus 0, Device 16, Function 0:

```
VX3035> pci 0 16 0 -i
  PCI Segment 00 Bus 00 Device 16 Func 00 [EFI 0000160000]
  00000000: 86 80 3A 1C 00 00 18 00-04 00 80 07 00 00 80 00  *..:.............*
  00000010: 04 70 F2 F7 00 00 00 00-00 00 00 00 00 00 00 00  *.p..............*
  00000020: 00 00 00 00 00 00 00 00-00 00 00 00 86 80 99 19  *................*
  00000030: 00 00 00 00 50 00 00 00-00 00 00 00 00 01 00 00  *....P...........*

  00000040: D5 41 00 00 20 00 01 80-06 00 17 16 00 00 00 00  *.A.. ...........*
  00000050: 01 8C 03 C8 08 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000060: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000070: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  00000080: 00 00 00 00 00 00 00 00-00 00 00 00 05 00 80 00  *................*
  00000090: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000A0: 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  *................*
  000000B0: 00 00 00 00 00 00 00 00-00 00 00 00 02 00 00 C0  *................*
  000000C0: 3D E3 B2 A6 9D 9B D8 C4-B1 5C 64 89 47 82 44 DD  *=........\d.G.D.*
  000000D0: FF A3 E7 48 50 CF 54 6E-58 76 A5 CB 33 DA B6 6E  *...HP.TnXv..3..n*
  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): 8086                    Device ID(2): 1C3A

Command(4): 0000
  (00)I/O space access enabled:       0  (01)Memory space access enabled:    0
  (02)Behave as bus master:           0  (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): 0018
  (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):     04                BIST(0F):  Incapable
Cache Line Size(C): 00                Latency Timer(D): 00
Header Type(0E):    80, Multi-function, PCI device
Class: Simple Communications Controllers - Other communication device -
Base Address Registers(10):
     Start_Address  Type  Space    Prefetchable?     Size             Limit
   ------------------------------------------------------------------------
  00000000F7F27000  Mem   64 bits  No     0000000000000010  00000000F7F2700F
   ------------------------------------------------------------------------
Expansion ROM Disabled(30)

Cardbus CIS ptr(28):   00000000
Sub VendorID(2C):          8086      Subsystem ID(2E):      1999
Capabilities Ptr(34):        50
Interrupt Line(3C):          00      Interrupt Pin(3D):       01
Min_Gnt(3E):                 00      Max_Lat(3F):             00
```

10.1.38 reconnect

Reserved - Not To Be Used

10.1.39 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

NOTICE

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.

10.1.40set

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

NOTICE

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
```

10.1.41 shift

Shifts batch file input parameter positions.

```
SHIFT
```

NOTICE

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

10.1.42 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

NOTICE

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.

10.1.43 smbutil

EFI SMBUS Utility . NOT RECOMMENDED

Usage:

```
smbutil /rspd [/low]
             smbutil /wspd [/low]
             smbutil /rdbyte Address Length Command
             smbutil /rdword Address Length Command
             smbutil /rdblock Address Length Command
             smbutil /wtbyte Address Length Command /o Filename
             smbutil /wtword Address Length Command /o FileName
             smbutil /wtblock Address Length Command /o FileName
             smbutil /testrw Address Length Command /o TestFileName
              Address, Length, Command in HEX
```

Address	is the device address on SMBUS
Length	is the amount of data to transfer
Command	is the offset to reach into the device

NOTICE

w* commands will change the EEPROM contents of the device.
They are not RECOMMENDED and can cause a malfunction of the board.

NOTICE

testrw can corrupt the EEPROM contents of the device and can cause a malfunction of the board.

10.1.44 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**

NOTICE

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

10.2 Environment Variables

EFI shell allows user to set environment variables.

Three environment variables are available on VX3035 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:

```
VX3035> set bootcmd "pci"
```

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

```
VX3035> set
bootcmd: pci
```

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

```
VX3035> 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:

```
VX3035> set StartupAuto 1
```

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

```
VX3035> 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:

```
VX3035> set StartupDelay 2
```

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

```
VX3035> set -d StartupDelay
```

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

11 / BIOS Versions Description

11.1 Recommendations and Known Limitations

1. Reserved Setup settings

⚠CAUTION 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 fresh Cold Boot 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 "`-sp`" option of the "`kflash`" command is used to preserve the BIOS parameters. However the boot devices order is not preserved by this option.

7. PEG1 Configuration

DO NOT CHANGE the PEG1 configuration setting in the Chipset / System Agent (SA) Configuration menu. It must be set to "Auto" as the speed is defined by the hardware thru a micro switch on board.

8. SATA speed in AHCI mode

Each SATA port can have its speed configured by the BIOS (refer to section 6.2 page 27). However, in AHCI mode, the operating system usually re-negotiate the SATA speed based on the capabilities registers configuration. When booting Linux, it is then possible to force the SATA speed again by using the `libata.force` option in the kernel command line. In IDE mode, SATA speed configured by the BIOS is kept.

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									
			12104	12174	12235	12297	12347	13010	13127	13245	15084	15300
1	4021	HEST and ERST ACPI Tables not supported	X	X	X	X	X	X	X	X	X	X
2	4022	EFI command "kvpx" does not work	X									
3	4035	Default BIOS settings are loaded if CMOS bad	X									
4	4036	Under shell, the top of memory > 4GB is uncacheable	X									
5	4037	USB activity during PBIT memory tests may block the board	X	X	X	X	X	X	X	X	X	X
6	4078	Garbage characters send on COM1 at poweron if hyper-threading is disabled	X	X	X	X						
7	4079	If the primary display is set to PEG in setup then it is reconfigured to AUTO after reset	X	X	X	X						
8	4087	Ethernet errors on i82579 interface (eth0)	X	X	X	X	X					
9	4088	kmac not compatible with eeupdate	X	X	X	X	X					
10	4133	L1, L2 or L3 cache sizes stored in the SMBIOS table may be wrong	X	X	X	X	X	X				
11	4134	Wrong memory information in SMBIOS table 17	X	X	X	X	X	X				
12	4178	Advanced Encryption Standard must be disabled	X	X	X	X	X	X	X			
13	4257	Lost send packet on 82579	X	X	X	X	X	X	X	X		
14	4231	`kuuid` command erases some setup parameters	X	X	X	X	X	X	X	X	X	
15	4295	Security password does not protect SETUP	X	X	X	X	X	X	X	X	X	

11.2.2 Detailed description of the problems

Item #1 HEST and ERST ACPI Tables not supported – CRP 4021

Description: The ACPI Hardware Error Source Table and the ACPI Error Record Serialization Table are not supported.

Hence, error messages appear under Linux at boot time.

Workaround: None

Item #2 EFI command “kvpx” does not work – CRP 4022

Description: The Kontron VPX Configurator EFI command “**kvpx**” is not operational

Workaround: None

Item #3 Default BIOS settings are loaded if CMOS bad - CRP 4035

Description: The default BIOS settings are reloaded automatically at BIOS boot time if the CMOS RAM is corrupted. The user setup is lost.

This is the case if the board is not equipped with any battery or in case of battery failure.

Workaround: None

Item #4 Under shell, the top of memory > 4GB is uncacheable - CRP 4036

Description: Under the EFI shell environment, the top of memory higher than 4GB may be uncacheable instead of write back depending on the board configuration and so on the mapping .
This impacts the PBIT (*)memory tests that may be very slow for the test of this area and may provoke a watchdog reset.

The cache attribute of this area is correctly set to write back when booting an operating system

Workaround: None

Item #5 USB activity during PBIT memory tests may block the board - CRP 4037

Description: USB activity during the PBIT(*) memory tests may block the board and provoke a watchdog timeout.

This is the case if the PBIT memory tests are launched from a graphic console (USB keyboard) or if activity is generated on a USB port during the tests are in progress.

Workaround: Launch the PBIT(*) memory tests from the serial console or launch them automatically by setting the BIOS bootcmd environment variable.

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

Item #6 Garbage characters send on COM1 at poweron if hyper-threading is disabled - CRP 4078

Description: If the CPU straps are changed due to hyper-threading disabled or cores not all activated, then the BIOS reprograms the straps and reset the board.

Garbage characters are send at this time.

Workaround: None.

Item #7 If the primary display is set to PEG in setup then it is reconfigured to AUTO after reset - CRP 4079

Description: If the primary display is set to PEG in setup (default value) to allow redirection of the display to a VPX graphic card then after reset the display is correctly redirected but the primary display selection has been reset to AUTO.

Then, display will be redirected to the internal graphic.

Workaround: Disable the internal graphic but in this case no more graphic outputs if no VPX graphic card is connected.

Item #8 Ethernet Errors on i82579 Interface (ETH0) - CRP 4087

Description: Receive errors may appear on the ETH0 interface. It depends on the NVM contents (Gbe region in SPI flash) starting at word offset 0x40.

Workaround: Apply the i82579 EEPROM rev 0.F4 patched included in BIOS ID13010.

This CRP is fixed in BIOS ID13127 but it must be combined with modifications in the OS Ethernet driver also (driver e1000e under Linux).

Item #9 `kmac` not compatible with `eeupdate` - CRP 4088

Description: When the Intel utility **`eeupdate`** is used combined with the **`kmac`** command, then **`kmac`** does not work correctly.

Workaround: None

Item #10 L1, L2 or L3 cache sizes stored in the SMBIOS table may be wrong - CRP 4133

Description: Randomly the L1, L2 or L3 cache sizes stored in the SMBIOS table may be wrong and the Linux "**`dmidecode -t cache`**" command may return wrong information.

Workaround: None

Item #11 Wrong memory information in SMBIOS table 17 - CRP 4134

Description: Wrong SMBIOS type17 table for channel B information and wrong datawidth info in case of SPD internal mode.

Workaround: None

Item #12 Encryption Standard must be disabled - CRP 4178

Description: BIOS must disable the AES feature to disable cryptography and be compliant with the export control requirement.

Workaround: None

Item #13 Lost send packet on 82579 - CRP 4257

Description: The 82579 Lewisville chip loses some packets in transmission with raw tool test.

The problem depends on the initial state of the link (power on, reboot, or link negotiation).

Workaround: The correction indicated by Intel consists of disabling a power management mode in the initial configuration of the 82579 MAC chip.

Item #14 `kuuid` command erases some setup parameters - CRP 4231

Description: **`kuuid EFI utility`** command must not be used because some setup parameters are modified.

Workaround: If this command is used, then all parameters in **`Kontron`** menu and **`Advanced`** -> **`CPU Configuration`** menu previously set, must be re-entered after using this command.

Corrected in BIOS ID15300

Item #15 Security password does not protect SETUP - CRP 4295

Description: When set into BIOS SETUP the Administrative password should protect the entire SETUP against any modification. This is not the case until BIOS ID15300.

In particular doing Restore Default Parameter will remove the password.

Workaround: None

11.3 BIOS ID12104 Release Notes

The identified problems relative to the BIOS release ID12104 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 22
- CPU Frequency Configuration - Section 5.1 page 14
- UUID Configuration - Section 5.4 page 17
- Internal SPD memory tables implementation - Section 5.12 page 25
- Watchdog timer implementation at OS boot time - Section 5.10 page 23
- SATA ports speed selection - Section 6.2 page 27
- Vital Product Data display - Section 5.5 page 18
- Write Protection Policy display - Section 5.9 page 23

These following are accessible by Kontron EFI commands. Refer to chapter 10 page 41 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 viewer.
- **`kpld`**, CPLD register and I2C device access
- **`ktemp`**, Board temperature display
- **`kvpd`**, Vital Product Data information
- **`kvpx`**, VPX configurator

11.4 BIOS ID12174 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

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

- Display the current CPU frequency in the setup main page instead of the max frequency. Also inquired in the SMBIOS tables
- Add the Kontron EFI shell command “kuuid” to configure UUID
- Remove the unsupported EFI shell commands “ifconfig” and “ping”
- Add the EFI shell commands “time” and “date”

11.5 BIOS ID12235 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

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

- adds the Barton Hills i82580 support in the **kmac** command to manage the MAC addresses and the EEPROM
- adds the external RTC RV8564 management
- adds an automatic probe of the OS and VPD EEPROMs to discover the speed of their CPLD I2C bus
- adds setup controls for RC class boards at the save or restore defaults time

This release also includes the PBIT software(*) V2.1 ID12233 implementing the following evolutions:

- adds the CPLD watchdog timer PBIT(*) test at 1 KHz
- adds the external RTC PBIT(*) test
- improves the PBIT(*) system test

(*) 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 ID12297 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

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

- adds support of Nuvoton NCT7802Y for hardware monitoring
- programs high speed CPLD I2C bus to 2MHz if the OS and VPD EEPROMs are connected to the low speed bus

This release also includes the PBIT software(*) V2.2 ID12270 implementing the following evolutions:

- new test Nuvoton NCT7802Y for hardware monitoring
- vpd test improved: includes MAC addresses checking based on the kmac shell command
- system test improved: BIOS checking (setup, version) and system_edit command added

(*) 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.

NOTICE The release BIOS ID12297 supersedes BIOS ID12275 described in the 3e version of this document in order to fix a compilation problem.

11.7 BIOS ID12347 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

⚠CAUTION

It is mandatory to **kflash** this BIOS with the "**-e**" option, in order to overwrite the Ethernet EEPROM area in system flash:

```
kflash -p -r -e VX3035_ID12347.bin
```

Before flashing, retrieve the MAC addresses by using:

```
kmac -r
```

After flashing, restore the MAC address for the i82579 with:

```
kmac -wf 0x0000DExxxxxx
```

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

- New EEPROM for i82579 (**82579-LM_NVM_LAN-Switch_rev0.F3.bin**). Refer to the warning above.
- Fix CRP 4078: garbage characters are displayed on COM1 when the BIOS performs a reset after setting new cpu straps.
- Fix CRP 4079: if PEG is selected as Primary Display (default setting) to select display to an external graphic card then after reseting the board the AUTO selection is forced.
- Change VCC low limit in Nuvoton to 3,20V

This release also includes the PBIT software(*) V2.3 ID12346 implementing the following evolutions:

- **hwmon** test: change VCC low limit to 3.20V
- Fix **kdiag** problem if both **promptonfail** flag is set and **kdiag** is in loop mode
- Fix bug 6787: **clrallstat** command
- Create a new variable **stopEfiShell** equivalent to **bootdontexit** for documentation. It is used to stop under EFI shell after automatic PBIT execution

(*) 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.

11.8 BIOS ID13010 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

⚠CAUTION

Before flashing, retrieve the MAC addresses by using:

```
kmac -r
```

It is mandatory to **kflash** this BIOS with the "**-e**" option:

```
kflash -p -r -e VX3035_ID13010.bin
```

in order to overwrite the gigabit Ethernet EEPROM area in system flash.

After flashing, restore the MAC address for the i82579 with:

```
kmac -wf 0x0000DExxxxxx
```

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

- New EEPROM revision 0.F4 for i82579 included in BIOS binary.
- CRP 4087 workaround: EEPROM revision 0.F4 for i82579 patched to solve receive errors on ETH0 interface problem.
- **kflash**: changed SPI flash GbE region management for i82579 MAC address saving when updating BIOS.
- **kmac**, CRP 4088: changed SPI flash GbE region management for i82579 MACaddr. Compatibility with Intel **eeupdate** utility.

This release also includes the PBIT software(*) V 2.4 ID13009 implementing the following evolutions:

- changes "**kdiag cfg <runflags>**" command: if no test is listed in the command line then it applies flags on the configured tests instead of the missing tests.

(*) 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.

11.9 BIOS ID13127 Release Notes

The identified or the fixed problems relative to this release are described in the section 11.2 “Known Problems Table” page 86.

⚠CAUTION

Before flashing, retrieve the MAC addresses by using:

```
kmac -r
```

It is mandatory to **kflash** this BIOS with the "**-e**" option:

```
kflash -p -r -e VX3035_ID13127.bin
```

in order to overwrite the gigabit Ethernet EEPROM area in system flash.

After flashing, restore the MAC address for the i82579 with:

```
kmac -wf 0x0000DExxxxxx
```

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

- CRP 4087 fixed: new EEPROM for i82579 (82579-LM_NVM_No-LAN-Switch_rev0.F4.bin)
  Applies Intel Technical Advisory 195 to fix the packet loss issue on i82579. To be combined with uptodate OS driver.
- **kvpx** enhancement: changes command parameters with "**prog**" and "**dump**". "**kvpx -plx_eeprom prog**" programs the PCIe switch EEPROM for Non Transparent mode only as the EEPROM is not loaded by PEX in Transparent mode. But a customer configuration can be programmed if a file name is given. The PCIe switch EEPROM can be now programmed and dumped even on a system controller board.
- Correct CRC bytes in SPD tables.
- Disable ME interface #1 (and so ME interface #2) as ME support is disabled in BIOS. By this way, MEI #1, MEI #2, IDE-R and SOL interfaces are all disabled.
- Set FDM SATA port 4 to default Gen1 speed instead of **NO_LIMIT**.
- Add support to manage the MAC addr for a 1000BASE-T interface in i82580
- Add the shell command "**dmpstore**" to dump the NVRAM variables resident in system flash.
- CRP 4133 fixed: randomly the L1, L2 or L3 cache sizes stored in the SMBIOS table may be wrong and the Linux "**dmidecode -t cache**" command may return wrong information.
- CRP 4134 fixed: wrong SMBIOS type17 table for channel B and wrong datawidth information in SPD internal mode.

This release also includes the PBIT software(*) V2.5 ID13072 implementing the following evolutions:

- System test: Link Training on PCIe bridge may cause PBIT system test failure.
- Add “**kdiag activate**” command to enable customer to activate PBIT on a specific board.

(*) 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.

11.10 BIOS ID13245 Release Notes

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

- CRP 4178: the Advanced Encryption Standard (AES) feature has been disabled for export control purpose
- Adds a PCI IRQ workaround to use legacy PMC under VxWorks.

This release includes the PBIT software(*) V2.5 ID13072. See BIOS ID13127 release notes.

(*) 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.

11.11 BIOS ID15084 Release Notes

The BIOS is based on BIOS 13127 and does not integrate the evolution of BIOS 13245

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

- Fix CRP 4257 lost send packet on 82579
- Support New Flash N25Q064 from Micron (in addition to SST25VF064)
- New feature, in Flash Write Protect Mode (SW1-3 ON) or NVMRO mode check the Flash CRC at boot time. In case of bad CRC then prompt for Operator Command to continue to boot. It will then enter SETUP automatically. The CRC of the Flash must be set with the command "`kflash -cs`". Then a power off must be done and Write protect must be activated before any new power on.
  See Appendix A.4 “Record BIOS CRC into BIOS ROM image and check CRC at boot time" page 101.

This release includes the PBIT software(*) V2.5 ID13072. See BIOS ID13127 release notes.

(*) 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.

11.12 BIOS ID15300 Release Note

The BIOS is based on BIOS 13245.

BIOS updates (compared to 13245):

- Supports new DDR3 4 GB single rank dual DIMM (new SPD)
- Bug7195 for Ethernet packet lost fixed: new EEPROM for i82579 (82579-LM_NVM_No-LAN-Switch_rev0.F4_NOK1_bug7195.bin)
- Fixes CRP#4231/Bug#7119/Bug#7122: kuuid issue/enhancement
- Supports New Flash N25Q064 from Micron (with SST25VF064)
- Fixes CRP4295: Security password does not protect SETUP
- Adds **`ksata`** command (not validated)
- Adds startup up watchdog, requires PLD rev >= 0x0A to work
- Adds Critical Thermal Trip Point in Kontron Menu
- New feature: in Flash Write Protect Mode (SW1-3 ON) or NVMRO mode check the Flash CRC at boot time. In case of bad CRC then prompt for Operator Command to continue to boot. It will then enter SETUP automatically. The CRC of the Flash must be set with the command " **`kflash -cs`** ". Then a power off must be done and **`Write protect`** must be activated before any new power on. See Appendix A.4 “Record BIOS CRC into BIOS ROM image and check CRC at boot time” page 101.

This release also includes the PBIT software(*) V2.6 ID15281 implementing the following evolutions:

- Fixes bug 7595: retries appear randomly with **`ether_loop0 (55)`** test.
- Adds **`kdiag bypass`** feature
- Adds test number when **`kdiag stat`** is called
- Bug #7085: Activation of PBIT lost after RMA.
- PBIT system test upgraded to its last version (New Menu)
- Bug #7096: **`kdiag clear system`** run if PBIT are disabled
- Bug #6947: Watchdog CPLD mode update with CPLD version >=4
- Bug #7198: Problem with Cold/Warm Reset status, now recorded by CPLD
- Supports PCB-D VX3035 with Quad ETH & SFP Cage, no P2, GPIO6,7 ...
- Memory Test: DIMM diagnostic correction for Interleaved case

(*) 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.

12 / Use Cases

This chapter gives some advise for following pratical cases:

- DEPLOY : How to deploy VX3035 - BIOS, section 12.1 page 97
- DEVEL: How to develop applications with VX3035 - BIOS, section 12.2 page 98
- EVAL: How to benchmark VX3035 - BIOS, section 12.3 page 98
- TROUBLESHOOT: How to troubleshoot VX3035 - BIOS, section 12.4 page 98

12.1 DEPLOY: How to deploy VX3035 - BIOS

Deploying with VX3035 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 VX3035 with another one in a system, cloning allows to duplicate VX3035 settings in the new board prior to replacement. This is how to proceed with VX3035:

- **On Original VX3035**

Duplicate the hardware settings. (see VX3035 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 100 of this document to know how to save a BIOS ROM image.

- **New VX3035**

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 99 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 VX3035.

NOTICE

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 VX3035 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 VX3035:

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 VX3035 User's Guide about the board identifcation labels.

See section 5.5 page 18 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.

NOTICE

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 VX3035 - BIOS

TBD

12.3 EVAL: How to benchmark VX3035 - BIOS

TBD

12.4 TROUBLESHOOT: How to troubleshoot VX3035 - BIOS

SETUP not accessible

If setup is not accessible, make sure the board IS operational in rescue mode (see VX3035 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 29 “Boot Method and Priority” of this document

Eventually restore the default manufacturing setup configuration. See chapter 9 page 39 “Save and Exit Menu” to restore setup.

Appendix A - How to Update and Restore BIOS

A.1 Update BIOS from UEFI Shell using USB device

This section details the update of the AMI BIOS Firmware on a VX3035 board. An USB key with the BIOS image to flash will be used.

Operating Mode

- Copy the BIOS image under the USB device
- Boot VX3035 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 **VX3035_IDYYXXX.bin** then flash the BIOS entering command

```
VX3035 > kflash -p -r  VX3035_IDYYXXX.bin
```

⚠CAUTION 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.

NOTICE 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 VX3035 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 VX3035-RESCUE.

- Enter command:

```
VX3035-RESCUE> kflash -c
```

⚠CAUTION 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 VX3035 board. An USB key will be used to store the BIOS image

Operating Mode

- Boot VX3035 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 **VX3035_CLONE.bin** then copy the BIOS image entering command

```
VX3035> kflash -s  VX3035_CLONE.bin
```

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

A.4 Record BIOS CRC into BIOS ROM image and check CRC at boot time

To program a BIOS CRC and check the BIOS CRC validity it is necessary to configure the BIOS Flash into Write Protect Mode.

Do the following instruction:

- Be sure to have the fine BIOS settings. Check your SETUP configuration and your BIOS environment variable, also take care of the BIOS boot device order. It will not be possible to add a new boot device after the Flash is set in Write Protect.
- Program the BIOS CRC with **kflash -cs**

```
VX3035> kflash -cs
```

- Power Off the board
- Set Flash Write Protect Mode (WP) (with switch SW1-3 ON)
- Power On and check CRC is verified at boot time (CRC checking takes 7s)

If the CRC is correct the board boots normally. If the CRC is not correct the boot is stopped and the operator is asked to Press the <Enter> key to continue. SETUP is then entered and allows the operator to take the appropriate action.

About Kontron

Kontron, a global leader in embedded computing technology and trusted advisor in IoT, works closely with its customers, allowing them to focus on their core competencies by offering a complete and integrated portfolio of hardware, software and services designed to help them make the most of their applications.

With a significant percentage of employees in research and development, Kontron creates many of the standards that drive the world's embedded computing platforms; bringing to life numerous technologies and applications that touch millions of lives. The result is an accelerated time-to-market, reduced total-cost-of-ownership, product longevity and the best possible overall application with leading-edge, highest reliability embedded technology

Kontron is a listed company. Its shares are traded in the Prime Standard segment of the Frankfurt Stock Exchange and on other exchanges under the symbol "KBC".
For more information, please visit: **www.kontron.com**

CORPORATE OFFICES

EUROPE, MIDDLE EAST & AFRICA

Lise-Meitner-Str. 3-5
86156 Augsburg
Germany
Tel.: + 49 821 4086-0
Fax: + 49 821 4086-111
info@kontron.com

NORTH AMERICA

14118 Stowe Drive
Poway, CA 92064-7147
USA
Tel.: + 1 888 294 4558
Fax: + 1 858 677 0898
info@us.kontron.com

ASIA PACIFIC

1~2F, 10 Building, No. 8 Liangshuihe 2nd Street,
Economical & Technological Development Zone,
Beijing, 100176, P.R. China
Tel.: + 86 10 63751188
Fax: + 86 10 83682438
info@kontron.cn